

MODERN VLSI DESIGN

System-on-Chip Design

Third Edition

Wayne Wolf



Prentice Hall PTR
Upper Saddle River, NJ 07458
www.phptr.com



TK7874.65

W65

2002

Library of Congress Cataloging-in-Publication Data

A CIP catalog record for this book can be obtained from the Library of Congress.

Editorial/production supervision: *Patti Guerrieri*
Acquisitions editor: *Bernard Goodwin*
Marketing manager: *Dan DePasquale*
Manufacturing manager: *Alexis R. Heydt-Long*
Editorial assistant: *Michelle Vincenti*
Cover design director: *Jerry Votta*
Cover designer: *Anthony Gemmellaro*



© 2002, 1998, 1994 by Prentice Hall PTR
Prentice-Hall, Inc.
Upper Saddle River, NJ 07458

Prentice Hall books are widely used by corporations and government agencies for training, marketing, and resale.

The publisher offers discounts on this book when ordered in bulk quantities. For more information, contact: Corporate Sales Department, Phone: 800-382-3419; Fax: 201-236-7141; E-mail: corpsales@prenhall.com; or write: Prentice Hall PTR, Corp. Sales Dept., One Lake Street, Upper Saddle River, NJ 07458.

Illustrated and typeset by the author. This book was typeset using FrameMaker. Illustrations were drawn using Adobe Illustrator, with layout plots generated by cif2ps.

Photo credits: Example 1-1: Intel Corporation; Figure 2-2: IBM; Figure 2-5: Agere; Figure 2-15: UMC; Figure 2-16: IBM; Figure 6-1: Hewlett-Packard Company; Example 7-1: Tom Way, IBM Microelectronics, Essex Junction, VT; Figure 7-19: Wayne Wolf; Figure 7-20: Wayne Wolf and IBM; Example 8-4: Philips.

All products or services mentioned in this book are the trademarks or service marks of their respective companies or organizations.

All rights reserved. No part of this book may be reproduced, in any form or by any means, without permission in writing from the publisher.

Second edition published under the title *Modern VLSI Design: Systems on Silicon*.

Printed in the United States of America
10 9 8 7 6 5

Fifth Printing

ISBN 0-13-061970-1

Pearson Education LTD.
Pearson Education Australia PTY, Limited
Pearson Education Singapore, Pte. Ltd.
Pearson Education North Asia Ltd.
Pearson Education Canada, Ltd.
Pearson Educación de México, S.A. de C.V.
Pearson Education — Japan
Pearson Education Malaysia, Pte. Ltd.

power-down mode;

- further modifying the control logic to ensure that the power-down and power-up operations do not corrupt the state of the powered-down section or the state of any other section of the machine.

The conditional clock for the power-down mode must be designed with all the caveats applied to any conditional clock—the conditioning must meet skew and edge slope requirements for the clocking system. Static or quasi-static memory elements must be used in the powered-down section for any state that must be preserved during power-down (it may be possible to regenerate some state after power-up in some situations). The power-down and power-up control operations must be devised with particular care. Not only must they put the powered-down section in the proper state, they must not generate any signals which cause the improper operation of other sections of the chip, for example by erroneously sending a *clear* signal to another unit. Power-down and power-up sequences must also be designed to keep transient current requirements to acceptable levels—in many cases, the system state must be modified over several cycles to avoid generating a large current spike.

8.6 Systems-on-Chips and Embedded CPUs

VLSI manufacturing can now easily put tens of millions of chips on a single die at reasonable cost. These manufacturing advances have enabled designers to create **systems-on-chips (SoCs)** that include all the components necessary for systems that only recently included multiple chips. A system-on-chip often contains a wide variety of elements: random logic, memory, CPUs, and analog circuitry. Systems-on-chips are challenging to design as much because of the diversity of elements that must be put on the chip as for the sheer size of the chips.

A system-on-chip is not useful unless it can be designed in a reasonable amount of time. If all the subsystems of an SoC had to be designed by hand, most SoCs would not be ready in time to make use of the manufacturing process for which they were designed. SoC design teams often make use of **intellectual property (IP) blocks** in order to improve their productivity. An IP block is a pre-designed component that can be used in a larger design. There are two major types of IP:

- **Hard IP** comes as a pre-designed layout. Because a full layout is available, the block's size, performance, and power consumption can be accurately measured.

- **Soft IP** comes as a synthesizable module in a hardware description language such as Verilog or VHDL. Soft IP can be more easily targeted to new technologies but it is harder to characterize and may not be as small or as fast as hard IP.

There are several challenges in IP-based design. The first is in designing **IP blocks** that are generally useful. It takes a considerable amount of work to fully test an IP block and document it so that it can be used by another designer; not all components on a chip are frequently-enough used to make it worthwhile to package them as IP. The second is choosing an interface standard for **IP blocks**. **IP blocks** must clearly be able to talk to each other and to custom logic in order to be useful; however, no one interface standard is best for all types of systems. The third challenge is ensuring that the IP block works properly in the SoC. The logical function, performance, power consumption, and testability of the IP block must all be verified in the context of the system.

