

CPLD/ FPGA

高级应用开发指南

任晓东 文博 编著



含·光·盘



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

CPLD/FPGA 高级应用开发指南

任晓东 文 博 编著

電子工業出版社·

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

目前,CPLD/FPGA 被广泛地应用在航天、通信、医疗、工控等各个领域。本书从现代电子系统设计的角度出发,基于全球最大的可编程逻辑器件生产厂商 Xilinx 公司的系列产品,系统全面介绍了 CPLD/FPGA 的内部结构、设计流程和开发工具,详细论述了 CPLD/FPGA 设计的时序约束、仿真验证和综合实现,重点介绍了嵌入 PowerPC405 的平台 FPGA 开发以及 FPGA 在 DSP 领域的最新应用。

对于从事各类电子系统设计的科研人员和应用技术工程师,本书是一本不可多得的针对电子系统原理和开发的参考书。本书也可作为有志于从事专用数字电路设计的高等院校高年级本科生和研究生的教学参考书。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

CPLD/FPGA 高级应用开发指南/任晓东编著. —北京:电子工业出版社,2003.6
ISBN 7-5053-8764-2

I . C . . . II . 任 . . . III . 可编程序逻辑器件 - 指南 IV . TP332.1 - 62

中国版本图书馆 CIP 数据核字(2003)第 040912 号

责任编辑:刘海艳 E-mail: lhy@phei.com.cn

印 刷: 北京天宇星印刷厂

出版发行: 电子工业出版社 <http://www.phei.com.cn>

北京市海淀区万寿路 173 信箱 邮编 100036

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 17.75 字数: 450 千字 附光盘 1 张

版 次: 2003 年 6 月第 1 版 2003 年 6 月第 1 次印刷

印 数: 5000 册 定价: 29.00 元(含光盘)

凡购买电子工业出版社的图书,如有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系。
联系电话: (010)68279077

前 言

当今世界,集成电路产业的发展日新月异,随着微电子工艺技术的不断进步,集成电路的性能也迅速提高。据国际权威机构估计,2010 年动态存储器的存取时间将降低至 10ns 以下,电源电压有可能降至 0.6V,数字电路的时钟频率可以达到 3GHz。与此同时,芯片的集成度和电路规模将不断提高,工业化生产的典型硅集成电路的管芯面积将达到 $2.5\text{cm} \times 2.5\text{cm}$,特征尺寸将下降到 $0.07\mu\text{m}$ 。

随着微电子技术的飞速发展,传统的可编程逻辑器件正在向大容量、高性能、低成本的方向发展,以 CPLD/FPGA 为代表的可编程逻辑器件受到了世界范围内广大电子设计工程师的普遍欢迎,应用日益广泛。在国内,越来越多的电子设计工程师迫切希望了解最新的 CPLD/FPGA 器件结构特性,熟悉先进的 CPLD/FPGA 设计工具,掌握高级的数字集成电路设计方法,本书正是在这样的背景下出版的。

Xilinx 公司是全球最大的可编程逻辑器件生产厂商,其 CPLD/FPGA 产品和 ISE 系列软件在国内的各个领域得到了广泛应用。本书基于 Xilinx 公司 CPLD/FPGA 系列产品,结合作者多年的数字集成电路设计经验,系统介绍 Xilinx 公司的 CPLD/FPGA 系列产品的结构特性、ISE 系列软件开发流程,详细论述 Xilinx 公司的 CPLD/FPGA 的高级设计技巧,具体讨论了 CPLD/FPGA 的仿真验证和设计综合,重点介绍了基于嵌入式处理器的平台 FPGA 软硬件设计以及 FPGA 在数字信号处理领域的应用。

