EE371

Advanced VLSI Design

Jason Stinson

Intel Corporation
jstinson@stanford.edu

Class Overview

This class builds on EE313 and EE271 to look at the circuit design issues in large digital VLSI chips. At the core of this class is the job of 'circuit design' and the tasks that a circuit designer does in the industry. As we will see, much of the current effort is directed at dealing with the wires, and various signal integrity issues.

Like EE271, we will look at large functional blocks; like EE313 we will use SPICE a lot. Since simulating a large functional block in SPICE is not fun, much of the class will be devoted to building up ways of evaluating a circuit without having to simulate the whole thing. This skill is really the art of circuit design.

Assumptions for this course include both EE271 and EE313 or equivalents. You should be familiar with basic VLSI digital design and SRAM design.

Class Readings

The lecture notes will serve as the principle reference material that will be used in the class. I am not familiar with a good comprehensive text on circuit design and past sessions of this course have not been taught from a textbook. Although the notes will cover the material in the class, they will not be as complete as a textbook.

To supplement these notes, Mark has graciously provided some key papers on circuit design issues. Additionally, we will often refer to “The Design of High-Performance Microprocessor Circuits.” It is a collection of papers that cover much of the material in the class and we will have a number of readings from this book. You will be responsible for the material in these articles, even if we don’t completely cover them in class. It would be a good idea to read the papers before coming to class and asking questions about the material you don’t understand.

To provide additional info and/or alternate explanations of the material in the notes, references to readings from other textbooks will be included in the notes. While these readings are not required, they are often helpful in understanding the material.

Course Goals

The main focus of this course is to build on your circuit design knowledge and introduce common concepts used in designing large digitial systems.

Additionally, this course will help provide a base set of circuit skills that can be applied to new problems and help you make the right tradeoffs in your own designs. This field is very dynamic and new issues/constraints are introduced every time you sit down to design a new circuit—this class should give you some of the tools needed to solve those problems.

Other Stuff

1. This is my first time teaching this course. I will likely make mistakes, sometimes in the material and even more often in class logistics. Please be patient and don't hesitate to let me know if you think I've said something that wasn't correct.
2. We will be relying heavily on the notes and work from Mark Horowitz's past years. Over 90% of the lecture notes are taken from his prior work and are very gratefully acknowledged and recognized.

Direct from Mark's notes last year:

- People lie. Sometimes intentionally, sometimes by accident, and sometimes it is because what you heard is not what they meant. Never take anything on blind faith. Work it out—make sure it works. Many clever circuits that are published either don't work or are very sensitive to certain conditions. Be careful.
- There are no RIGHT answers, and there are no PERFECT circuits. Everything has its warts. A good circuit simply has the right set of warts to meet the constrains of the problem.
- Simulation is no substitute for thinking. SPICE can make your job much easier, but it also can make it much harder. Like all tools it helps only if you use it well. And that requires thinking.

Class Topics

The first part of the class will be focused on learning the fundamental tools:

- What to model and what to verify
- Device model, MOS and wires
  - Assume everyone has a background in MOS models from EE313
  - Add information about matching issues and wires
- Worst-case design, simulation methods, and margin testing

Next we'll take a look at one of the most basic large structures in circuit design:

- Adder algorithms and Implementation

After that, we will look at circuit electrical environments:

- Timing – clocking methods and elements
- Electrical – Power/ground signal

And we'll finish the second half looking at larger/complex structures and issues:

- Multipliers, I/O, signaling and clock recovery
- Hi-speed circuits
- Test and Debug

Logistics

- Please come to class.
  - If no one comes to class, then no one asks questions…and no one learns anything.
  - And if no one comes to class, Mark will show me how to turn off the in-campus broadcast ☺
- There is a review session for the class
  - Times will be announced (most likely on Friday).
  - It will be televised and on the internet. It will explain the homework in more detail and answer any other questions that you have. It will also go over the tools that we will be using for the class (Sue, HSPICE)

Homework

There will be 3-4 week problem sets and a final design project.