One critical type of intellectual property for SoC design is the **embedded CPU**. An embedded processor can be programmed to perform certain functions on the chip, much as an embedded processor is used in a board design. Embedded CPUs have been used on chips for many years: early embedded processors were mostly 8-bit CPUs used for basic sequencing; today, powerful 32-bit CPUs can be embedded on a system-on-chip. The fact that not just the CPU but also its cache, main memory, and I/O devices can be integrated on the same chip make embedded processors especially attractive?

Embedded CPUs are increasingly popular on SoCs for several reasons. First, many sophisticated applications are best implemented in software. Multimedia applications like MP3 audio and MPEG video are examples of functions that are difficult to implement without some amount of embedded software. Second, many complex systems must run embedded software in order to implement their applications. For example, digital audio systems must run digital rights management software that is available only in binary form. Many systems-on-chips also use Linux, Windows CE, the Palm OS, or some other OS to provide file management and networking. Third, embedded CPUs help decrease design time. Because the embedded processor is a relatively well-understood component, the design of the software can be somewhat decoupled from the hardware design.

The next example describes a system-on-chip for multimedia applications that makes use of a sophisticated embedded CPU.

The system-on-chip includes a variety of I/O devices and accelerators as well as the VLIW CPU. The I/O devices are chosen to satisfy a variety of needs for multimedia systems. Audio and video input and output are clearly important; the I²C interface talks to an industry standard bus that is often used for chip-to-chip communication of low-rate control information. Accelerators are chosen to augment the VLIW CPU's capabilities and speed up critical operations, such as variable-length (Huffman) coding and descrambling of encrypted DVD bit streams.

There are several important issues in system-on-chip design:

- **I/O devices.** Systems-on-chips inherently require input and output devices. Some I/O devices may be available as **IP blocks**. Others may not be available from outside sources, either because the block does not command a high enough price to be worth selling or because it is so strategically important that the designers will not release it.
- **Mixed-signal design.** A system-on-chip that interacts with the world probably needs mixed-signal elements that combine analog and digital processing. However, many digital VLSI manufacturing processes are not especially good at analog components. It may be possible to put some types of analog interfaces on a largely digital chip. In other cases, it may be necessary to put the mixed-signal processing on a separate chip manufactured with a different process.
- **Memory systems.** Many systems-on-chips are geared toward memory-intensive applications. The on-chip memory system may include caches for the embedded CPUs as well as on-chip main memory. The on-chip memory may be flash memory, SRAM, or embedded DRAM. The chip may also need to connect to off-chip memory. The memory system must be carefully designed to meet performance and power requirements as well as to make best use of silicon area.
- **CPU selection.** A wide variety of CPUs are available as both soft and hard IP for use in systems-on-chips. Some processors are **configurable**—the designer can choose a number of CPU design parameters, such as added instructions and data path bit width, then generate a custom CPU optimized for the application at hand. The choice of CPUs is determined by a variety of factors ranging from program performance to software compatibility through development environment.
- **Hardware/software co-design.** Many systems-on-chips require sophisticated architectures: multiple CPUs, custom memory systems, and I/O

be inserted and power consumption can be analyzed. A more detailed set of timing checks can also be performed, including delay, clock skew, and setup/hold times. In extreme cases, perhaps because of a limited number of choices in the gate and register libraries, it may be necessary to make more drastic changes to the logic to correct problems found late in the logic design process.

- **Physical design.** Physical design starts with floorplanning to determine the overall structure of the layout. If the logic was designed in large blocks, it may be necessary to partition those large blocks into smaller pieces at this point. Placement and routing will generate layouts of blocks, or layouts can be designed by hand. Once the layout is complete, the wiring parasitics must be extracted and back-annotated to the logic design. The back-annotated design can then be simulated to verify that layout did not violate any timing constraints. Hopefully, problems can be fixed with minor modifications to the layout but changes to the logic design may be required.
- **Back-end checks.** ATPG must be performed late to ensure that minor design changes did not inadvertently cause testability problems. Similarly, design-rule and electrical checks of the complete layout are an important sanity check to ensure that shorts or opens were not introduced late in design.

As we saw in Chapter 8, large chips are often designed with intellectual property (IP) blocks. IP-based design offers advantages over reducing design time: the blocks have been functionally verified; and they provide more accurate estimates of area, speed, and power early in the design process. Providers of libraries generally refer to two major categories of pre-defined components: **hard macros** are complete layouts, while **soft macros** are specified as logic or perhaps behavior-level blocks. Hard macros, because they are complete designs, provide accurate information about area, delay, and power as soon as they are selected. On the other hand, hard macros are harder to port to new processes and harder to modify to meet variations in customer requirements. Soft macros, by comparison, are not as well-characterized but can be more easily adapted to changing needs. Macros have a broad range, ranging from memory arrays through UARTs, to central processing units.

Integrated circuits are notoriously hard to debug after fabrication. The few hundred pins on a large chip cover only a tiny fraction of the state contained in the multiple millions of electrical nodes in a large VLSI IC. While it may be possible to deduce the internal behavior of the chip, some errors may be difficult to detect and may also mask other flaws. **Voltage contrast** [Ben95] is a technique for observing the chip's internal behavior using a scanning electron microscope (SEM). The electron beam of the SEM is reflected differently off electrical nodes at high and low voltages; a raster