全书共分为 12 章:第 1 章为可编程逻辑器件与 EDA 技术概述;第 2 章和第 3 章分别系统介绍 Xilinx CPLD/FPGA 系列产品的内部结构和技术特性;第 4 章全面介绍 Xilinx ISE 系列软件开发流程和基本开发工具;第 5 章详细介绍 Xilinx 高级设计工具,包括 Floorplanner、FPGA Editor、XPower 和 ChipScope;第 6 章和第 7 章详细介绍 Xilinx CPLD/FPGA 开发过程中的高级设计技巧,包括时序约束与分析、下载配置设计、DLL/DCM 设计技巧、BlockRAM 使用等;第 8 章和第 9 章结合 ModelSim 和 Synplify 的工具使用,详细介绍 CPLD/FPGA 设计过程中的仿真验证与设计综合;第 10 章介绍嵌入式系统软硬件设计的基本概念;第 11 章围绕 Xilinx 公司最新的 FPGA 产品,详细介绍基于 PowerPC405 处理器的平台 FPGA 软硬件设计;第 12 章详细介绍 Xilinx FPGA 在数字信号处理领域的应用。本书图文并茂、示例丰富,并附带一张实验光盘。光盘中的实验可以帮助读者更好更快地理解书中的基本概念和设计的方法技巧。

本书由任晓东同志和文博同志共同执笔,得到了清华大学微波与数字通信国家重点实验室 CAD 中心研究生刘鸿和个人通信系统实验室刘柳等的大力支持。威盛电子(中国)有限公司王镔同志也给了极大的帮助和鼓励,在此表示衷心感谢。本书第 9 章还得到上海冠讯科技有限公司杨建中的协助,在此谨表谢意。

值得一提的是,本书所针对的是 Xilinx 公司的 CPLD/FPGA 的系列产品,其电路中的符号和物理量均采用 Xilinx 公司常用模式。

本书取材于国际上关于 CPLD/FPGA 设计的最新文献和专著,综合了分散于不同文献中

的知识和设计经验,也包含了作者多年的数字集成电路设计开发经验。作者有幸通过本书的出版与广大读者交流 CPLD/FPGA 设计方面的体会,并期望为我国 CPLD/FPGA 设计人才的培养与信息产业的发展贡献绵薄之力。但是限于作者的水平,书中必有不妥和错误之处,殷切期待读者的批评和指正。

作 者
北京清华园

目 录

第 1 章 可编程逻辑器件与 EDA 技术	1
1.1 可编程逻辑器件及其特征	1
1.1.1 简单 PLD	1
1.1.2 CPLD	2
1.1.3 FPGA	2
1.1.4 门阵列	3
1.1.5 标准单元法	3
1.2 FPGA 的应用及设计要求	4
1.2.1 FPGA 的应用	4
1.2.2 FPGA 设计要求	5
1.3 EDA 技术简介及软件	6
1.3.1 EDA 技术	6
1.3.2 FPGA 开发用的软件	7
1.4 因特网上的 FPGA 及其他资源	7
1.5 小结	9
第 2 章 Xilinx CPLD 系列产品	10
2.1 XC9500 系列产品	10
2.1.1 简介	10
2.1.2 器件结构	11
2.2 CoolRunner-II 系列产品	16
2.2.1 简介	16
2.2.2 器件结构	17
2.2.3 设计技巧	19
2.3 小结	21
第 3 章 Xilinx FPGA 系列产品	22
3.1 Spartan-II E 系列产品	22
3.1.1 概述	22
3.1.2 器件结构	23
3.2 Virtex-II 系列产品	27
3.2.1 概述	27
3.2.2 器件结构	28
3.3 Virtex-II Pro 系列产品	33
3.3.1 概述	33
3.3.2 器件结构	34

3.4 小结	41
第4章 Xilinx ISE 应用基础	42
4.1 ISE 设计流程	42
4.1.1 设计输入	43
4.1.2 功能仿真	45
4.1.3 综合	45
4.1.4 实现	46
4.1.5 时序仿真	48
4.1.6 下载配置	48
4.2 ISE 基本设计工具	49
4.2.1 Architecture Wizard	49
4.2.2 PACE	49
4.2.3 内核生成器(CORE Generator)	50
4.2.4 iMPACT	52
4.2.5 PROM File Formatter	57
4.3 小结	59
第5章 FPGA 高级设计技巧(一)—— ISE 高级工具	60
5.1 Floorplanner	60
5.1.1 概述	60
5.1.2 Floorplanner 使用说明	61
5.2 FPGA Editor	61
5.2.1 概述	61
5.2.2 FPGA Editor 使用说明	63
5.3 XPower	65
5.3.1 概述	65
5.3.2 XPower 使用说明	66
5.4 ChipScope	68
5.4.1 概述	68
5.4.2 Chipscope 使用说明	68
5.5 小结	72
第6章 FPGA 高级设计技巧(二)—— 约束与配置	73
6.1 约束设计	73
6.1.1 概述	73
6.1.2 时序约束	74
6.1.3 其他重要的约束参数	81
6.1.4 UCF 文件和 Constraint Editor	82
6.2 配置设计	83
6.2.1 概述	83
6.2.2 下载配置模式	84
6.2.3 CPLD 下载配置设计	84

