Course: ENPM808R – Design and Synthesis of Digital Systems
Semester: Spring 2014
Day(s): W
Time: 7:00pm to 9:40pm
Location: JMP 2120
Instructor: Dr. George Zaki
Phone: 
Email: gzaki@umd.edu

IMPORTANT: when sending email to me about course-related matters, please be sure to include a descriptive subject, and start the subject with “ENPM 808R” (without the quotes). This will make it easier for me to quickly identify course-related messages for faster response.

Course Description

Design methodology for modern digital systems is based heavily on use of hardware description languages (HDLs), such as Verilog and VHDL, and use of automated synthesis from HDL programs into implementations on field programmable gate array (FPGA), and application- specific integrated circuit (ASIC) platforms. This trend towards HDL-based digital system design has been driven by the complexity of modern digital integrated circuits, and advances in the simulation and synthesis capabilities provided by electronic design automation (EDA) tools.

This course will introduce students to HDL-based design of modern digital systems, and will cover in depth the design and implementation of digital systems using the Verilog HDL. Students will learn fundamental concepts of the Verilog language; modeling of complex digital systems; simulation and verification; and Verilog coding styles for synthesis. Hands-on experience will be developed through practical designs, exercises, and projects. Students will use state-of-the-art EDA tools to design, simulate, and test digital systems. The latter part of the course will focus on customized programmable platforms such as graphics processors (GPUs) multicore platforms and FPGAs as well as coding, building, and debugging for such platforms.

Required/Recommended Textbooks

- Andy Oram, Mike Loukides, *Programming With GNU Software*, O'Reilly Media. *(optional)*

Course Outline

Course Overview
Introduction to Design and Synthesis of Digital Systems
Introduction to Unix Concepts and Course Software Environment
Shell Scripting and Cross-platform Unit Testing
Verilog Tutorial and Syntax
Logic Synthesis, Behavioral Modeling, and Structural Modeling
Synthesis of Combinational and Sequential Logic
Verilog Timing Model, Event-Driven Simulation, Tasks and Functions
Timing Issues of Digital System Design
Globally Asynchronous Locally Synchronous (GALS) Design Style
FPGA Tutorial and Synthesis
General Purpose vs. Specialized Multicore Programming
Graphics Processors
CUDA - Building, Running, Debugging

Grading Guideline:
- (15%) In-class quizzes
- (10%) In-class paper presentation
- (15%) Project 1 and report
- (25%) Project 2 and report
- (35%) Final project, report, and oral presentation

Academic Integrity
From The Code of Academic Integrity:
Academic dishonesty is a serious offense which may result in suspension or expulsion from the University. In addition to any other action taken, such as suspension or expulsion, the grade XF denoting “failure due to academic dishonesty” will normally be recorded on the transcripts of students found responsible for acts of academic dishonesty.

Unless otherwise stated, all quizzes, exams, programming assignments and any other assignments are individual assignments: collaboration is not permitted unless explicitly stated on the assignment handout. Students may discuss among themselves concepts pertaining to the programming assignments. However, at no point, should any code, pseudocode, or anything that resembles code be exchanged.

Students with Disabilities
If you have a documented disability and wish to discuss academic accommodation with me, please contact me as soon as possible and no later than the end of the second week.