Part 3 steps back from the purely digital world to focus on the critical analog support circuitry that is important to any viable computing system. These topics include basic DC and AC circuit analysis, diodes, transistors, op-amps, and data conversion techniques. The fundamental topics from the first three parts are tied together in Part 4 by discussing practical digital design issues, including clock distribution, power regulation, signal integrity, design for test, and circuit fabrication techniques. These chapters deal with nuts-and-bolts design issues that are rarely covered in formal electronics courses.

More detailed descriptions of each part and chapter are provided below.

PART 1 DIGITAL FUNDAMENTALS

The first part of this book provides a firm foundation in the concepts of digital logic and computer architecture. Logic is the basis of computers, and computers are intrinsically at the heart of digital systems. We begin with the basics: logic gates, integrated circuits, microprocessors, and computer architecture. This framework is supplemented by exploring closely related concepts such as memory and communications that are fundamental to any complete system. By the time you have completed Part 1, you will be familiar with exactly how a computer works from multiple perspectives: individual logic gates, major architectural building blocks, and the hardware/software interface. You will also have a running start in design by being able to thoughtfully identify and select specific off-the-shelf chips that can be incorporated into a working system. A multilevel perspective is critical to successful systems design, because a system architect must simultaneously consider high-level feature trade-offs and low-level implementation possibilities. Focusing on one and not the other will usually lead to a system that is either impractical (too expensive or complex) or one that is not really useful.

Chapter 1, "Digital Logic," introduces the fundamentals of Boolean logic, binary arithmetic, and flip-flops. Basic terminology and numerical representations that are used throughout digital systems design are presented as well. On completing this chapter, the awareness gained of specific logical building blocks will help provide a familiarity with supporting logic when reading about higher-level concepts in later chapters.

Chapter 2, "Integrated Circuits and the 7400 Logic Families," provides a general orientation to integrated circuits and commonly used logic ICs. This chapter is where the rubber meets the road and the basics of logic design become issues of practical implementation. Small design examples provide an idea of how various logic chips can be connected to create functional subsystems. Attention is paid to readily available components and understanding IC specifications, without which chips cannot be understood and used. The focus is on design with real off-the-shelf components rather than abstract representations on paper.

Chapter 3, "Basic Computer Architecture," cracks open the heart of digital systems by explaining how computers and microprocessors function. Basic concepts, including instruction sets, memory, address decoding, bus interfacing, DMA, and assembly language, are discussed to create a complete picture of what a computer is and the basic components that go into all computers. Questions are not left as exercises for the reader. Rather, each mechanism and process in a basic computer is discussed. This knowledge enables you to move ahead and explore the individual concepts in more depth while maintaining an overall system-level view of how everything fits together.

Chapter 4, "Memory," discusses this cornerstone of digital systems. With the conceptual understanding from Chapter 3 of what memory is and the functions that it serves, the discussion progresses to explain specific types of memory devices, how they work, and how they are applicable to different computing applications. Trade-offs of various memory technologies, including SRAM, DRAM, flash, and EPROM, are explored to convey an understanding of why each technology has its place in various systems.

Chapter 5, "Serial Communications," presents one of the most basic aspects of systems design: moving data from one system to another. Without data links, computers would be isolated islands. Communication is key to many applications, whether accessing the Internet or gathering data from a remote sensor. Topics including RS-232 interfaces, modems, and basic multinode networking are discussed with a focus on implementing real data links.

Chapter 6, "Instructive Microprocessors and Microcomputer Elements," walks through five examples of real microprocessors and microcontrollers. The devices presented are significant because of their trail-blazing roles in defining modern computing architecture, as exhibited by the fact that, decades later, they continue to turn up in new designs in one form or another. These devices are used as vehicles to explain a wide range of computing issues from register, memory, and bus architectures to interrupt vectoring and operating system privilege levels.

PART 2 ADVANCED DIGITAL SYSTEMS

Digital systems operate by acquiring data, manipulating that data, and then transferring the results as dictated by the application. Part 2 builds on the foundations of Part 1 by exploring the state of the art in microprocessor, memory, communications, and logic implementation technologies. To effectively conceive and implement such systems requires an understanding of what is possible, what is practical, and what tools and building blocks exist with which to get started. On completing Parts 1 and 2, you will have acquired a broad understanding of digital systems ranging from small microcontrollers to 32-bit microcomputer architecture and high-speed networking, and the logic design methodologies that underlie them all. You will have the ability to look at a digital system, whether pre-existing or conceptual, and break it into its component parts—the first step in solving a problem.

Chapter 7, "Advanced Microprocessor Concepts," discusses the key architectural topics behind modern 32- and 64-bit computing systems. Basic concepts including RISC/CISC, floating-point arithmetic, caching, virtual memory, pipelining, and DSP are presented from the perspective of what a digital hardware engineer needs to know to understand system-wide implications and design useful circuits. This chapter does not instruct the reader on how to build the fastest microprocessors, but it does explain how these devices operate and, more importantly, what system-level design considerations and resources are necessary to achieve a functioning system.

Chapter 8, "High-Performance Memory Technologies," presents the latest SDR/DDR SDRAM and SDR/DDR/QDR SSRAM devices, explains how they work and why they are useful in high-performance digital systems, and discusses the design implications of each. Memory is used by more than just microprocessors. Memory is essential to communications and data processing systems. Understanding the capabilities and trade-offs of such a central set of technologies is crucial to designing a practical system. Familiarity with all mainstream memory technologies is provided to enable a firm grasp of the applications best suited to each.

Chapter 9, "Networking," covers the broad field of digital communications from a digital hardware perspective. Network protocol layering is introduced to explain the various levels at which hardware and software interact in modern communication systems. Much of the hardware responsibility for networking lies at lower levels in moving bits onto and off of the communications medium. This chapter focuses on the low-level details of twisted-pair and fiber-optic media, transceiver technologies, 8B10B channel coding, and error detection with CRC and checksum logic. A brief presentation of Ethernet concludes the chapter to show how a real networking standard functions.

Chapter 10, "Logic Design and Finite State Machines," explains how to implement custom logic to make a fully functional system. Most systems use a substantial quantity of off-the-shelf logic products to solve the problem at hand, but almost all require some custom support logic. This chapter begins by presenting hardware description languages, and Verilog in particular, as an efficient