6.2.4	FPGA 下载配置设计	85
6.2.5	Virtex-II 系列器件下载配置设计	86
6.3	小结	90
第 7 章	FPGA 高级设计技巧(三)——时钟与 RAM	92
7.1	时钟设计技巧	92
7.1.1	全局时钟网络	92
7.1.2	数字延迟锁相环(DLL)	94
7.1.3	数字时钟管理器(DCM)	95
7.2	BlockRAM 设计技巧	102
7.2.1	BlockRAM 设计基础	102
7.2.2	BlockRAM 设计说明	103
7.3	小结	105
第 8 章	FPGA 设计仿真验证——ModelSim	106
8.1	数字电路设计的仿真验证	106
8.1.1	仿真验证基本概念	106
8.1.2	Testbench 基本概念	107
8.1.3	Testbench 设计提示	107
8.1.4	Testbench 设计示例	108
8.2	ModelSim 应用说明	109
8.2.1	ModelSim 基本命令	109
8.2.2	ModelSim 应用说明	112
8.3	小结	122
第 9 章	基于 Synplify 的 FPGA 综合设计	123
9.1	Synplify 和 Synplify Pro 概述	123
9.1.1	选择 Synplify Pro 的原因	123
9.1.2	FPGA 设计流程	126
9.1.3	Synplify Pro 界面	126
9.2	Synplify Pro 使用入门	127
9.2.1	建立一个工程	127
9.2.2	编译	127
9.2.3	设置 Xilinx 约束	129
9.2.4	设置与 Xilinx 器件有关的选项	131
9.2.5	执行综合	132
9.3	综合报告的解析	133
9.4	时序修正	138
9.5	Verilog HDL 的设计风格	139
9.5.1	在编写代码之前的准备工作	139
9.5.2	可综合的 Verilog HDL 风格	143
9.5.3	Verilog HDL 代码风格	144
9.5.4	程序员提示	146

9.6 小结	151
第 10 章 嵌入式系统的软硬件设计	152
10.1 嵌入式系统的结构	153
10.1.1 CISC 结构	153
10.1.2 RISC 结构	154
10.2 系统的软硬件协同设计	155
10.3 设计重用与基于平台的设计	156
10.4 软硬件协同设计的方法	158
10.5 以太网桥的设计实例	160
10.5.1 以 LOTOS 为基础的协同设计	160
10.5.2 要设计的以太网桥	162
10.5.3 实例的具体分析	163
10.5.4 结果分析	169
10.6 小结	170
第 11 章 嵌入 PowerPC 的平台 FPGA 软硬件设计	171
11.1 PowerPC405 内部结构和外部接口	172
11.1.1 PowerPC405 内部结构	172
11.1.2 PowerPC405 外部接口	173
11.2 EDK 应用说明	178
11.2.1 系统描述文件	178
11.2.2 EDK 开发流程	180
11.3 嵌入 PowerPC 的平台 FPGA 设计	181
11.3.1 嵌入 PowerPC 的平台 FPGA 设计	181
11.3.2 嵌入 PowerPC 的平台 FPGA 设计实例	183
11.4 小结	188
第 12 章 Xilinx FPGA 在数字信号处理领域的应用	189
12.1 System Generator 基础	189
12.1.1 系统级建模	190
12.1.2 DSP 的设计流程	190
12.1.3 算术数据类型	191
12.1.4 硬件握手	192
12.1.5 按位和按周期对齐的建模	193
12.2 软件使用基础	193
12.2.1 一般设计步骤	194
12.2.2 Simulink 系统周期与硬件时钟频率	196
12.2.3 同步机制	197
12.2.4 创建高性能设计的提示	199
12.2.5 LogiCORE	199
12.2.6 自动创建的文件	200
12.3 积分的例子	201

