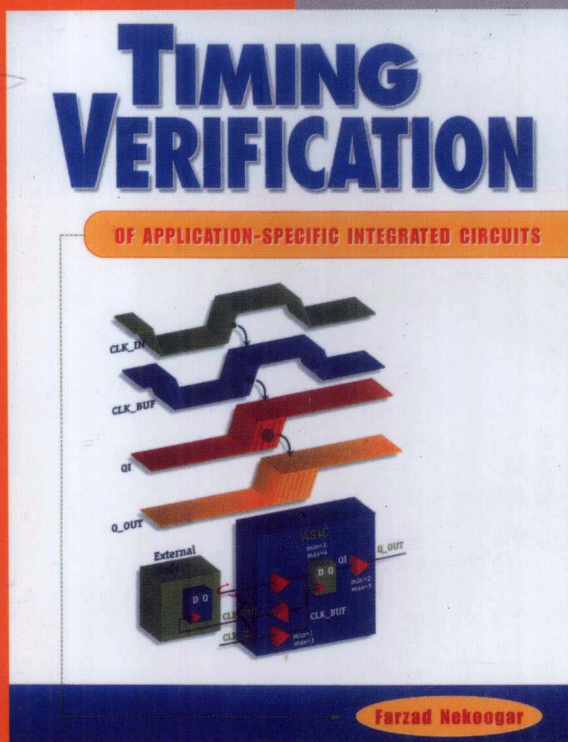


PEARSON

国外大学优秀教材 —— 微电子类系列 (影印版)

Farzad Nekoogar

专用集成电路 时序验证



PEARSON
Education

清华大学出版社

PEARSON

国外大学优秀教材 —— 微电子类系列（影印版）

专用集成电路 时序验证

Timing Verification of
Application-Specific Integrated Circuits

Farzad Nekoogar

清华大学出版社
北京

北京市版权局著作权合同登记号 图字: 01-2009-5131

Original edition, entitled TIMING VERIFICATION OF APPLICATION-SPECIFIC INTEGRATED CIRCUITS (ASICs), 9780137943487 by FARZAD NEKOOGAR, published by Pearson Education, Inc, publishing as Prentice Hall PTR, copyright © 1999.

All Rights Reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

China edition published by PEARSON EDUCATION ASIA LTD., and TSINGHUA UNIVERSITY PRESS
Copyright 2009

This edition is manufactured in the People's Republic of China, and is authorized for sale only in the People's Republic of China excluding Hong Kong, Macao and Taiwan.

**For sale and distribution in the People's Republic of China exclusively
(except Taiwan, Hong Kong SAR and Macao SAR).**

仅限于中华人民共和国境内(不包括中国香港、澳门特别行政区和中国台湾地区)销售发行。

本书封面贴有 Pearson Education (培生教育出版集团) 激光防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13701121933

图书在版编目(CIP)数据

专用集成电路时序验证 = Timing Verification of Application-Specific Integrated Circuits (ASICs): 英文 / (美) 内库加 (Nekoogar, F.) 著. —影印本. —北京: 清华大学出版社, 2009.10
(国外大学优秀教材. 微电子类系列)

ISBN 978-7-302-21342-0

I. 专… II. 内… III. 集成电路—时序电路—验证—高等学校—教材—英文 IV. TN407

中国版本图书馆 CIP 数据核字 (2009) 第 187072 号

责任编辑: 邹开颜

责任印制: 何 芊

出版发行: 清华大学出版社

地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编: 100084

社 总 机: 010-62770175

邮 购: 010-62786544

投稿与读者服务: 010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈: 010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者: 北京市清华园胶印厂

经 销: 全国新华书店

开 本: 185×230 印 张: 13.25

版 次: 2009 年 11 月第 1 版

印 次: 2009 年 11 月第 1 次印刷

印 数: 1~3000

定 价: 25.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题, 请与清华大学出版社出版部联系调换。联系电话: 010-62770177 转 3103 产品编号: 034678-01

