MARYLAND APPLIED GRADUATE ENGINEERING

Professional Master of Engineering, University of Maryland at College Park

ENPM674: Design and Synthesis of Digital Systems

Spring 2021

General Information

Number of credits: 3

Time and location: Tuesday, 7:00pm to 9:40pm, JMP 2116

Instructor: Dr. George Zaki

Office: TBD

E-mail: gzaki@umd.edu

Office hours: Online: Monday and Thursday at 7pm, or email to schedule

IMPORTANT: when sending email to me about course-related matters, please be sure to include a descriptive subject, and start the subject with “ENPM 674" (without the quotes). This will make it easier for me to quickly identify course-related messages for faster response.

Course Description

Design methodologies and platforms for modern embedded digital systems have been evolving over time. In order to meet the system specifications, many models and tools exist to automate the design space exploration phase and bridge the implementation gap between having a DSP algorithm and final system realization.

In this course we will introduce students to modern tools and platforms used to implement embedded digital systems. We will start by explaining different methods for DSP algorithm modeling, scheduling, and automated unit testing. Using hands-on experience developed through practical designs, exercises, and projects; we will discuss in detail how to implement and synthesize these systems on three platforms: Digital Signal Processors, FPGA and ASICS using Verilog Hardware Description Language (HDL), and Graphics Processing Units (GPUs).

For Digital Signal Processors, student will be exposed to implementation models for DSP systems using C. For ASIC and FPGA platforms, we 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. Finally, we will focus on programmable platforms such as Graphics Processing Units and multicore systems where we will introduce and practice the fundamental programing concepts and challenges for such emerging hardware.

Textbooks and other Required Reading Materials

Textbook:

- M. D. Ciletti, *Advanced Digital Design With the Verilog HDL*, (2nd Edition) (required)
- E. Lee and S. Seshia, *Introduction to Embedded Systems, A Cyber-Physical Systems Approach*, (2nd Edition) (required)
- D. Thomas and P. Moorby, *The Verilog Hardware Description Language*, Springer, 2008. (optional)
- Jan M. Rabaey, Anatha Chandrakasan, Borivoje Nikolic, *Digital Integreated Circuits: A Design Perspective*, 2nd edition. Prentice Hall, 2003. (optional)
- Cameron Newham, *Learning the bash Shell: Unix Shell Programming*, O'Reilly Media. (optional)

Grading Guideline:

- (5%) Quizzes
- (5%) Assignments
- (15%) Project 1: Embedded system simulation using C
- (20%) Project 2: System modeling and synthesis using Verilog
- (25%) Final project, report, and oral presentation
- (10%) In-class paper presentation
- (10%) GPU assignments
- (10%) Final exam

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

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.

Looking forward to a mutually enjoyable semester!!