12.3.1	Simulink 建模	201
12.3.2	VHDL 等文件的生成	203
12.3.3	ModelSim 仿真	203
12.3.4	Synplify Pro 综合	205
12.3.5	翻译、映射、布局布线	206
12.4	自编写 VHDL 的例子	206
12.4.1	计数器的 VHDL 代码	206
12.4.2	创建计数器的 Simulink 仿真模型	207
12.4.3	计数器例子的文件	211
12.4.4	运行 System Generator	211
12.4.5	计数器的仿真和综合	211
12.5	自编写 Verilog HDL 的例子	212
12.5.1	计数器的 Verilog HDL 代码	212
12.5.2	创建计数器的 Simulink 仿真模型	213
12.5.3	计数器例子的文件	217
12.5.4	运行 System Generator	217
12.5.5	计数器的仿真和综合	217
12.6	小结	218
附录 A	IC 和 FPGA 设计新手必读	219
A.1	引言	219
A.2	硬件描述语言的诞生及电子设计自动化工具的发展	220
A.3	各种 EDA 工具介绍与自动化设计流程之关系	221
A.3.1	系统规范制定(Define Specification)	222
A.3.2	设计描述(Design Description)	222
A.3.3	功能验证(Function Verification)	222
A.3.4	逻辑电路综合(Logic Synthesis)	223
A.3.5	逻辑门级的电路功能验证(Gate-Level Netlist Verification)	223
A.3.6	布局与布线(Place and Routing)	224
A.3.7	布线后的电路功能验证(Post Layout Verification)	224
A.4	结论	224
附录 B	Verilog HDL 与 VHDL 的比较	225
1.	VHDL	225
2.	Verilog HDL	225
3.	两种语言的描述层次	225
4.	1995 年 HDL 的使用情况	226
5.	VHDL 与 Verilog HDL 在各方面的比较	226
附录 C	Virtex-II 和 Spartan-II E 的管脚	228
附录 D	实验指导	230
实验 1	ISE 应用基础	230
	实验目的	230

实验内容·····	230
实验说明·····	230
实验步骤·····	230
实验 2 ModelSim 应用基础 ·····	239
实验目的·····	239
实验内容·····	239
实验说明·····	239
实验步骤·····	239
实验 3 Synplify 的设计实例 ·····	242
实验目的·····	242
实验内容·····	242
实验说明·····	243
实验步骤·····	243
实验 4 基于 PowerPC405 的 FPGA 设计实例 ·····	245
实验目的·····	245
实验内容·····	245
实验说明·····	245
实验提示·····	245
实验步骤·····	246
实验 5 System Generator 的设计实例 ·····	251
实验目的·····	251
实验内容·····	251
实验说明·····	251
积分实验步骤·····	251
VHDL 步骤 ·····	254
Verilog HDL 步骤 ·····	256
附录 E IC 和 FPGA 专业术语的中英文对照·····	258
附录 F Xilinx FPGA 支持的 IP 列表 ·····	261
参考资料·····	271

第 1 章 可编程逻辑器件与 EDA 技术

1.1 可编程逻辑器件及其特征

当今社会是数字化的社会,是数字集成电路广泛应用的社会。数字集成电路不断进行更新换代,由早期的电子管、晶体管、中小规模集成电路,发展到超大规模集成电路(VLSI, Very Large Scale Integrated Circuit)以及许多具有特定功能的专用集成电路(ASIC, Application Specific Integrated Circuit)。专用集成电路就是为专门的产品或应用而设计的芯片。随着微电子技术的发展,设计与制造集成电路的任务已不完全由半导体厂商来独立承担,设计师们更希望在研究室里就能设计出可用的集成电路芯片。因此出现了各种可编程逻辑器件。

除了全定制的专用集成电路之外,目前有 5 种半定制和定制的器件可供实现专用集成电路的要求,它们是:

- (1)简单可编程逻辑器件(PLD, Programmable Logic Device);
- (2)复杂可编程逻辑器件(CPLD, Complex Programmable Logic Device);
- (3)现场可编程门阵列(FPGA, Field Programmable Gate Array);
- (4)门阵列(Gate Array);
- (5)标准单元(Standard Cell)。

在这些器件中,尤其是前三种器件的出现,使得电子系统的设计师利用与器件相应的计算机辅助设计(CAD, Computer Aided Design)软件,在研究室里就可以设计出自己的 ASIC,其中 FPGA 的应用尤其广泛。下面分别简要介绍各种半定制和定制的器件的结构。

1.1.1 简单 PLD

人们通常说的 PLD,就是指简单可编程逻辑器件。之所以简单,就是因为这种 PLD 是由简单的逻辑阵列宏单元组成的。逻辑阵列是 PLD 用户可编程的部分,它由与门、或门和反相器组成。输入信号是在逻辑阵列间的布线信道上运行的。一个简单的逻辑阵列宏单元的例子就是能可编程输出极性的宏单元:只要加一额外的熔丝到逻辑阵列的每个输出端就可实现,用户通过对这个极性熔丝编程来配置输出信号的极性。下面简要介绍一下 PLD 的逻辑阵列结构。

在 PLD 的技术实现上,输入到 PLD 的信号必须首先通过一个与门阵列,在这里形成输入信号的组合,每组相与的组合被称为布尔代数的子项或 PLD 术语中的乘积线,这个乘积线在第二个或门阵列中被相加。所以 PLD 实际上是由两个逻辑门组成的阵列构成的,即一个与阵列和一个或阵列。注意到 PLD 中的布线是通过阵列来提供的,影响一般门阵列的布局布线问题就不会影响 PLD,PLD 中的布线布通率是 100%。

基于这种简单的与或阵列结构的 PLD 有 4 种,按哪个阵列和输出是否可编程可分为:

- (1)可编程只读存储器 PROM 与阵列固定,或阵列可编程,输出不可编程。
- (2)可编程阵列逻辑 PAL 与阵列可编程,或阵列固定,输出可编程。
- (3)可编程逻辑阵列 PLA 与、或两个阵列都可编程,输出不可编程。

(4)通用阵列逻辑 GAL 与阵列可编程,或阵列可编程,输出宏单元可编程。

1.1.2 CPLD

CPLD 是指结构比较复杂的可编程逻辑器件,它包括下述输出宏单元结构。

- (1)可编程 I/O 允许用户对这些引脚编程,作为输入或输出。
- (2)寄存器输出和反馈 可用于实现计数器和移位寄存器等。
- (3)异或门输出结构 可用于一般用户多功能计数,能十分有效地建立大的计数器。

AMD 公司最先生产带有宏单元的可编程逻辑器件 PAL22V10。目前 PAL22V10 已成为划分 PLD 的界限。可编程逻辑器件所包含的门数大于 PAL22V10 所包含的门数,就被认为是复杂可编程逻辑器件,即 CPLD。

可以认为 CPLD 基本上是原来的可编程逻辑器件的扩展。它常常由可编程逻辑的功能块围绕一个位于中心、时延固定的可编程互连矩阵构成。由于用固定长度的金属线实现逻辑单元之间的互连,而可编程逻辑单元又是类似 PAL 的与阵列,使得 CPLD 与 FPGA 相比较很容易计算输入到输出的传输延迟,显然也会有一些灵活性的限制。但是,CPLD 的设计比 FPGA 简单。

虽然 CPLD 的结构一般都很相似,而且一般认为 CPLD 都有 100%的布通率,但是由于可编程互连矩阵的结构不同,实际上也会有差别。这里不进行进一步分析,请联系 CPLD 厂商获取详细信息。

1.1.3 FPGA

和其他可编程逻辑器件一样,FPGA 也由未完成的逻辑阵列所组成,通过将这些逻辑阵列连接到一起来完成一定的功能。像 PAL 一样,各个阵列单元之间的互连线是可以编程的。下面介绍 FPGA 的发展过程。