出版前言

微电子技术是信息科学技术的核心技术之一，微电子产业是当代高新技术产业群的核心和维护国家主权、保障国家安全的战略性产业。我国在《信息产业“十五”计划纲要》中明确提出：坚持自主发展，增强创新能力和核心竞争力，掌握以集成电路和软件技术为重点的信息产业的核心技术，提高具有自主知识产权产品的比重。发展集成电路技术的关键之一是培养具有国际竞争力的专业人才。

微电子技术发展迅速，内容更新快，而我国微电子专业图书数量少，且内容和体系不能反映科技发展的水平，不能满足培养人才的需求，为此，我们系统挑选了一批国外经典教材和前沿著作，组织分批出版。图书选择的几个基本原则是：在本领域内广泛采用，有很大影响力；内容反映科技的最新发展，所述内容是本领域的研究热点；编写和体系与国内现有图书差别较大，能对我国微电子教育改革有所启示。本套丛书还侧重于微电子技术的实用性，选取了一批集成电路设计方面的工程技术用书，使读者能方便地应用于实践。

我们真诚地希望，这套丛书能对国内高校师生、工程技术人员以及科研人员的学习和工作有所帮助，对推动我国集成电路的发展有所促进。也衷心期望着广大读者对我们一如既往的关怀和支持，鼓励我们出版更多、更好的图书。

清华大学出版社
理工出版事业部
2003.9

数字专用集成电路（ASIC）设计是一项覆盖学科内容广泛的系统工程，是将系统、逻辑与性能的设计要求转化为具体的物理版图的过程，也是将产品从抽象的设计要求一步步具体化、直至最终物理实现的过程，涉及系统级、算法级、微架构级、寄存器传输级、门级（网表级）、电路级、版图（GDSII）级等多个设计抽象层次。

成功地设计一款数字集成电路，不仅需要保证逻辑功能正确，还要充分保证各种电特性（如时序、压降、串扰等）因素以及各种商业需求（如面世时间、市场窗口、设计与生产成本等）得到充分的满足。

事实上，电特性方面的需求（尤其是时序因素）成为电路（专用集成电路和 FPGA）设计中贯穿从算法级到版图级中的每一个环节都不容轻视的设计考虑。时序因素不仅往往决定了算法级、微架构级以及寄存器传输级设计过程中究竟采用何种做法、策略与技巧，而且将决定后端物理设计各个环节的处理方式和可实现性，并最终决定该芯片制造出之后能否稳定地正常工作。这一切，必然决定了整个芯片的设计难度、设计周期、设计成本与生产成本（如硅片面积、生产工艺等），进而影响到该芯片的市场前景。

正如计算机软件中的任何一种算法求解过程都离不开特定的数据结构实现方式¹，可以毫不夸张地说：**数字集成电路=逻辑+时序**。

中文“时序”一词有两种不同的含义，对应的英文分别是 **sequence**（顺序、因果关系）和 **timing**（时机），前者是对信号间相互控制与处理的顺序和因果关系，属于逻辑的范畴；后者就是这里我们所言及的时序，它是指信号的变化在时间上的相对关系。维基百科全书上对 **timing** 的定义为“**Timing is the spacing of events in time**（即事件在时间上的间隔）”，即信号出现的时机是否恰当。

与逻辑相比较，时序是一个相对新的概念，因而很少有专业书籍论及这一概念及其相关论题。在学术界，**timing** 一词的中文术语尚未统一，也经常被翻译为“定时”、“时机”，甚至是“时间”；在工业界，人们往往习惯于将其称为“时序”。

本书是近年来惟一一本专门讨论时序及时序验证的专著。

本书前两章侧重于介绍时序及时序验证的概念与方法。它首先通过对比功能仿真、动态时序仿真与静态时序验证，指出静态时序验证是唯一快速有效并全面验证芯片是否满足其时序规约的手段；通过电路图和表格方式细致地介绍了时序方面的各种基础概念，如路径延迟、传播延迟、相位偏差、关键路径；解释了内在延迟和外在延迟的各种成因；并以