Homework:

- Handed out on Thursday, due the following Thursday in class. No late assignments.
- Homework will be to get experience with the technology and simulation and to experiment with new design techniques. There will usually be one more open-ended design problem on each problem set.

HSPICE

The problem sets will use a version of SPICE called HSPICE. This version has a number of features (like parameter sweeps and optimization) that will make your life easier. If you do not have access to HSPICE at your location please let me know…you will need to get approval from me to use another simulator.

There is a library provided for this class:

-0.35u CMOS technology, including ‘corner’ models

-Wire models are also provided

Some of the assignments will require you to estimate parameters for a scaled technology

One of the review sessions will explain how to use the key features in HSPICE. I assume you have all used it in EE313. Unfortunately, I am not familiar with HSPICE so I will be able to help you with basic SPICE questions, but will always refer you to our TAs for specific HSPICE issues.

Honor Code

Please remember you are bound by the honor code:

- I will trust you not to cheat
- I will try not to tempt you

But if you are found cheating it is very serious

- There is a formal hearing
- You can be thrown out of Stanford

Save yourself and me a huge hassle and be honest

Lecture 1:

Models, Simulation, and Circuit Design

Computer Systems Laboratory

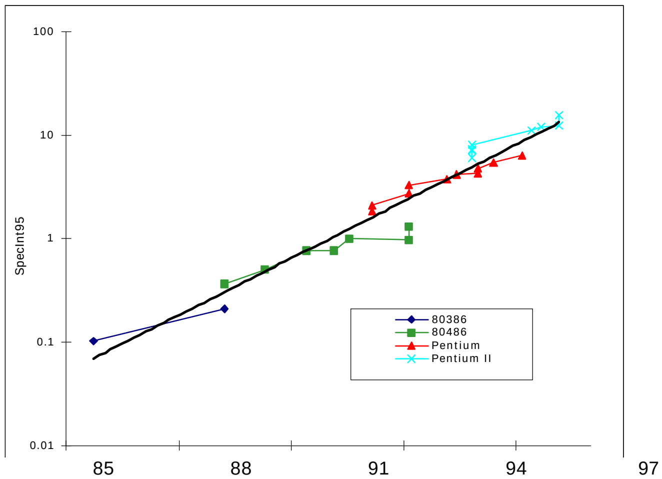
Stanford University

horowitz@stanford.edu

Integrated Circuit Trends

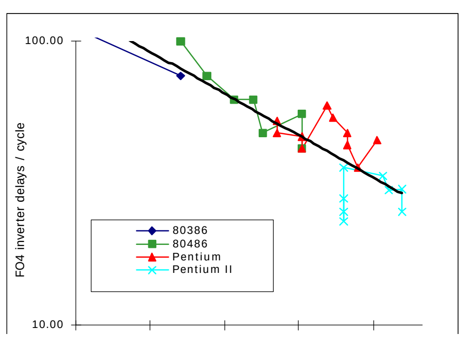
Two trends have made circuit designers in high demand:

- Process scaling and demands for high-clock rate processors



What These Trends Mean

The cycle time of digital systems is falling faster than the speed of the gates



- Need to build functions and wires that are faster (in terms of speed of the basic building blocks) than last time

Long wires are becoming wet noodles

- Pushing information through them is getting harder
- Need to worry about wire delay and noise coupling issues

Faster Functions

There are two approaches to building faster functions:

- Build faster logic gates
  - Here you use some experience with transistors to craft faster basic gates
  - Dynamic logic, and certain passgate logic is the best example of this
  - Customize the layout to make the critical wires short
- Build faster architectures
  - Leverage all those additional transistors that are possible
  - Do more things in parallel, do some speculative calculations
  - Decrease the time for the high-level function, by using many standard gates

Today, the faster architecture approach is ALWAYS used, so digital circuit designers must deal with environments where there are a large number of gates and wires to deal with.

