Winter 2018 Special Topics

CRN	Title	Units	Instructor	Days/Time
52477	289K – Terahertz and mm-Wave Integrated Circuit Design	4	Momeni	TR 4:40-6:00pm
74444	289Q – Modern Parallel Computing	4	Owens	TR 10:30-11:50am

Title: Terahertz and mm-Wave Integrated Circuit Design

4 units – Winter Quarter

Prerequisite EEC 132A and EEC 112

Grading: Letter, Design projects (50%), Homework assignments (20%), Mid-term exam (20%), and in-class presentation (10%).

Catalog Description: Brief overview of classical and emerging applications at terahertz (THz) and mm-wave frequency range, Fundamental theory of RF transmitter and receiver including noise figure, nonlinearity and transceiver architectures, Antenna arrays, Design challenges and limitations of passive and active components for mm-wave and terahertz systems, Fundamental limitations in amplifiers and oscillators, Theory and design of mm-wave and THz amplifiers, oscillators and signal sources.

Expanded Course Description

I- Introduction
A- THz and mm-wave applications
B- High-speed Integrated circuits technologies
C- Review of active devices such as small-signal model and frequency response

II- System and Device Specifications
A- Effect of nonlinearity in transceivers such as gain compression and blocking
B- Noise probabilistic definitions and modeling in circuit analysis
C- Noise source correlations and its effect in mm-wave and THz systems
D- Noise sources in CMOS transistor
E- Noise figure and dynamic range definitions and analysis
F- Transmitter and receiver architectures used in THz and mm-wave systems
G- Phased array systems
H- Design challenges and fundamental limitations of passive components for mm-wave and THz systems

III- Signal Amplification
A- Mason's invariant function
B- Maximum oscillation frequency (f_{max})
C- Power gain definitions and relations
D- Power gain limits of active devices
E- Device unilateralization and gain-boosting techniques
F- Tuned amplifier design
G- Noise figure and power gain calculations in tuned amplifiers

IV- Signal Generation
A- Oscillation mechanisms and theory in self-sustained oscillators

B- Signal swing analysis in resonator-based oscillators
C- Oscillator design for frequencies close to f_{max}
D- Harmonic oscillators for mm-wave and THz signal generation
E- Voltage controlled oscillators and the design challenges at high frequencies
F- Frequency multipliers for mm-wave and THz signal generation

Textbooks: Relevant reading material and exercises will be provided by the instructor. Some of the materials are covered in the following textbooks:

1- RF Microelectronics, by Behzad Razavi, Prentice-Hall, 2011.
2- The Design of CMOS Radio-Frequency Integrated Circuits, by Thomas H. Lee, Cambridge University Press, 2003.

Statement of Course Design Projects:

Two projects are assigned, one on mm-wave amplifier and the other on mm-wave oscillator design. The students are expected to use the material learned in the class and do a thorough research to come up with the best circuit architecture for the given specifications. The students are encouraged to be innovative and to understand the detailed operation of the circuit architecture before using it. Next, they use Cadence simulation to design the circuits and verify the functionality. At the end of the quarter students choose to present one of their projects to the class. Their presentation is graded based on different criteria such as clarity and organization.

Instructor: Omeed Momeni, Jane Gu

Course Overlap: There are overlaps in sections II-A, II-E, and II-F with Prof. Gu's graduate course. However materials in these sections are very fundamental to any RF system and would be very useful if covered from different angles in different courses. Moreover these sections provide critical background for the rest of the sections covered in this course. There are also minor overlaps with EEC211 in section II in noise, with EEC222 in section I in small signal models.

EEC 289Q – Modern Parallel Computing

4 units – Winter Quarter; alternate years
Lecture: 3 hours
Prerequisite: Required: ECS 30; optional but desirable: EEC 170 or ECS 154a.
Grading: assigned projects (30%), final project (60%), classroom participation (10%).

Catalog Description:

Exploration of the architecture of modern parallel computers, their programming models, and their programming systems.

Expanded Course Description:

This course focuses on modern parallel computing and for this particular offering of the course, GPU computing. We will explore using the programmable GPU as a parallel computer, primarily using the CUDA programming language (an extension to C/C++). We will cover the architecture of the GPU and its programming model; the CUDA programming language; fundamental data structures and algorithms on the GPU; numerous application domains and how they can be expressed on the GPU; programming models and high-level languages for GPU computing; and current research challenges in GPU computing. We expect that students who successfully complete this course will be ready to use GPU computing in their own projects and research, and/or be ready to conduct GPU-computing research on their own.

We assume that students who take the course will have experience using the C/C++ programming language; prior experience in any of computer architecture, computer graphics, algorithms, and data structures will also be useful. All assignments will use C/C++.

The assigned projects in this class will contain significant design elements that allow students to design and implement parallel solutions to computationally challenging problems, to analyze and improve their performance, and to use these solutions to understand the architectures, programming models, and programming systems of these modern parallel computers.

1. History of modern parallel computing
   - Hardware predecessors (vector machines, massively parallel machines, graphics processors)
   - Software predecessors
2. Architecture of modern parallel processors
3. Programming model of modern parallel processors
4. Programming systems for modern parallel processors
5. Fundamental parallel primitives
6. Survey of computational motifs ("dwarfs") and parallel implementation strategies
7. Optimization techniques
8. Heterogeneity and multi-node issues
9. Application case studies
10. Future directions

Course Overlap:

EEC 171, EEC 270, and EEC 277 all summarize the material in this class at a high level (for each of these courses, roughly 5-10% of class time is spent on modern parallel computers, mostly from the hardware perspective). ECS 158 covers the programming of numerous parallel architectures, one of which is the GPU. ECS 223 covers parallel algorithms, some of which could be applicable in this course.