Digital Design

Principles and Practices

Fourth Edition

John F. Wakerly

Network System Technologies Ex. 2006
Qualcomm v. Network System Technologies - IPR 2024-00355

Wakerly, John F.
Digital design: principles and practices/ John F. Wakerly.--4th ed.
p. cm.
Includes index
ISBN 0-13-186389-4
1. Digital integrated circuits--Design and construction. I. Title
TK7874.65,W34 2005
621.39'5-dc22
2005048710

Vice President and Editorial Director, ECS: *Marcia Horton*
Editorial Assistant: *Richard Virginia*
Executive Managing Editor: *Vince O'Brien*
Managing Editor: *David A. George*
Production Editor: *Scott Disanno*
Director of Creative Services: *Paul Belfanti*
Art Director: *Kenny Beck*
Cover Designer: *Bruce Kenselaar*
Art Editor: *Xiaohong Zhu*
Manufacturing Manager: *Alexis Heydt-Long*
Manufacturing Buyer: *Lisa McDowell*
Senior Marketing Manager: *Holly Stark*
About the Cover: Original cover artwork © 2001 Ken Bakeman, www.kennyzen.com

© 2006, 2000, 1994, 1990 by Pearson Education, Inc.
Pearson Prentice Hall
Pearson Education Inc.
Upper Saddle River, NJ 07458

Pearson Prentice Hall™ is a trademark of Pearson Education Inc.

The author and publisher of this book have used their best efforts in preparing this book. These efforts include the development, research, and testing of the theories and programs to determine their effectiveness. The author and publisher shall not be liable in any event for incidental or consequential damages with, or arising out of, the furnishing, performance, or use of these programs.

Verilog is a trademark of Cadence Design Systems, Inc. Silos is a trademark of Simucad Inc. Synopsys, and Foundation Express are trademarks of Synopsys, Inc. Xilinx® is a registered trademark of Xilinx Corp. Aldec is a trademark of Aldec.

ISBN 0-13-186389-4

10 9 8 7 6 5 4 3 2 1

Pearson Education Ltd., London
Pearson Education Australia Pty., Ltd., Sydney
Pearson Education Singapore, Pte. Ltd.
Pearson Education North Asia Ltd., Hong Kong
Pearson Education Canada, Inc., Toronto
Pearson Educación de Mexico, S.A. de C.V.
Pearson Education—Japan, Tokyo
Pearson Education Malaysia, Pte. Ltd.
Pearson Education, Inc., Upper Saddle River, New Jersey

Network System Technologies Ex. 2006
Qualcomm v. Network System Technologies - IPR 2024-00355

HDLs such as Verilog and VHDL also support hierarchical design; for example, a module can instantiate another module. In a hierarchical design, it's possible for some modules (or "hierarchical schematic pages") to be specified by gate-level logic diagrams, while others are specified by HDL programs. In such a "mixed" environment, a given schematic page may contain gates, other MSI and LSI hardware components, and blocks that represent HDL modules or other schematic pages.

Most computer-aided logic design environments support both flat and hierarchical schematics. As in HDLs, proper signal naming is very important in both styles, since there are a number of common errors that can occur:

- Like any other program, a schematic-entry program does what you say, not what you mean. If you use slightly different names for what you intend to be the same signal on different pages, they won't be connected.
- Conversely, if you inadvertently use the same name for different signals on different pages of a flat schematic, many programs will dutifully connect them together, even if you haven't connected them with an off-page flag. (In a hierarchical schematic, reusing a name at different places in the hierarchy is generally OK, because the program qualifies each name with its position in the hierarchy, that is, based on its scope.)
- In a hierarchical schematic, you have to be careful in naming the external interface signals on pages in the lower levels of the hierarchy. These are the names that will appear inside the blocks corresponding to these pages when they are used at higher levels of the hierarchy. It's very easy to transpose signal names or use a name with the wrong active level, yielding incorrect results when the block is used.
- This is not usually a naming problem, but all schematic programs seem to have quirks in which signals that appear to be connected are not. Using the "T" convention in Figure 6-13 can help minimize this problem.

Fortunately, most schematic programs have error-checking facilities that can catch many of these errors, for example, by searching for signal names that have no inputs, no outputs, or multiple outputs associated with them. But most logic designers learn the importance of careful, manual schematic double-checking only through the bitter experience of building a printed-circuit board or an ASIC based on a schematic containing some dumb error.

6.1.8 Buses

As defined previously, a bus is a collection of two or more related signal lines. For example, a microprocessor system might have an address bus with 16 lines, ADDR0–ADDR15, and a data bus with 8 lines, DATA0–DATA7. The signal names in a bus are not necessarily related or ordered as in these first examples. For example, a microprocessor system might have a control bus containing five signals, ALE, MIO, RD_L, WR_L, and RDY.

Network System Technologies Ex. 2006
Qualcomm v. Network System Technologies - IPR 2024-00355