EE 541: Automated Layout of Integrated Circuits

Professor Scott Hauck
EE-307Q, hauck@ee.washington.edu
Office Hours: by appointment (email schedule)

Book:
Kahng, Lienig, Markov, Hu,
VLSI Physical Design: From Graph Partitioning to Timing Closure,
2011, Springer.

Grading:
35% - Programming assignments
25% - Written homeworks
30% - Final Exam
10% - Class Participation

Prerequisites

C/C++ or Java Programming (CSE 373, CSE 326, or equivalent)
Ability to implement complex algorithms

Data Structures (CSE 373, CSE 326, or equivalent)
Linked Lists, Graphs

Basic Logic Design and Boolean Algebra (EE 271 or equivalent)
AND, OR, NAND, NOR gates
Boolean Algebra
Karnaugh Maps

We will provide background in Computational Complexity, VLSI, chip fabrication.
Joint Work Policy

Unless otherwise indicated, assignments and final projects must be done individually.
Students may not collaborate with each other on the specifics of homework or projects.

OK:
- Studying together for exams
- Discussing lectures or readings
- Talking about general approaches
- Help in debugging, CAD tools peculiarities, etc.

Not OK:
- Developing an algorithm/program together
- Writing code/doing design together
- Checking homework answers with each other

Violation of these rules is at minimum grounds for failing the class

VLSI: Very Large-Scale Integration
CAD & Physical Design

CAD = Computer Aided Design

Complexity of today’s circuits requires computer support for most design tasks

CAD split into Synthesis, Physical Design

Synthesis = translating designer requirements into a circuit graph

PD = translating circuit graph into layout (“blueprint”) for fabrication

Why Physical Design? CAD Developers

Rapidly developing field with many “classic” algorithms
   Fiduccia-Mattheyses, Simulated Annealing, …

Very inter-related process
   Good placement eases routing, better routers allow easier placement

Class Goals:
   Give basic background in overall flow & important classical algorithms
   Develop understanding of overall process
   Provide background for further learning
      ICCAD, DAC, ISPD, TCAD, TVLSI, …
Why Physical Design? VLSI Designers

Most chip design highly automated
   Chip complexity in the billions of transistors on a chip

Physical Design is the “compiler” for designs
   Understanding how specification becomes circuits guides logic design
   Understanding errors/problems/restrictions important for design

Why Not Physical Design?

This class does NOT teach the following:
   CMOS Design (EE 476, EE 525, EE 526)
   Transistor Physics (EE 331, EE 482, EE 531)
   Fabrication Techniques (EE 486, EE 539)

This class is NOT a general introduction to VLSI/CAD
   Students broadening into CMOS should take CMOS Design

This course requires a mature understanding of programming concepts
   You will develop your own complete standard cell layout system
      Partitioning, floorplanning, placement, global routing, detailed routing
   “Aphyds” system will provide skeleton within which you will write your code
**Partitioning**

Circuits can exceed chip capacity
Split circuits into chip-sized subcircuits
Meet capacity constraints
Reduce interconnect demand
Meet performance requirements

**Floorplanning**

Assign portions of a design to regions of the chip area
Blocks have adjustable sizes
Seek to reduce routing delay & area

**Control**

FP
Reg
Mult
Cache
ALU

**Control**

FP
Reg
Mult
Cache
ALU
**Placement**

Pick relative location for each gate

Seek to improve routeability, limit delay, reduce overall area

- NAND
- AOI

- DFF
- INV
- NOR

- DFF
- NOR
- DFF

**Global Routing**

Determine overall path of all routes

Pick channels to route through

Seeks to reduce delay, channel widths

- NAND
- AOI

- DFF
- INV
- NOR

- DFF
- NOR
- DFF
**Detailed Routing**

Determine exactly how each signal is routed through each region

Seeks to reduce routing area

- NAND
- AOI
- DFF
- INV
- NOR
- DFF

**Compaction**

Squeeze layout to reduce chip area

Helps eliminate inefficiencies caused by other steps

- NAND
- AOI
- DFF
- INV
- NOR
- DFF