1 图灵奖 1984 年获得者尼克劳斯·沃思（Niklaus Wirth）曾提出著名的公式：程序=算法+数据结构。

i960 处理器的 EDRAM 接口时序图为例讲述了接口时序分析方法。

本书继而详细地阐述了时钟和时序规约、时序验证的各种概念；讨论了各种时钟方案，如门控时钟、时钟网络/时钟分布结构、多频率时钟和多相位时钟等；细致地讨论了静态时序分析中可以采用的各种手段，如设置无效路径、多周期路径、施加各方面时序约束。

本书后两章分别针对专用集成电路（ASIC）和基于可编程逻辑器件的设计展开时序方面的讨论。

专用集成电路在逻辑综合（即版图设计之前）还没有实际的布局布线信息，因此只能采用较为粗略的连线负载模型进行时序分析，而版图设计之后有了实际的布线结果，因而可以充分考虑各种寄生效应及相应的延迟模型精确地分析时序。本书第三章通过电路延迟模型分析了寄生效应对时序的作用机制；介绍了在 Verilog HDL 代码中标注延迟的各种方法；介绍了设计流程中各环节间用于交换延迟信息的各种标准延迟格式与寄生效应数据格式；讨论了 ASIC 设计过程中如何施加设计规则约束及优化手段；并给出了版图设计之后的信号线电容、电阻以及 RC（电容电阻）延迟的计算方法；探讨了 ASIC 送交生产前的签付检查应当履行的各项检查。

第四章结合最常用的可编程逻辑器件，如 Actel 公司、Altera 公司和 Xilinx 公司的器件，从设计流程、可编程逻辑结构、时序参数、时序分析、HDL 综合、软件开发系统等方面具体讨论了可编程逻辑器件及相关的时序问题。

本书全面地讨论了静态时序验证的各方面内容；全书不仅紧密结合电路图和波形图进行讲解，还结合 Synopsys 公司的逻辑综合和静态时序分析工具讲解如何通过命令加以实现；介绍过程中不仅从理论上阐述了延迟模型，而且注重实践环节，引入了大量实际示例加以深入探讨。相信这种写作风格将促进读者能够更全面、更细致地理解所讲内容，因此本书十分适合于（集成）电路工程师、FPGA 工程师以及（电路板级）硬件设计工程师自学，电子工程和微电子专业高年级本科生和研究生可以将本书用作时序分析与时序验证方面的教材和教学参考书。

孙海平

2009 年 10 月于同济大学嘉定校区

Preface

This book describes the theory and applications of timing verification of application-specific integrated circuits (ASICs). Timing verification is a relatively new concept, which is why most books on digital systems do not cover the issue. This book lays out the fundamental principles of effective timing verification, and it makes good use of the examples that reflect the current issues facing logic designers.

The following items characterize this book:

- ✦ Timing verification as opposed to functional verification is the primary focus. (Functional simulation has been adequately covered in other books.)
- ✦ Principles and techniques as opposed to specific tools are emphasized. Once designers understand the underlying principles of timing analysis, they can apply them with various timing tools.
- ✦ Design flow for deep-submicron ASICs and FPGA designs are fully covered.
- ✦ Numerous design examples and HDL codes to illustrate the concepts discussed in the book are provided.

This book is to be used for self-study by practicing engineers. Design and verification engineers who are working with ASICs and FPGAs will find the book very useful. Upper-level undergraduate and graduate students in electrical engineering can use it as a reference book in design courses in timing analysis and related topics.

The material covered in this book requires some understanding of the Electronic Design Automation (EDA) tools and an initial course in logic design.

The book is organized into two parts:

Part I (chapters 1 and 2) introduces the fundamental concepts involved in timing verification. Including clock definitions, multicycle paths, false paths, and phase-locked loops.

Part II (chapters 3 and 4) covers specific timing issues related to ASICs and FPGAs, respectively.

Chapter 1 gives an overview of timing verification and static timing analysis. It contrasts timing verification with functional verification. Typical goals of timing verification in digital systems are presented. This chapter ends with an example of interface timing analysis.

Chapter 2 introduces the concepts of timing analysis with design examples. It specifically discusses such clocking methods as gated clocks, multifrequency clocks, and multiphase clocks. It introduces the concepts of multicycle paths, false paths, and timing constraints (such as setup, hold, recovery, and pulse width).

