ECE 429/629 Organization of Computers
Fall 2004

Instructor: Stan Birchfield, 207A Riggs Hall, 656-5912, stb at clemson
Office Hours: 10:45-11:45 TTh, or by appointment

Grading assistant: TBD

Class meets: 9:30-10:45 TTh, 227 Riggs Hall

Website: **http://www.ces.clemson.edu/~stb/ece429**

Text:

- Hennessy and Patterson, *Computer Architecture: A Quantitative Approach*, 3rd ed., Morgan Kaufmann, 2003 (required)
- Patterson and Hennessy, *Computer Organization and Design: The Hardware / Software Interface*, 2nd ed., Morgan Kaufmann, 1998 (recommended, on reserve at the library)

Prerequisites: Basic computer architecture

Overview: This course introduces the principles of advanced computer architecture. Students are expected to enter this class with a basic understanding of computer architecture, along with assembly language. Building upon these fundamentals, the students will learn advanced architectural techniques for making computers run orders of magnitude faster than would be possible from technological improvements alone.

Objectives: By the end of the course, students should be able to do the following:

- *Fundamental concepts.* Describe the basic concepts of processor architecture, including datapath, control, instruction set, pipelining, and memory hierarchy. Explain the difference between RISC and CISC, their historical development, factors to consider in designing a processor, and effective methods for evaluating processor performance. Calculate performance using the Iron Law, Amdahl's Law, and SPEC ratings.
- *CPU internals.* Describe how to build a CPU from individual transistors. Determine values of control and data lines on single-cycle, multi-cycle, and pipelined CPUs. Identify pipeline hazards from a piece of MIPS assembly code, and describe solutions to overcome them. Determine stage of execution for each instruction using scoreboarding and Tomasulo's algorithm.
- *Memory hierarchy.* Explain the gap in processor-memory performance. Identify software and hardware techniques for minimizing the impact of the gap. Compute state of cache after accesses.

Grading: homework (20%), two midterms (50%), final exam (30%)

Topics:	**Lectures**
• introduction	1
• performance evaluation	3
• instruction sets	2
• datapath and control	4
• pipelining	6
• dynamic scheduling	3
• software techniques for ILP	2
• memory hierarchy	4
• thread-level parallelism	2
• tests	2

	29

Academic integrity: "As members of the Clemson University community, we have inherited Thomas Green Clemson's vision of this institution as a 'high seminary of learning.' Fundamental to this vision is a mutual commitment to truthfulness, honor, and responsibility, without which we cannot earn the trust and respect of others. Furthermore, we recognize that academic dishonesty detracts from the value of a Clemson degree. Therefore, we shall not tolerate lying, cheating, or stealing in any form." – from the university's web page on academic integrity: http://www.cs.clemson.edu/html/academics/academic_integrity_2002.html