Example: Adder

If you want to build a fast 64 bit adder:

- Look over the different architectures proposed
  - Ripple, carry bypass, carry select,
  - Tree adders
  - Ling adder (reformulation of logic equations)
- Look over the circuit issues
  - Static CMOS
  - Pass transistor logic
  - Dynamic Logic
  - Number of stages of logic
  - Clever implementation of individual logic gates, or groups of logic gates

Circuit Design

Most people think about:

- Innovative configurations of transistors that perform some function better.
  - Where better might be smaller, faster, lower power, etc.

That is part of the job. The part that takes more time is:

- Making sure that this collection of transistors will work

OR

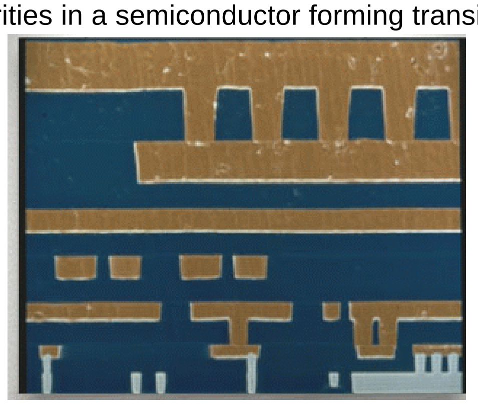
- Figuring out why this collection of transistors does not work, or only works on a few parts.

To do either, you need to be able to reason about circuits ...

The Problem

ICs are very complicated, contain many millions of tiny 3D structures

- Interleaving of conductors and insulators
- Diffusions of impurities in a semiconductor forming transistors



Could talk about the system in terms of 3D electric fields, and carriers

- But takes a long time for a computer to simulate a single transistor

The Solution: Models

Need to use some kind of model to simulate anything

- In the 3-D simulation has some models for how the electrons behave
  - This relies on other models for carriers in crystal structures
  - Maxwell's equations
- Simulating complex designs mean that the base models must account for more effects in each element you 'model'
  - Simulate transistors, not carriers
  - Assume wires are equipotentials (quastatic models)

To reason about a system (either you or a machine) one needs a model of it.

Models

Are an approximation of the real world

- Must leave many details out
- Must (to be useful) retain the important 'details'
- Appropriate level depends on questions you want to answer

CAUTION:

- Simulation and analysis do not tell you what the circuit will do
- It tells you what your MODEL of the circuit will do
- So remember:

  Garbage in, garbage out

Some of the hardest work is figuring out the right model for a problem

Modeling / Simulation Problem

There are really two problems:

- Need to generate the correct model of the circuit
- Need to stimulate that circuit in ways that exercise the problem

  Add coupling noise the critical time

  Set initial conditions for the worst-case charge-sharing

  Inject substrate noise

SPICE limitation:

- Only evaluates the model of the circuit that you gave it
- Does the evaluation for the conditions you specify

  Answers the question you ask,

  But does not tell you whether it was the right question

What Model - The SPICE Approximation

Use lumped element model:

- Quasi-static (size is small compared to wavelength/4)
- Approximate devices by terminal characteristics
- Connections in the model are equipotentials

Major equation:

- $\left(\mathbf{i} = \mathbf{C} \cdot \frac{\Delta \mathbf{V}}{\Delta \mathbf{t}}\right) \rightarrow \Delta \mathbf{t} = \frac{\mathbf{C}\Delta \mathbf{V}}{\mathbf{i}}$ or $\Delta \mathbf{t} = \frac{\mathbf{Q}}{\mathbf{i}}$ charge control model

- So I need to model devices on the chip by their terminal iV and CV behavior - Figuring out the right model can be hard.

  (Sometimes need inductor equation too)

What Needs to be Modeled?

Transistors

- nMOS, pMOS

Wires

- They are not ideal connectors
- How complex?

    Resistance effects, iR drops in lines?

    Coupling, Inductance?

Circuit Environment

- Temperature, Power Supply, Substrate Voltage, Chip ‘Gnd’ vs. Board ‘Gnd’
    - Modeling of the package and power and ground distribution nets
    - Modeling of heat generation and flow

Appropriate model depends on question being asked