Chapter 3 discusses the deep submicron ASIC design flow and application of timing analysis in the design process. It includes discussion of prelayout and postlayout timing verification. The chapter also discusses behavioral and structural RTL coding for timing, synthesis and timing constraint, and the ASIC sign-off checklist. We make the concepts concrete with numerous examples.

Chapter 4 discusses timing concepts in programmable logic-based designs. It covers design flow, timing parameters, timing analysis, and HDL synthesis and software development systems. We present the most commonly used programmable logic devices (Actel, Altera, and Xilinx) and associated timing issues.

Appendices A, B, and C discuss the EDA timing tools of PrimeTime, Pearl, and TimingDesigner respectively.

Appendix D covers some concepts of transistor-level timing verification.

ACKNOWLEDGMENTS

I would like to thank the following individuals from Intrinsix Corp. for their contributions to this book:

- ✧ Lawrence Letham, for authoring chapter 3 and contributing to other topics.
- ✧ Paul Brown, for authoring chapter 4.
- ✧ Faranak Nekoogar, for her knowledge of PrimeTime and her help on chapter 2 and appendix A.
- ✧ Gene Petili, for his contribution on the subject of transistor-level timing verification.
- ✧ Peter J. Militello, for his help in the area of phase-locked loops.
- ✧ Tomislav Ilic, for his help on chapter 2.
- ✧ Mark Beal, for the help he provided to this project.

In addition, I'd like to thank the following people and companies:

- ✧ The staff of Prentice Hall, especially Bernard Goodwin, for his support of this project.
- ✧ Wilbur Luo, director of field applications, Chronology Corporation.
- ✧ Terry Strickland, director of marketing, Chronology Corporation.
- ✧ The staff of BooksCraft, Inc., for their help in producing the book.

—Farzad Nekoogar

Contents

List of Figures	xi
List of Tables	xv
Preface	xvii
Acknowledgments	xix
1 Introduction to Timing Verification	1
1.1 Introduction	1
1.2 Overview of Timing Verification	2
1.2.1 Intrinsic vs. Extrinsic Delay	4
1.2.2 Path Delay	7
1.3 Interface Timing Analysis	11
2 Elements of Timing Verification	17
2.1 Introduction	17
2.2 Clock Definitions	17
2.2.1 Gated Clocks	19
2.2.2 Clock Skews and Multiple Clock Groups	22
2.2.3 Multifrequency Clocks	25
2.2.4 Multiphase Clocks	26
2.3 More on STA	28
2.3.1 False Paths	28
2.3.2 Multicycle Path Analysis	30
2.3.3 Timing Specifications	31
2.3.4 Timing Checks	35
2.4 Timing Analysis of Phase-Locked Loops	38
2.4.1 PLL Basics	38
2.4.2 PLL Ideal Behavior	39
2.4.3 PLL Errors	41

3 Timing in ASICs	45
3.1 Introduction	45
3.2 Prelayout Timing	48
3.2.1 RTL vs. Gate-Level Timing	49
3.2.2 Timing in RTL Code	50
3.2.3 Delay with a Continuous Assignment Statement	53
3.2.4 Delay in a Process Statement	54
3.2.5 Intra-Assignment Delays	56
3.2.6 The Verilog Specify Block	58
3.2.7 Timing in-Gate Level Code	64
3.2.8 Synthesis and Timing Constraints	64
3.2.9 Design Rule Constraints	65
3.2.10 Optimization Constraints	66
3.2.11 Gate and Wire-Load Models	69
3.2.12 The Synthesis Flow	72
3.2.13 Synthesis Tips	81
3.2.14 Back Annotation to Gate-Level RTL	82
3.3 Postlayout Timing	83
3.3.1 Manual Line-Propagation Delay Calculations	83
3.3.2 Signal-Line Capacitance Calculation	84
3.3.3 Signal Line Resistance Calculation	90
3.3.4 Signal Trace RC Delay Evaluation	92
3.4 ASIC Sign-Off Checklist	93
3.4.1 Library Development	94
3.4.2 Functional Specification	94
3.4.3 RTL Coding	94
3.4.4 Simulations of RTL	95
3.4.5 Logic Synthesis	95
3.4.6 Test Insertion and ATPG	95
3.4.7 Postsynthesis Gate-Level Simulation or Static Timing Analysis	95
3.4.8 Floorplanning	96
3.4.9 Place and Route	96
3.4.10 Final Verification of the Extracted Netlist	96
3.4.11 Mask Generation and Fabrication	97
3.4.12 Testing	97
4 Programmable Logic Based Design	99
4.1 Introduction	99
4.2 Programmable Logic Structures	101
4.2.1 Logic Block	105
4.2.2 Input/Output Block	106
4.2.3 Routing Facilities	106
4.3 Design Flow	108

