CMOS VLSI Design Techniques - Textbook DRAFT

Alice C. Parker             Fall 2015

Forward

CMOS circuit design is all about moving and storing charge. Charge represents data and the flow of charge represents change in circuit state. For good design, we want to move as little charge as possible, minimizing power consumption, and we want to move the charge as quickly as possible, maximizing performance. We want to prevent charge from leaking away or finding a path to ground when it is used to represent data or to control the state of the circuit. Conversely, we want to store charge or move charge to ground as quickly as possible to change the data stored or change the state of the circuit.

Physical CMOS VLSI designs are constructed using conductive materials and semiconducting materials that are arranged to form field-effect transistors. Physical design of CMOS circuits involves determining the physical dimensions and position of each transistor, and the location and sizing of each conductor connecting the transistors. The choice of transistor sizing and the interconnection sizing and location determine the ideal circuit properties of the design. However, VLSI design is never that straightforward, and complications fill the pages of most VLSI textbooks. The complications involve interactions between different connected logic gates at the circuit level, interactions between circuit elements not directly connected due to capacitive coupling or relative positions, and changes in device properties when electric fields are applied. Also fabricated circuits have properties that differ from chip to chip, from fabrication run to fabrication run, and from one corner of the chip to another. These complications, taken together, result in designs with individual attributes (size, power, performance) that are not easily optimized without sacrificing other attributes. Attempts to lower power alter performance, and attempts to achieve higher performance usually carry size and power penalties. Power, size and performance must be traded off to meet constraints while producing a marketable product. Methodical top-down design is difficult to achieve. Many processes involved in design are bottom-up, and the relationship between the logic and the physical design is not isomorphic. We will return to this complication as a theme throughout the text. The final complication results from the rapid evolution of CMOS technology as devices and circuits are scaled to smaller and smaller dimensions, and innovations such as high-K dielectrics and FINFETs are introduced.

The first complication, circuit interactions, is not apparent at the logic level, when gates are connected to form more complex functions. Changing the transistor sizes in a NAND gate, for example, does not affect the logic function of the gate. However, the logic gates supplying inputs to the NAND gate experience changes in performance and power consumption when the NAND gate devices are resized, and the logic gates driven by the NAND gate output are likewise affected by NAND device resizing. Often, an entire block of logic, from inputs to outputs, must be
analyzed, simulated and evaluated when a single device size is changed. A connected circuit is formed between a driving gate and the gates it drives, and design decisions must include properties of both gates concurrently.

The second complication is the fact that the materials forming integrated circuits are not ideal. Resistance and capacitance, along with other parasitic effects like inductance, alter the ideal circuits we design and simulate, and introduce complexities to the behavior of the circuits that varies depending on how the circuits are arranged on the chip itself, and how they are interconnected. The interaction between unconnected components must be taken into account, with the coupling capacitance between components being the most prominent. Silicon chip real estate is as expensive as the real estate in Silicon Valley. Every square micron is valuable, and design rules, like building codes, tell us how close components can be to each other as well as minimum sizes of components to achieve an acceptable likelihood that the circuits will operate as planned. However, just as there is no perfect isolation from noisy neighbors, the flow and storage of charge affect neighboring components, and physical structures affect nearby physical structures due to the inexact nature of fabrication. Good design practice involves taking unavoidable interactions into account, while minimizing silicon real estate.

Understanding the interactions involves knowledge about the locations of components in 3-dimensional space. Background understanding of how CMOS integrated circuits are fabricated provides understanding of design rules and potential component interactions, as well as the cost (in terms of power, delays and size) of complicated interconnection structures that occupy multiple layers in the integrated circuit.

One of the trickiest complications to good VLSI design is the situation that CMOS FETs change device properties as electric fields are applied. While some of these changes are subtle, and can be ignored to gain a basic understanding of circuit operation, some have major effects on how the transistors behave, whether as an open switch, voltage-controlled current source, or voltage-controlled resistance. Any analysis of CMOS circuits obviously must incorporate device models that represent these effects. Moving charges quickly and at low energy cost depends on how the transistors behave.

Although basic physical laws hold true over time, nothing is certain in VLSI design. Achieving an optimal design is an illusion due to many variations that can occur as a result of imperfect fabrication and subtle changes in materials, contaminants, and processing. The best designs are designs that are centered, that operate correctly and meet specifications regardless of expected variations in supply voltage, temperature, and device characteristics. Designs that are uncomplicated enough to be changed quickly as the deadline to begin fabrication nears (usually called tape-out) are more desirable than designs that are complicated in order to achieve peak performance and energy consumption or small die size, but cannot be modified quickly due to their complexity. Designs that are able to be incorporated into products in a timely fashion (to meet the *market window*) are more valuable than
designs that are highly optimized but are too late for products that have planned timing.

