CSEE E6861y Tentative Syllabus

WEEK #1: January 21

Introduction.

Course overview, modern computer-aided digital design, Boolean representations.

Exact 2-Level Logic Minimization.

The Quine-McCluskey method (basics).

Heuristic 2-Level Logic Minimization.

Introduction to the "espresso" algorithm. Basic Boolean representation and terminology.

WEEK #2: January 28

Heuristic 2-Level Logic Minimization.

Multiple-output functions. Local search, iterative improvement algorithms.
Overview of key espresso steps: *expand, irredundant, reduce.*
Details of expand: cube order, expansion direction.

WEEK #3: February 4

Heuristic 2-Level Logic Minimization.

Details of irredundant and reduce. The Unate Recursive Paradigm.
Cofactoring. Fast tautology checking (introduction): basic termination rules.

WEEK #4: February 11

Heuristic 2-Level Logic Minimization.

Shannon decomposition: recursive divide-and-conquer.
Advanced techniques: exploiting properties of unate functions and covers.
Fast tautology checking (conclusion): final flow, advanced termination rules.
The containment problem. Fast complementation method.

WEEK #5: February 18

Heuristic 2-Level Logic Minimization.

Generating all essentials without generating all primes.
Final iterative espresso loop.
Advanced optimizations: "make-sparse"; avoiding local minima with "last-gasp"/"super-gasp".

Multi-Level Optimization: Algebraic Techniques.

Motivation. Circuit modelling: logic network graphs. Approximate cost models: area, delay.
Overview of logic transforms: extraction, substitution, collapse, simplify, decomposition.

WEEK #6: February 25

Multi-Level Optimization: Algebraic Techniques.

Introduction to the UC Berkeley "SIS" CAD tool environment. Designer scripts.
Algebraic vs. Boolean models. Network collapse with fast eliminate.
Algebraic division; substitution.
Foundations: kernels, co-kernels. Brayton/McMullen's Fundamental Theorem.
Optimal single- and multi-cube extraction.

WEEK #7: March 3

Multi-Level Optimization: Algebraic Techniques.

Rudell's rectangle covering formulation. Optimal logic decomposition.

Technology Mapping: Basics.

Optimal mapping of abstract gates to VLSI library cells.
VLSI cell libraries, cell characterization. Exact vs. heuristic solutions.
Overview of heuristic tech map: decomposition, partitioning, matching/covering.

WEEK #8: March 10

Technology Mapping: Basics.

Subject and pattern graphs.

Tree-based matching and covering algorithms. Introduction to dynamic programming.

Targeting different cost functions: area, delay, power. Inverter-pair heuristic.

WEEK OF MARCH 14–MARCH 18: SPRING BREAK.

WEEK #9: March 24

Technology Mapping: Advanced.

Delay-oriented mapping (load-independent): dynamic programming with specified arrival times.

Delay-oriented mapping (load-dependent): load binning, modeling drive strengths and capacitive loads.

Power-oriented mapping: using stochastic activity models, area/delay tradeoffs.

WEEK #10: March 31

Physical Design Basics: Partitioning and Placement/Routing.

Partitioning of large-scale circuits: Kernighan-Lin divide-and-conquer method.

Place-and-route: problem formulation, introduction to simulated annealing techniques.

WEEK #11: April 7

System-Level Optimization: Retiming.

Optimizing system area and clock cycle time by repositioning registers.

Graph-based models. Case study: correlator example.

Leiserson/Saxe's method. Bellman-Ford algorithm, linear programming solutions.

WEEK #12: April 14

Architectural Synthesis: Register-Transfer Level Design.

Specifying systems using control-dataflow graphs (CDFG's).

Overview of steps: scheduling, resource allocation and binding.

Architectural Synthesis: CAD Optimization Techniques.

System-level cost tradeoffs: latency, area and power.

Detailed case study: square root approximation.

Optimal scheduling algorithms: resource-constrained, time-constrained, force-directed.

WEEK #13: April 21

Architectural Synthesis: CAD Optimization Techniques.

Optimal resource sharing: registers, function units, buses.

Asynchronous Design: Hazard-Free Logic Minimization.

Introduction to modern asynchronous design. Hazards and hazard-free logic.

Exact hazard-free 2-level logic minimization. Safe multi-level "hazard-non-increasing" transformations.

WEEK #14: April 28

Advanced Topics (1): Technology Mapping for FPGA's.

Optimal mapping techniques for LUT-based FPGA's. Cong's DAG-Map and Flow-Map.

Advanced Topics (2): Approximate Computing.

Recent approaches using approximate arithmetic with limited errors for low power.

Advanced Topics (3): Design Automation for Synthetic Biological Systems.

Recent advances in "bio-design automation" to synthesize or re-engineer biological systems.

FINAL EXAM: Thursday, May 12, 4:10-7:00pm (location TBA)