

## **EE653 -- Advanced Topics in Microarchitecture**

**Fall 2016**

**Instructor: Murali Annavaram**

Office: EEB232/Tel:(213) 740-3299

E-mail: annavara@usc.edu

Office hours: Just email to make an appt

**Preliminary Syllabus--subject to change**

**Final syllabus will depend on the number of students in the class.**

### **1. Overview**

This course bridges the gap between EE557 and the current research on microarchitecture. It focusses on research in microprocessor design to speed up the execution of instructions in various contexts and under power, complexity, reliability and cost constraints. Current computer architecture research has been extensively focused on parallel-everything; chip multiprocessors, multithreading, GPUs, parallel software etc., Of course there is continued research that targets improvements to OOO processors, including scheduling, memory wall, multithreading, speculative execution, heterogeneity, and branch prediction and recovery.

While performance improvements still play a large role in design exploration, power is an equally important parameter. As technology scales new challenges arise. Thus research has focused on novel architectural techniques to improve power dissipation, temperature control, noise and reliability, and to deal with the impact of wire delays.

In future Moore's law as it applies to computing may continue to hold valid by increasing the number of threads executing in parallel in a CMP/MT micro-architecture. Exploiting this ever-increasing concurrency together with architectural support to solve technological problems (e.g., power, reliability, wire delays) and to help the programming problem are the challenges of micro-architecture research in the next decade.

### **2. Textbooks**

Dubois, Annavaram and Stenström: "Parallel Computer Organization and Design" Cambridge University Press, 2012. ISBN: 978-0-521-88675-8. Purchase from the USC bookstore or from Amazon.com. REQUIRED. Background material is in this book. Material not covered in EE557 will be taught from this book. Also some problems and reading assignments will be picked from the book.

A list of required readings (probably just a few papers) and notes will be posted on the blackboard, from which copies can be downloaded.

### **3. Prerequisite**

EE557. Computer System Architecture.

If you did not take EE557 with me or Professor Dubois, please review the following material from the book: chapters 1, 3, 4, 6, and 7 (pp. 342-379, 388-410) and don't hesitate to ask questions if you have any.

### **4. Venue and time**

KAP 167--MW 3:30-4:50PM.

## **5. TA**

TBA Office hours: TBD. The TA will help with simulation environments and projects.

## **6. Project**

There will be just a couple of research projects that students can choose from to work on during the entire semester. The project will involve simulation. Each team will report on their research as part of the final exam. At the end of the each team will have 15 minutes to present its project results.

## **7. Course Work**

The course has three major components.

1. Regular lectures at first on material from the book (basics) and then on research papers (advanced topics). There will be one midterm (possibly two) about the material covered in lectures. Mid-terms are take-home.
2. An independent research project. More information about the research areas, simulation tools and project descriptions will be made available as we move along through the semester. The grade on the project will be shared by all the students in each team.
3. There will be two or three programming assignments on programming highly parallel systems such as GPUs using CUDA and CMPs using OpenMP and PIN based simulation tools.
4. There will be several in-class short quiz exams.

## **8. Grading Policy (Tentative)**

Midterm 1 & 2: 20% each; Programming assignments: 20%; Project: 30%; Quiz: 10%.

Class attendance is expected.

## **9. Statement for Students with Disabilities**

Any student requesting academic accommodations based on a disability is required to register with Disability Services and Programs (DSP) each semester. A letter of verification for approved accommodations can be obtained from DSP. Please be sure the letter is delivered to me (or to TA) as early in the semester as possible. DSP is located in STU 301 and is open 8:30 a.m.-5:00 p.m., Monday through Friday. The phone number for DSP is (213) 740-0776.

## **10. Statement on Academic Integrity**

USC seeks to maintain an optimal learning environment. General principles of academic honesty include the concept of respect for the intellectual property of others, the expectation that individual work will be submitted unless otherwise allowed by an instructor, and the obligations both to protect one's own academic work from misuse by others as well as to avoid using another's work as one's own. All students are expected to understand and abide by these principles. Scampus, the Student Guidebook, contains the Student Conduct Code in Section 11.00, while the recommended sanctions are located in Appendix A: <http://www.usc.edu/dept/publications/SCAMPUS/gov/>. Students will be referred to the Office of Student Judicial Affairs and Community Standards for further review, should there be any suspi-

tion of academic dishonesty. The Review process can be found at: <http://www.usc.edu/student-affairs/SJACS/>.

### **11. Web Resources[please help add to the list!]**

<http://www.princeton.edu/~jdonald/research/cmp/> :links to publications on CMPs

<http://www.princeton.edu/~jdonald/research/hyperthreading/> :publications on CMTs

<http://www.cs.wisc.edu/arch/www/> : tools, people, resources

<http://www.itrs.net/Links/2005ITRS/ExecSum2005.pdf> : the road map for semi-conductors.

<http://www.cs.wisc.edu/~markhill/conference-talk.html> : how to prepare a talk.

## 12. Tentative Course Schedule:

Lecture	Topics	MILESTONES/REMARKS
Lect 1	Introduction to CMPs	Read Chapter 8
Lect 2	CMT/CMPs	
Lect 3	CMP programming models	
Lect 4	OpenMP	Programming assignment#1
Lect 5	CUDA & Simulation Models (PIN)	
Lect 6	Full system and Parallel Simulations	Project selection
Lect 7	Quantitative evaluations	
Lect 8	Transactional memory	Programming assignment#2
Lect 9	Transactional memory	
Lect 10	Thread level speculation	
Lect 11	Thread level speculation	
Lect 12	Advanced topics in Coherence	Read Chapter 5 and 7
Lect 13	Advanced topics in Coherence	
Lect 14	Consistency models	
Lect 15	Consistency models	
Lect 16	Datacenters	MIDTERM 1. On material up to Lect 11
Lect 17	Datacenters	
Lect 18	Datacenters	
Lect 19	Virtualization	Project mid-way status#1
Lect 20	Virtualization	
Lect 21	Virtualization	Programming assignment#4
Lect 22	Flash memory	
Lect 23	Flash memory	
Lect 24	Accelerators	Project mid-way status#2
Lect 25	Accelerators	MIDTERM 2. On material up to Lect 23
Lect 27	Two overflow classes	
Lect 28	Two overflow classes	
Lect 29	END OF CLASSES	FINAL project presentations.