Department of Electrical Engineering
University of Southern California
EE 533 — Network Processor Design and Programming

Instructor: Young H. Cho, Research Assistant Professor
Office: USC/Information Sciences Institute, 4676 Admiralty Way, 1147
Marina del Rey, Ca 90292, 310-448-9107
E-mail: youngcho@isi.edu

The web page will contain course announcements, laboratory assignment and relevant handouts.

Course Objectives: Understand network processor architecture, applications, and other relevant issues. Program network processor and test under realistic network environment. Design and deploy custom network processor.

Pre-requisite: EE457 (or approval from the instructor)
Recommended Courses: EE450
Other Requirements: experience in C programming and experience in hardware description languages such as Verilog and VHDL

Grading:
25% Attendance/Quiz
10% Reading assignments
25% Laboratory assignments
10% Final Presentation
30% Final Project Report and Demo
Final grades will be assigned by a combination of student score and the discretion of the instructor.

Preparation for classes:

- Students will be using Linux based system through the course. It is strongly recommended that the students become familiar with its navigation and use.
- Some of the laboratory assignments will require the use of C/C++ under Linux environment. It is recommended that students become familiar with the language and typical development environment.
- It is recommended that the students become familiar with some form of hardware description languages.

Grading policies:

- **Late Policy**: Any late but completed assignments result in a 50% of grade.
- **Participation Grade**: Individualized grade will be assigned during the demo/progress meetings as well as lecture times.
- **Grade Adjustment**: If you dispute any scoring of a problem on an exam or homework set, you have one week from the date that the graded paper is returned to request a change in the grade. After this time, no further alterations will be considered. All requests for a change in grade must be submitted in writing to me.
- **Changes/Information**: The student is responsible for all assignments, changes of assignments, announcements, lecture notes etc. All such changes should be posted on the course web-site.
- **Other**: As per university guidelines published in SCampus, the academic integrity policy will be upheld.

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.

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

Syllabus:
The course will be driven mainly by the lecture materials and assigned reading materials (key publications in the field).

Weekly Lectures and Reading List (May be updated year to year.)

No.	Lectures	Assigned Readings
1-1	Introduction to Network Processor Course	L. White, et al, "An Integrated Experimental Environment for Distributed Systems and Networks", OSDI 2002, Boston, MA, Dec 2002.
1-2	Computer Networks and Internet	
2-1	Measuring Computer Networks	S. J. Lee, P. Sharma, S. Banerjee, S. Basu, and R. Fonseca, "Measuring bandwidth between planetlab nodes," Passive and Active Network Measurement, p. 292–305, 2005.
2-2	Computer Network Emulation and Testbed	
3-1	Digital Logic Design and Memory	Xilinx, "ISE In-Depth Tutorial," http://www.xilinx.com/direct/ise10_tutorials/ise10tut.pdf
3-2	Hardware Description Languages	
4-1	Field Programmable Gate Array	Y. H. Cho, "Optimized Automatic Target Recognition Algorithm on Scalable Myrinet/Field Programmable Array Nodes." 34th IEEE Asilomar Conference on Signals, Systems, and Computers, Monterey, CA, October 2000.
4-2	Streaming Processing and Hardware	
5-1	Reconfigurable Hardware in Emulation Environment	DETER team, "Cyber defense technology networking and evaluation", In Communications of the ACM, Special issue on Emerging Technologies for Homeland Security, Vol. 47, Issue 3, pp 58-61, March 2004.
5-2	Introduction to NetFPGA	
6-1	Hardware Software Co-designing	G. Watson, "NetFPGA: A Tool for Network Research and Education", 2nd Workshop on Architecture

