CSCE 5373/4373: Electronic Design Automation

Catalog Description:

This course studies physical design, analysis and optimization of VLSI circuits and systems with emphasis on computational realizations and optimization. We start with some related topics such as graph algorithms and discuss various well-known algorithms and methodologies in the design process of VLSI circuits, including design partitioning, logic synthesis, floorplanning, routing, static timing analysis and performance-driven layout. It requires a basic knowledge of digital circuit design, data structure, and object-oriented programming.

Prerequisites:

- CSCE 3953 System Synthesis and Modeling (C or higher) **and** CSCE 3193 Programming Paradigms (C or higher)

Textbook:

There is no required textbook for this course. Course notes for all lectures will be used. However, the following books are recommended:

- Practical Problems in VLSI Physical Design Automation, Sung Kyu Lim, Springer, 2008, ISBN 978-1402066269
- VLSI Physical Design Automation: Theory and Practice, Sadiq M. Sait and Habib Youssef, World Scientific, 1999, ISBN 978-9810238834

Goals:

The objective of this course is to study algorithms and methodologies to solve practical problems in computer-aided VLSI design. Students will also learn to compare the complexity and efficiency of various algorithms for physical design, analysis and optimization, and can implement such algorithms with a programing practice.

Student Learning Outcomes. By the end of this course, students will be able to:

- Write their own computer aided design tools
- Implement common algorithms to design circuit and systems
- Transform a circuit from a structural to a gate-level representation.
- Create a tool for circuit partitioning and floorplanning
- Understand how to perform placement and routing
- Analyze the complexity of design automation algorithms

Topics Covered:

- Introduction to computer-aided design (1 week)
- Design partition (3 weeks)
- Floorplanning (3 weeks)

- Placement (3 weeks)
- Routing (3 weeks)
- Static timing analysis (1 weeks)
- Interconnect optimization (1 weeks)

Homework Assignments and Project:

Homework will be assigned.
CSCE4373: For the final project, student will need to implement an algorithm introduced in the class and demonstrate the flow with a command line interface.
CSCE5373: For the final project, student will need to implement a working CAD tool based on multiple algorithms introduced in the class and demonstrate the design flow with graphic user interface and visualization of the output.

Grading:

Attendance and Participation: 10%
Homework or Lab Assignments: 30%
Midterm Exams: 30%
Final Project: 30%

We will use the following scale to assign final grades:

A: [90, 100] B: [80, 90), C: [70, 80), D: [60, 70), F: below 60%

Academic Dishonesty Policy

As a core part of its mission, the University of Arkansas provides students with the opportunity to further their educational goals through programs of study and research in an environment that promotes freedom of inquiry and academic responsibility. Accomplishing this mission is only possible when intellectual honesty and individual integrity prevail. Each University of Arkansas student is required to be familiar with and abide by the University's 'Academic Integrity Policy' at honesty.uark.edu. Students with questions about how these policies apply to a particular course or assignment should immediately contact their instructor.

Class/laboratory schedule:

Meets either 3 times a week for 50 minutes or 2 times a week for 75 minutes for 15 weeks.

Relationship of course to ABET Computer Engineering Student Outcomes:

- CE1. An ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics.
- CE6. An ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions.

Relationship of course to ABET Computer Science Student Outcomes:

- CS1. An ability to analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions.
- CS6. An ability to apply computer science theory and software development fundamentals to produce computing-based solutions.

Relationship of course to ABET Computer Science Topics:

- T4. Substantial coverage of algorithms and complexity, computer science theory, concepts of programming languages, and software development.
- T11. A major project that requires integration and application of knowledge and skills acquired in earlier course work.

Prepared by: Yarui Peng **Date**: 08/07/2020