The final challenge is one that keeps the author interested in teaching what appears on the surface to be the same course over the years - there has been a rapid evolution in technology and, with the changes, simplistic approaches to design do not endure. Design techniques that are fundamental can transcend the design changes required to keep pace with technological evolution. The author has applied some of the design techniques to circuits simulated with nanodevice simulation models. While the ultimate circuits are sometimes quite different from their CMOS analogues, the fundamental approaches to design of FET digital (and some analog) circuits are universal.

The complexities of VLSI design have led to several textbooks over the decades that collectively provide students with practical design approaches, beginning with the revolutionary Mead and Conway textbook [ref], followed by notable texts, including the Weste and Eshraghian text [ref]. This textbook combines the best of the best design guides, along with the author’s own experience. The text is designed to provide the engineer with CMOS VLSI design approaches, and to assist in developing an intuitive sense of how CMOS VLSI circuits behave. Although detailed assessment of designs is only possible with CAD (Computer-Aided Design) software, it is impossible to investigate the impact of all possible design choices using software, even with massive supercomputers and cloud computing. The human designer must have an intuitive feeling for how the circuits she is designing behave, so that the software tools can be used in an effective manner. Intuition is gained by students using this text via the consistent use of equivalent circuits for connectors and transistors in solving design problems. Similarly, the use of RC time constants (resistance times capacitance) provides significant guidance when choosing between design alternatives that trade off performance, power and chip area.

In the process of producing good designs, there is no substitute for experience. Students inevitably express the need for a deadline extension to begin again now that the completed design reveals shortcomings. Anticipating such shortcomings and designing with an ability to predict outcomes is the product of experience, of experiencing the design flow again and again. Such prediction is the hallmark of a good designer. By accelerating the effects of experience with well-chosen design examples, this textbook guides the student to design maturity within the course of a single semester.

This book is a beginning for students, not an end. Entire texts address aspects of design not covered here, the detailed behavior of semiconductor devices, the physical arrangement of memories, the architecture of processors, systems on chip, networks on chip, arithmetic circuits, testing methodologies, and many other topics. The revolution in integrated circuit design education enabled by Carver Mead and Lynn Conway’s fundamental textbook has grown into an entire graduate degree program here at USC, and in other organizations. Mastery of this field requires tall, thin designers, as Carver Mead stated years ago, tall in the sense of bridging from
system design to semiconductor device design and the manner in which devices are fabricated in integrated circuits that become working systems. Even tall thin designers need to position themselves on firm ground, and this textbook provides the foundation to support designers as they stretch taller with each successive evolution of technology.

The organization of the text is as follows: In Chapter 1, brief review of digital logic is followed by presentation of both static and dynamic transistor circuits that implement logic functions and storage for VLSI implementation. An overview of transistor behavior, including the role of gate capacitance, is provided, and basic equations for approximating current flow in CMOS FETs are given. The use of equivalent circuits is introduced. The chapter concludes with a description of transistor behavior under changing electric fields in both the inverter and transmission gate.

Chapter 2 describes the physical structure of CMOS transistors, summarizes a simple set of design rules for VLSI layout and outlines the steps involved in MOS fabrication. Layout techniques are discussed, including the Euler Path method for compound gate layout. The chapter concludes with discussion of unintended behavior of MOS VLSI circuits encountered due to the arrangement of semiconductor materials along with environmental conditions (e.g. high-energy radiation from outer space), and discusses remedies for these problems. At this point the student is able to design simple logic and storage layouts that perform correctly.

The focus of Chapter 3 is performance, the speed at which changes in circuit state occur, whether in the logic or in storage. The first part of the chapter introduces parasitic capacitance computations, and overviews complexities introduced by modern VLSI structures, including fringing fields. The role of capacitance in performance of logic circuits is presented, and the notion of RC time constants is introduced. Lumped and distributed (Elmore) RC time constants are derived for example circuits, including the inverter, NAND and NOR gates, compound gates, transmission gates and an XOR gate. Integral models of delay computations are presented for the inverter and for connections. The chapter concludes with a discussion of the precise timing required in data storage circuits in order to achieve correct behavior under a variety of conditions. At this point in the text, the student engineer is able to design and lay out circuits that meet timing constraints.

Power consumption, the evil twin of performance, is the subject of Chapter 4. High performance circuits are plagued by high power consumption, so the two must be carefully traded off to get effective designs. Simple techniques for understanding the parameters that control power consumption are presented. Design techniques that minimize power consumption are included.

