ECE 453 - Advanced Computer Architecture II: Multiprocessor Design
Winter 2006

Course Description

This course examines fundamental issues and design trade-offs in multiprocessor architecture. It covers both quantitative and qualitative analysis of parallel computer systems and serves as a entry point to further research in high performance computing.

Course material covers both traditional topics in parallel architectures such as parallel programming models, symmetric multiprocessors, coherency policies, distributed shared memories, and scalable multiprocessors, as well as important emerging topics like chip multiprocessors and on-chip interconnection networks. The project component is open-ended and students are encouraged to draw on their own research interests and prior background for inspiration.

Instructor

Professor Russ Joseph
Office: Tech L467
Email: rjoseph@ece.northwestern.edu
Phone: 1-3061
Office Hours: Friday 2:00pm-4:00pm
(I have an open door policy. If my door is open, I will be happy to answer quick questsions. If my door is closed – I will probably tell you to go away.)

Class Meetings

Day: Tuesdays and Thursdays
Time: 3:30pm-5:00pm
Location: Tech LG72

Prerequisite

ECE 361 or equivalent knowledge of uniprocessor architecture. Note: ECE 358 (Parallel Programming) and ECE 452 (Advanced Computer Architecture I) are not required.

Required Textbook

David E. Culler and Jaswinder Pal Singh with Anoop Gupta, *Parallel Computer Architecture: A Hardware/Software Approach*, Morgan Kaufmann Publishers, 1999

Tentative Schedule

Date	Topic
1/3	Introduction to Parallel Architectures
1/5	Programming Models and Communication Architectures
1/10	On-Chip Thread Parallelism: SMT and CMP
1/12	Discussion: Asymmetric CMPs
1/17	Coherence and Consistency
1/19	Basic Bus Based Coherence
1/24	Implementing Bus Based Coherence
1/26	Exam 1
1/31	Scaleable Systems and Directory Based Coherence
2/2	Discussion: Tradeoffs in Replication vs. Communication
2/7	Implementing Directory Based Coherence
2/9	Discussion: Fault Recovery for Distributed Shared Memory Systems
2/14	Instructor Out Of Town: No Lecture
2/16	Basics of Interconnection Networks
2/21	Advanced Interconnection Networks
2/23	Discussion: Designing Interconnection Networks for CMPs
2/28	Implementing Synchronization
3/2	Exploiting Other Kinds of Parallelism
3/7	Exam 2
3/9	Project Presentations

Lectures

The primary delivery vehicle in this course will be lectures given by the instructor. The lectures will cover material presented in the text, as well as information from published papers and articles. Partial lecture notes will be made available.

Readings and Discussion Sessions

Students will be assigned readings of both recent and classic research papers from the field. They will take turns leading discussion sessions on these papers.

Homework

Ungraded homework assignments will be posted periodically. Solutions will be made available approximately one week later. These exercises are meant for students to test their own understanding of the course material and serve as practice for the in-class exams.

Exams

There will be two in-class exams during the quarter. They will be similar in style to the homework assignments, but will be "closed-notes" exams.

Projects

Students will work on a quarter-long research project in multiprocessor design. The first component is an assigned excercise which introduces students to a multiprocessor simulator. The second component is an open ended research project which will allow students to propose and evaluate multiprocessor systems. Students may work independently or in groups of two.

Grading

Grades will be assigned according to the following distribution:

Homework	0%
Exams	50%
Project	40% (10% + 30% for the two components)
Participation	10%

Course Objectives

The goal of this course is to provide students with (1) a broad understanding of parallel computer architecture and (2) to the extent possible, an understanding of the current state-of-the-art in parallel computer architecture. Specifically, upon completion of this course, a student should have developed:

1. Broad understanding of the design of parallel computer systems, including modern parallel architectures and alternatives.
2. Understanding of the interaction amongst parallel architecture, applications, and technology.
3. Understanding of a framework for evaluating design decisions in terms of application requirements and performance measurements.
4. Gain experience with using and modifying a parallel computer simulator.
5. Gain experience on identifying and pursuing a research project.
6. Gain experience with writing a research report.
7. Gain insight on evaluating research papers.
8. Gain experience in technical presentations.