4.4 Timing Parameters	111
4.4.1 Timing Derating Factors.....	112
4.4.2 Grading Programmable Logic Devices by Speed	114
4.4.3 Best-Case Delay Values	115
4.5 Timing Analysis	116
4.5.1 Actel ACT FPGA Family	117
4.5.2 Actel ACT 3 Architecture	117
4.5.3 Actel ACT 3 Timing Model.....	118
4.5.4 Altera FLEX 8000	122
4.5.5 Altera FLEX 8000 Architecture.....	122
4.5.6 Altera FLEX 8000 Timing Model	123
4.5.7 Xilinx XC3000 /XC4000 FPGA Families	128
4.5.8 Xilinx XC9500 CPLD.....	130
4.5.9 Xilinx XC9500 CPLD Architecture.....	131
4.5.10 Xilinx XC9500 CPLD Timing Model	133
4.6 HDL Synthesis	137
4.7 Software Development Systems.....	139
4.7.1 Timing Constraints	140
4.7.2 Operating Conditions	142
4.7.3 Static Timing Analysis	142
4.7.4 Vendor-Specific Timing-Verification Tools	143
4.7.5 Actel Designer	144
4.7.6 Altera MAX+PLUS II.....	146
4.7.7 Xilinx XACT/M1	147
A PrimeTime	151
B Pearl	157
C TimingDesigner	161
D Transistor-Level Timing Verification	163
References.....	169
Index.....	173
About the Author.....	185

List of Figures

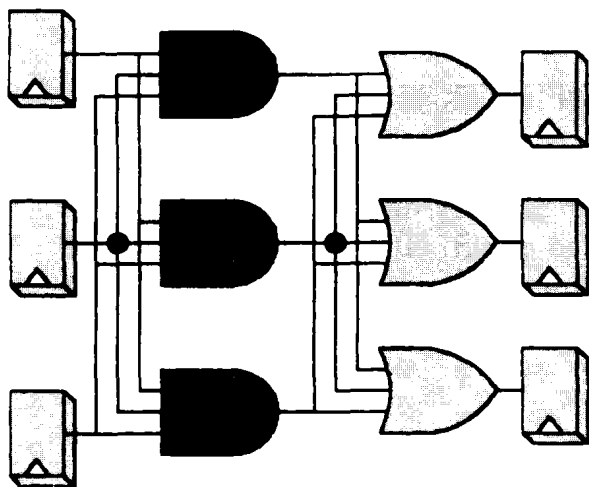
Fig. 1.1 Typical ASIC Design Flow	2
Fig. 1.2 Spiral Design Flow	4
Fig. 1.3 STA Measures Path Delays Through a Circuit.....	7
Fig. 1.4 STA Is Best Suited for Use on Synchronous Circuits.....	8
Fig. 1.5 Time Available for Propagation Between Two Flip-Flops Depends on Clock Skew and Flip-Flop Setup Time.....	9
Fig. 1.6 Block-Diagram of the System in Example 1.1	14
Fig. 1.7 Merged Read Diagram of the CPU Plus the EDRAM	15
Fig. 2.1 Single Clock Pulse: (a) Positive Clock Pulse (b) Negative Clock Pulse.....	18
Fig. 2.2 Rising/Falling Edges and Uncertainties in a Clock.....	18
Fig. 2.3 Positive Clock Pulse in Relation with Rising-Edge Triggered Flip-Flop.....	19
Fig. 2.4 Negative Clock Pulse in Relation with Falling-Edge Triggered Flip-Flop.....	20
Fig. 2.5 A Clock Network	20
Fig. 2.6 Clock Distribution Structure	21
Fig. 2.7 Gated Clock.....	21
Fig. 2.8 Polarity Skew.....	23
Fig. 2.9 Phase Skew	24
Fig. 2.10 Frequency Skew.....	24
Fig. 2.11 Cycle Skew	25
Fig. 2.12 Multifrequency Clock	26
Fig. 2.13 Multiphase Clock.....	27
Fig. 2.14 Setup and Hold Times in Generic Pipeline Stage of Synchronous Design	28
Fig. 2.15 Slight Modification to Circuit in Figure 1.3 Makes the a-out Path a False Path	29
Fig. 2.16 Valid Paths That Take More Than One Cycle to Complete Must Be Identified as Multicycle Paths	31
Fig. 2.17 Zero Cycle Path.....	32

