Course Objective:
This unique course aims to prepare students with skills and disciplines, instead of domain knowledge, in network system design and evaluation. The skills and disciplines would enable students to do both system development/research and algorithmic research in either academia or industry. 6 topics are covered in two tracks. In the design track, we cover major system development issues from IC design to software techniques for BSP (board support package) and turn-key solutions, with each one lectured in 2 or 3 weeks where the front part and rear part are on concepts and tools/projects, respectively. In the evaluation track, the skills of benchmarking, simulation, and modeling are presented, with each one lectured in 2 weeks. 2 weeks are allocated, before the final exam, for student presentations of their term projects.

Students are asked to do a mini-project (could be finished in 3 hours) after the lecture of each topic, demo and turn in a 2-page report to TA 2 weeks later. Students need to select one term-project topic before 4/15, and turn in an 8-page report and present in the class on 6/1 or 6/8. Both mini-projects and term-projects are done individually. The final exam is open-book and contains 6 problems asking how and why on the covered 6 methodologies and their tools.

Prerequisite: Computer Networks or equivalent
Textbook: On-line course slides
Grade: 6 mini-project reports (42% = 7% x 6), 1 8-page term-project report & presentation (30%), and 1 final exam (28%)
Course homepage: http://speed.cis.nctu.edu.tw/~ydlin/course/cn/nsd.html
Class hours: Monday 9:00AM-12:00NN in EC427

Course agenda:

<table>
<thead>
<tr>
<th></th>
<th>Topic</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Network IC Design</td>
<td>3/2, 3/9</td>
</tr>
<tr>
<td>2</td>
<td>BSP (Board Support Package)</td>
<td>3/16, 3/23, 3/30</td>
</tr>
<tr>
<td>3</td>
<td>Turn-Key Solutions</td>
<td>4/6 &amp; 4/13</td>
</tr>
<tr>
<td>5</td>
<td>Benchmarking</td>
<td>4/27 &amp; 5/4 (part of)</td>
</tr>
<tr>
<td>6</td>
<td>Simulation</td>
<td>5/4 (part of) &amp; 5/11</td>
</tr>
<tr>
<td>7</td>
<td>Modeling</td>
<td>5/18 &amp; 5/25</td>
</tr>
<tr>
<td>8</td>
<td>Term-Project Presentations</td>
<td>6/1 &amp; 6/8</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(8 min per student)</td>
</tr>
<tr>
<td>9</td>
<td>Final Exam</td>
<td>6/15</td>
</tr>
</tbody>
</table>

Course Outline:
I. Design Track

1. NIC (Network Interface Card):
   Topics: Hardware design, block diagram, state machine, Verilog programming, Ethernet MAC
   Tools: Verilog related tools
   Mini-Project: Compile/simulate/synthesize Ethernet MAC at opencores.org
   Term-Project: Write a simulation or testing module for Ethernet MAC

2. BSP (Board Support Package):
   Topics: BSP, bootloader, embedded OS and device drivers
   Tools: GNU tool chain, iBoot, embedded Linux for ARM
   Mini-Project: Compile/Install/Configure bootloader and embedded Linux kernel on target
   Term-Project: Modify driver or bootloader to enhance its functions or performance

3. Turn-Key:
   Topics: Integrating connectivity, QoS, and Security, and Image building
   Tools: CVS, uclibc/glibc, binutils, gcc, buildroot, and application packages
   Mini-Project: Build an embedded Web server (thttpd) image with a video content by BuildRoot
   Term-Project: Build an embedded firewall (iptables) with transparent proxies (squid, dansguardian) by BuildRoot

II. Evaluation Track

4. Benchmarking:
   Topics: Black box testing and white box profiling
   Tools: WebBench, Email Test Tool, Smartbits, Avalanche, gprof, kprof
   Mini-Project: Benchmarking the Apache Web server by WebBench
   Term-Project: Bottleneck analysis of the anti-virus proxy (Postfix + AmaVis + ClamAV) by gprof and mail testing tool

5. Simulation:
   Topics: Simulation models, and result interpretation
   Tools: ns-2 and CPN (Color Petri Net)
   Mini-Project: Simulating TCP on ns-2 & simulating a simple protocol on CPN
   Term-Project: Simulating and comparing RED and FIFO on ns-2,
   Or simulating dining philosophers on CPN

6. Modeling:
   Topics: System modeling and Markov chain
   Tools: Matlab7
   Mini-Project: Computing the Markov chain steady state probability of a dual-core processor with a coprocessor
   Term-Project: Extend the mini-project by automatically generating the transition rate matrix and comparative analysis