KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
COLLEGE OF COMPUTER SCIENCES & ENGINEERING

COMPUTER ENGINEERING DEPARTMENT

COE 561: Digital System Design and Synthesis
Syllabus - Term 051

Catalog Description

Design representations, levels of abstraction & domains, Digital system design methodologies, Hardware Description Languages (HDLs), Modeling of Digital Systems using HDLs, High Level Synthesis – Internal representation (CDFG), Scheduling, Allocation & Binding, Controller and Data Path synthesis, Logic Synthesis – Two-level & Multi-level logic synthesis, Sequential logic synthesis (FSM synthesis), Technology Mapping- Library binding approaches, some case studies. The course emphasizes hands on experience through the use of available synthesis tools.

Pre-requisite : COE 308 or Equivalent

Instructor Dr. Aiman H. El-Maleh Room: 22/318 Phone: 2811
Email: aimane@ccse.kfupm.edu.sa

Text Book

Synthesis and Optimization of Digital Circuits – Giovanni De Micheli, McGraw Hill International edition, ISBN –0-07-113271-6, 1994.

Office Hours

SUMTW 11:00-12:00 (or by appointment)

References

Books

- Logic synthesis & verification algorithms – Gary D. Hachtel, Fabio Somenzi, Kluwer Academic Publishers; ISBN: 0792397460, 1996.
- Logic synthesis and verification, S. Hassoun and T. Sasao, Kluwer Academic Publishers, 2002.
- Logic Synthesis Using Synopsys – Pran Kurup, Taher Abbasi, Second Edition, Kluwer Academic Publishers, 1996.
- VHDL: Analysis and Modeling of Digital Systems, Navabi, McGraw-Hill, Inc., 2nd edition, 1998.

Journals

- IEEE Transactions on CAD
- IEEE Transactions on VLSI Design
- IEEE Transactions on Computers

Conference Proceedings

- Design Automation Conference (DAC)
- International Conference on Computer Aided Design (ICCAD)
- Design Automation and Test in Europe (DATE)
- International Conference on Computer Design (ICCD)

Tools

We will be using the following tools in this course: SIS package, Modelsim, Synposys synthesis tools, and Xilinx tools.

Grading Policy

Assignments	15%	
Paper Presentations	10%	
Implementation Project	20%	
Exam I	15%	**(Thursday, Oct. 20, 1:00 PM)**
Exam II	20%	**(Thursday, Dec. 22, 1:00 PM)**
Final	20%	

- Late assignments will be accepted (upto 3 days) but you will be penalized 10% per each late day.
- A student caught cheating in any of the assignments will get 0 out of 20%.
- No makeup will be made for missing Quizzes or Exams.

Detailed Syllabus

- **INTRODUCTION** **(0.5 week)**

  - Microelectronics, semiconductor technologies, microelectronic design styles, design representations, levels of abstraction & domains, Y-chart, system synthesis and optimization, issues in system synthesis.

- **MODELING OF DIGITAL SYSTEMS** **(1.5 week)**
  - Introduction to Hardware description languages (HDLs). Hardware Description and design using VHDL. Basic modeling concepts, Language elements, Behavioral modeling, Dataflow modeling, Structural modeling, some hardware modeling examples.

- **LOGIC SYNTHESIS (9 weeks)**

  - **Introduction to logic synthesis (1 week)**
    - Boolean functions representation, Binary Decision Diagrams, Satisfiability and Cover problems
  - **Two-level logic synthesis and optimization (2 weeks)**
    - Logic minimization principles, Exact logic minimization, Heuristic logic minimization, The Espresso minimizer, Testability properties of two-level circuits
  - **Multi-level logic synthesis and optimization (3 weeks)**
    - Models and transformations of combinational networks: elimination, decomposition, extraction. The algebraic model: algebraic divisors, kernel set computation, algebraic extraction and decomposition. The Boolean model: Don't care conditions and their computations, input controllability and output observability don't care sets, Boolean simplification and substitution. Optimization based on redundancy addition and removal. Testability properties of multilevel circuits. Synthesis of minimal delay circuits. Rule-based systems for logic optimization.
  - **Sequential Logic Synthesis (2 weeks)**
    - Introduction to FSM Networks, Finite state minimization, state encoding: state encoding for two-level circuits, state encoding for multilevel circuits, Finite state machine decomposition, Retiming, and Testability consideration for synchronous sequential circuits.

  - **Technology Mapping (1 week)**
    - Problem formulation and analysis, Library binding approaches: Structural matching, Boolean matching, Covering & Rule based approach.

- **HIGH LEVEL SYNTHESIS (4 weeks)**

  - **Design representation and transformations (0.5 week)**
    - Design flow in high level synthesis, HDL compilation, internal representation (CDFG), data flow and control sequencing graphs, data-flow based transformations.
  - **Architectural Synthesis (1 week)**
    - Circuit specifications: resources and constraints, scheduling, binding, area and performance optimization, datapath synthesis, control unit synthesis.
  - **Scheduling & Allocation (2.5 weeks)**

- **Unconstrained scheduling**: ASAP scheduling, Latency-constrained scheduling: ALAP scheduling, time-constrained scheduling, resource constrained scheduling, Heuristic scheduling algorithms: List scheduling, force-directed scheduling. **(1.5 week)**
- **Allocation and Binding**: resource sharing, register sharing, multi-port memory binding, bus sharing and binding, unconstrained minimum-performance-constrained binding, concurrent binding and scheduling. **(1 week)**

Paper Presentation Guidelines

- You need to select two recent papers (published within the last three years) related to the course topics.
- One paper must be a Journal paper and the other one a Conference paper.
- The two papers could be on different topics.
- You need to get my approval when you select the papers to be presented.
- Each paper presentation has a weight of 5% and you will be evaluated based on your ability to comprehend the paper and present it to the class.

	Paper Selection	**Paper Presentation**
Paper#1	**Oct. 25, 2005**	**Nov. 17, 2005**
Paper#2	**Jan. 1, 2005**	**Jan. 21, 2006**

Implementation Project Guidelines

- This project involves writing programs for implementing certain algorithms related to the course topics.
- Knowledge of C language is necessary.
- The projects will be assigned by the instructor to the students.
- The student will write a report documenting the results of his work in the project.