6-2	Hardware Reference IP Router Design	Research using FPGA Platforms (WARFP) February, 2006.
7-1	Pattern Match based Network Monitoring	A. Goodney, S. Narayan, V. Bhandwalkar, and Y. H. Cho, "Pattern Based Packet Filtering using NetFPGA in DETER Infrastructure," First Asia NetFPGA Developers' Workshop, Daejeon, Korea, June 2010.
7-2	Advanced Network Intrusion Detection System	
8-1	Interface via Hardware and Software Registers	P. Crowley, M. E. Fluczynski, J. L. Baer, and B. N. Bershad, "Characterizing processor architectures for programmable network interfaces," in Proceedings of the 14th international conference on Supercomputing, 2000, p. 54–65.
8-2	Essentials of Computer Architecture	
9-1	Multi-core Processor Architectures and Design	M. Labrecque, J. G. Steffan, G. Salmon, M. Ghobadi, and Y. Ganjali, "NetThreads Routing Edition: Programming NetFPGA with Threaded Software." NetFPGA Developers' Workshop, 2009.
9-2	Architecture for Hardware Multiple Threads	
10-1	Network Processor Design	M. Labrecque, J. G. Steffan, G. Salmon, M. Ghobadi, and Y. Ganjali, "NetThreads: Programming NetFPGA with threaded software," NetFPGA Developers' Workshop, 2009.
10-2	Advanced Network Processor Technologies	
11-1	Hardware Accelerators for Computer Networks	Moscola, J. et al., "Reconfigurable Content-based Router Using Hardware-Accelerated Language Parser", ACM Transaction on Design Automation of Electronic Systems on Demonstrable Software Systems and Hardware Platforms, Volume 13, Number 2, April 2008.
11-2	Integration with Interrupts and Exceptions	
12-1	Commodity Network Processors	M. Adiletta, M. Rosenbluth, D. Bernstein, G. Wolrich, and H. Wilkinson, "The next generation of Intel IXP network processors," Intel technology journal, vol. 6, no. 3, p. 6–18, 2002.
12-2	IXP Network Processor Architecture	
13-1	Custom Network Processor	McKeown, N. et al, "OpenFlow: enabling innovation in campus networks", ACM SIGCOMM Computer Communication Review, vol. 38, New York, NY, April 2008.
13-2	Introduction to Openflow Switches	
14-1	Openflow on NetFPGA	D. Culler, D. Estrin, and M. Srivastava, "Guest editors' introduction: overview of sensor networks," Computer, pp. 41–49, 2004.
14-2	Network Processing in Mobile Embedded Systems	
15-1	Future of the Computer Networking	P. Molinero-Fernández, N. McKeown, and H. Zhang, "Is IP going to take over the world (of communications)?," ACM SIGCOMM Computer Communication Review, vol. 33, no. 1, pp. 113–118, 2003.
15-2	Computer Networks HW/SW Engineering	

Laboratory #	Topic
1	DETER Tutorial (week 1)
2	Advanced DETER Tutorial (week 1)
3	Network performance measurement and tuning in DETERlab(week 2)
4	Xilinx ISE/Verilog Tutorials (week 2-3)
5	Mini Network Intrusion Detection System (HW Accelerator) on NetFPGA/DETER (week 4-5)
6	Custom RISC-V/Compiler based Processor in NetFPGA/DETER (week 6-7)
7	Special Network FIFO Design and integration with Custom Processor (week 8)
8	Network Processor Integration in NetFPGA/DETER (week 9)

9 Final Project – Accelerated Networking Application on DETER using High Performance Multicore Processor/Multiple Threaded Network Processor with HW Accelerator (week 10-16)

Course Project Description: The expected result of the final project is a working implementation of a relevant networking application running on your custom IP router with multiple-core/multiple threaded Network Processors and Hardware Acceleration modules in a realistic network environment within DETER testbed. The final project must be approved by the instructor before you begin the project. Working as a group is permitted, and a team can consist of 2-3 persons. The grade will be based on the demonstration of your system as well as the final project report that you will submit at the last day of the finals.

Timeline
Week 2: Identifying team members and project topics
Week 11: Proposal due (team member, topics and milestone)
Week 12-16: Weekly report/demo due (demonstration of intermediate results)
Week 17: Project presentation (all students must attend)
Week 18: Final report due (use the provided template)

Sample project
"*Hardware Accelerated Deep Packet Inspection with 8-threaded Dual core Network Processor*": the goal of the project is to develop a custom hardware based IP Router augmented with Dual core 4-way hardware multi-threaded network processor. The network appliance will execute multiple network security software simultaneously as the packets are sent and routed through the router over 1Gbps under DETER network emulation testbed.