Fig. 2.18 Time Available for Propagation Between Two Flip-Flops Depends on Clock Skew and Flip-Flop Setup Time	33
Fig. 2.19 CAD Tools Offer Two Delay Models	35
Fig. 2.20 Phase-Locked Loop Block Diagram	38
Fig. 2.21 Ideal PLL Behavior	40
Fig. 2.22 Nonideal PLL Behavior	42
Fig. 3.1 Accurate Models Require Inclusion of Parasitic Capacitors	46
Fig. 3.2 Components of Circuit Delay: Input Slew Rate, Inherent Gate Delay, Line Propagation Delay, Fanout Load	47
Fig. 3.3 Parasitic Capacitance of a Metal Line	48
Fig. 3.4 Circuits Can Be Represented as RTL Code or Instantiated Gates	49
Fig. 3.5 Delays of Entire Modules Are Easy to Implement in an HDL	51
Fig. 3.6 RAM Timing Diagram	52
Fig. 3.7 Signals Corresponding to Example 3.3	54
Fig. 3.8 Signals from the Blocking Assignment Statements with Regular Delays from Example 3.4	55
Fig. 3.9 Signals from Blocking Assignment Statements with Intra-Assignment Delays from Example 3.5	57
Fig. 3.10 Signals from Nonblocking Assignment Statements with Intra-Assignment Delays from Example 3.6	59
Fig. 3.11 Partial Timing Diagram for a Synchronous RAM	60
Fig. 3.12 The set_input_delay Command Adds Additional Delay to Module Input Times	68
Fig. 3.13 The set_output_delay Command Adds Additional Delay to Module Output Times	68
Fig. 3.14 Circuit Corresponding to the SDF Constraint File of Example 3.14	75
Fig. 3.15 Circuit Corresponding to the SDF File of Example 3.15	77
Fig. 3.16 Parasitic Capacitors Exist Between All Layers of a Process	85
Fig. 3.17 Conductors Have Both Plate and Fringe Capacitors	86
Fig. 3.18 Fringe Capacitance Can Be Modeled as a Cylindrical Conductor	87
Fig. 3.19 Line Resistance Is Measured in Squares	90
Fig. 3.20 Distributed RC Model Provides Greater Accuracy than Lumped Model	92
Fig. 4.1 CPLD Structure	102
Fig. 4.2 Simplified FPGA Structure	103
Fig. 4.3 Xilinx X3000/X4000 Family Routing Facilities	107
Fig. 4.4 Actel ACT Family Routing Facilities	108
Fig. 4.5 FPGA Design Flow	109
Fig. 4.6 Actel ACT 3 Array Structure	118