Chapter 5 describes the impact of device scaling on circuit, layout and system design. The "embarrassment of riches" resulting from the ability to fabricate silicon dies (chips) with a billion transistors brings with it the problems of large-scale
design. Solutions in the form of multiple cores interconnected via shared buses or networks are discussed. A section of the chapter presents nanodevices that are fabricated entirely differently from CMOS, and speculates on the future of such technologies. Finally, the pervasive nature of modern technology, and the proliferation of not only electronic hardware but also millions of lines of software code is discussed. Reliability and testability of designs are introduced, and consideration of process variations in design is covered. The text concludes with a discussion of engineering ethics, particularly with respect to VLSI design and corresponding software.

Appendix A contains a discussion of synthesis tools, particularly tools that synthesize chip layouts from designs at the logic level and register-transfer level.

Chapter 1: Introduction to VLSI Design

All electronic circuits, including CMOS VLSI circuits are analog. We simplify circuit operation so that certain voltage levels correspond to 1’s and 0’s but in actuality there are ranges of voltages representing logic values and much of the interesting behavior involves transitions between voltage levels. Analysis of VLSI circuits as digital logic yields a rapid, simplified understanding of the technology, but competitive physical design that is low power and high speed, with small devices that result in massive systems on chip, involves currents and voltages, conductance, capacitance and sometimes inductance. Ohm’s law, Kirchoff’s current and voltage laws, the differential equations describing the relationship between current and voltage in capacitors, and the empirical equations describing the relationships between current and voltage in semiconductor devices all must be applied to produce good VLSI designs.

Chapter 1 Introduction to VLSI Design

VLSI design bridges the design process from systems to fabrication of integrated circuits. Hence we will launch our study of VLSI design by considering a simple system, reducing the complexity so that the relationship between steps in the design process can be highlighted in a simple, compact example, a hypothetical system that follows a stylus (or fingertip) on a touch screen, and displays the cursor as the stylus moves. This example will allow us to explore design descriptions at different levels of detail, and with different properties. The design descriptions will each focus on a different aspect of the design, from the behavior to the logic design to the physical implementation - the structure of the integrated circuit. Each of these descriptions can be detailed or quite abstract. The behavior of an adder can be described by a simple arithmetic equation (or equations, if you want to include overflow/underflow checking). At a lower level, the behavior of an adder can be described by Boolean equations. Finally the behavior of an individual transistor can
be described by non-linear equations. We begin with abstract behavior that has implications for the resultant design, but no information about the physical design.

As a stylus or fingertip moves on a touch screen, the cursor follows the stylus position. This basic system involves using the stylus movement to send a new position of the cursor for the display. There is a basic assumption here that the stylus moves slowly enough that the system can react as quickly as the stylus moves. The touchscreen is 512x512 pixels, labeled 0,0 in the upper left, and 511,511 in the lower right.

Figure 1.1 Example 1: The Touchscreen Stylus and Cursor

**System Behavior Description of the Touchscreen Stylus and Cursor**

At the highest level, the system description has several variables representing stylus movement and cursor position.

Cursor positions:

- `cursor.x`
- `cursor.y`

Stylus movement:

- `stylus.left {0,1}` stylus has moved to the left when `stylus.left`=1
- `stylus.right {0,1}` stylus has moved to the right
- `stylus.up {0,1}` stylus has moved up the screen
- `stylus.down {0,1}` stylus has moved down the screen

We are assuming the system samples quickly enough so that there is either no motion, or motion in only one of four directions to an adjacent pixel.

We also have two signals, `go` and `reset`, that control the system.
A portion of the overall operation (behavior) of the hardware is described in pseudocode (not an actual language) as follows:

\[
\text{Do while go and not reset}
\]

\[
\text{if stylus.left = 1 and cursor.x /= 0}
\]
\[
cursor.x <- cursor.x - 1
\]

\[
\text{if stylus.right=1 and cursor.x /=511}
\]
\[
cursor.x = cursor.x + 1
\]

Similar code could be used to describe up and down motions to the top and bottom of the touchscreen.

**Detailed Behavior of the Touchscreen System using Boolean Algebra**

For the detailed behavior, we focus on a specific command to the system hardware to increment the x position of the cursor that is generated when the stylus has been moved to the right and the cursor is not already on the far right of the screen. We assume the cursor positions are 10-bit binary numbers labeled 0-9 from left to right. The leftmost bit is the sign bit.

\[\text{increment.x} = \text{stylus.right} * \left( / \{ [\text{cursor.x}_0*\text{cursor.x}_1*\text{cursor.x}_2*...*\text{cursor.x}_9] \} \right)\]

Of course, many other Boolean equations will be required to describe the entire system in this way.