Fig. 4.7 Actel ACT 3 Timing Model (Values Shown for Actel A1425A-3 at Worst Case Commercial Conditions).....	119
Fig. 4.8 Actel ACT 3 Example Timing Calculation	121
Fig. 4.9 Altera FLEX 8000 Array Structure	123
Fig. 4.10 Altera FLEX 8000 Timing Model.....	124
Fig. 4.11 Altera FLEX 8000 Timing Model, Simplified for Example 4.2	127
Fig. 4.12 Simplified Block Diagram of the Xilinx XC4000 Series CLB.....	129
Fig. 4.13 The Routing Delays in Xilinx XC3000/4000 Devices Are Not Deterministic Prior to Placement and Routing	131
Fig. 4.14 Xilinx XC9500 CPLD Architectural Features.....	132
Fig. 4.15 Product Term Allocation with 15 Product Terms.....	133
Fig. 4.16 Xilinx XC9500 CPLD Timing Models	134
Fig. 4.17 Synthesis Design Flow (Exemplar Leonardo to Altera MAX+ PLUS II)	139
Fig. 4.18 DirectTime Flow	144
Fig. 4.19 Altera MAX+PLUS II Design Environment	146
Fig. 4.20 Timing Analyzer in the Xilinx Design Flow	148
Fig. A.1 PrimeTime's Input/Output Files	153
Fig. A.2 PrimeTime User Flow.....	154
Fig. A.3 Exemplar Synthesis Tool Input/Outputs	155
Fig. A.4 AMBIT Synthesis Tool Input/Outputs	155

List of Tables

Table 1.1 Delay Paths of Figure 1.3	8
Table 1.2 Paths Measured in STA	10
Table 1.3 Truth Table for Circuit in Figure 1.3	10
Table 3.1 Synchronous Memory Timing Parameters	60
Table 3.2 Typical Metal Widths for 0.8 μ m Process	88
Table 3.3 Typical Capacitance Values for 0.8 μ m Process	88
Table 3.4 Capacitance of Long Lines	89
Table 3.5 Typical Material Resistance in a 0.8 μ m Process	91
Table 3.6 Line resistances for Figure 3.19	91
Table 3.7 RC Delay Times	93
Table 4.1 PAL-to-Gate Array Continuum	103
Table 4.2 Performance for CMOS Devices as a Function of Voltage, Temperature, and Process	112
Table 4.3 Actel ACT 3 Temperature and Voltage Derating Factors (Normalized to Worst-Case Commercial Conditions, $T_J = 4.75$ V, 70°C)	113
Table 4.4 Actel ACT 3 Operating Condition Derating Factors (Normalized to Worst-Case Commercial Conditions)	113
Table 4.5 ACT 3 Logic Module and Routing Delays as a Function of Speed Grade and Fanout (Worst-Case Commercial Conditions)	114
Table 4.6 Speed Grade Designations	115
Table 4.7 Actel ACT Family Timing Parameter Definitions	120
Table 4.8 Altera FLEX 8000 Timing Parameters	124
Table 4.9 Xilinx XC4000E CLB Switching Characteristic Guidelines (partial listing)	130
Table 4.10 Xilinx XC95144 Timing Parameters (partial listing)	135
Table 4.11 Xilinx XC9500 Family Performance Summary	136
Table 4.12 Summary of Vendor Timing-Verification Facilities	143
Table D.1 Table of EDA Timing Tool	164



Introduction to Timing Verification

1.1 INTRODUCTION

In this book we introduce the concept of timing verification of ASICs. The fundamental concepts can be applied to ASIC, field programmable gate array (FPGA), and system-on-a-chip designs. It should be noted that most of the material presented in this book is electronic design automation (EDA)-tool independent. However, any mention of specific commands for Static Timing Verification (STA) is based on the Synopsys PrimeTime static timing analyzer. Also, any mention of a specific synthesis tool or command refers to Synopsys Design Compiler and its related modules.

Overview of some current EDA tools that are used for timing verification can be found in appendices A, B, and C. Also, Verilog-HDL is used throughout the book for hardware description language (HDL) coding examples.

In this chapter we cover the intrinsic versus extrinsic and static versus dynamic timing verification. We present the concept of path delay, which is the fundamental concept in STA. We also discuss timing interface, and an example of interfacing an Intel i960 processor to an EMS EDRAM through an FPGA that acts as the memory controller is presented.