1985 年,Xilinx 公司推出了世界上第一款 FPGA。推出的这款 FPGA 包括两个器件和支持布局布线的设计工具。FPGA 的发展非常快,在不到 10 年的时间里,时钟频率就从不到 10MHz 提高到 100MHz。设计规则已经达到亚微米级别,FPGA 芯片的规模也从几千门增加到 2 万多等效门^①。大量功能强大易用的软件工具也相继推出,使得 FPGA 很快占领了电子设计领域很大的一块市场。

20 世纪 80 年代推出的 FPGA 可以说是 Intel 公司于 1971 年推出第一款商用微处理器的延续。那个时期,典型的微处理器系统包含微处理器、存储器和一些特殊功能的中小规模(MSI/SSI)器件。为追求更佳的性能、更小的尺寸、更低的成本、更快的错误恢复能力、高可靠性以及更快更易使用的原型,集成电路的设计者都意识到一定会有一种器件要取代当时的中小规模电路。这个概念的第一个尝试是 Signetics 公司于 1975 年推出的 83S100 FPLA(现场可编程逻辑阵列)。这款可编程器件实际上是一款 PLA 结构的器件。它由 16 个输入、48 个乘积项与阵列、8 个输出、48 个乘积项或阵列构成,通过 Ni-Cr 熔丝实现连线的断开或连接。这种方法在以降低速度和增加功耗为代价的前提下,给了设计师们很大的电路设计空间。但是这款可编程器件需要人工来设置熔丝的断或合,因此很复杂而且容易出错。

鉴于当时各种可编程器件的速度、结构等都不能很好地满足市场的需求,Xilinx 于 1985

^① 有关等效门的概念见 9.2.3 节。

年推出了 2000 系列的 FPGA。该系列的 FPGA 是世界上第一款基于 SRAM 的可编程 FPGA, 包括两个器件: 第一个器件由 8×8 (共 64 个) 的可配置逻辑模块 (CLB, Configurable Logic Block) 构成, 并在芯片的周边提供了 58 个输入输出接口模块 (IOB, I/O Block); 第二个器件由 10×10 的 CLB 构成, 并提供了总共 74 个 IOB 单元。

自 Xilinx 推出第一款 FPGA 之后, 世界上的其他公司也相继推出各自的 FPGA 产品。如 Actel 推出了很有特色的反熔丝 (Anti-Fused) FPGA。FPGA 市场内的竞争也愈演愈烈, IC 的制造商都意识到必须提供更加强大更易使用的新产品才能占有市场。在这种形势下, Xilinx 在 1987 年就推出了它的第二款 3000 系列的 FPGA, 距第一款 FPGA 的推出只有 2 年的时间。也就是在那时, AT&T 成功地获得这款 FPGA 器件的设计使用权, 并开始提供自己的芯片和开发系统, 即 AT&T 3000 系列 FPGA。

自从第二代 FPGA 问世以来, 各种 FPGA 的应用开始层出不穷, 电路复杂度也相继上升。这时, Xilinx 就开始研制第三代 FPGA 产品, AT&T 也开始开发自己的下一代 FPGA。Xilinx 的第三代 FPGA 产品于 1991 年问世, 而 AT&T 的下一代产品直到 1992 年才研制成功。认识到 FPGA 市场潜在的广阔空间, 很多 IC 和软件厂商也开始向 FPGA 领域进军, 包括一些著名的公司, 如 Actel、AMD、Altera、Intel、Mento Graphics、Texas Instruments 以及 Toshiba。

Xilinx 公司的成功不仅仅得益于它的硬件产品, 软件也是很重要的一个砝码。更快、更智能、易用的原理图编辑、设计实现和验证工具始终都是 Xilinx 在 FPGA 领域内成功的一个法宝。

为了能让不太熟悉 FPGA 的读者能够对 FPGA 有一个整体印象, 这里以 Xilinx 4000 系列的 FPGA 为例简单说明它的设计能力。XC4025 包含大约 1024 个 CLB, 它们按 32×32 的矩阵形式排列在 FPGA 芯片里, 这相当于 25 000 个等效门。这款 FPGA 包含 422Kbit 的 RAM, 主要用于编程。一个 CLB 的运行频率可达 250MHz, 但如果将互连线网络引入的延迟以及像加法器这样更复杂的逻辑考虑进去, 还可以获得 20 ~ 50MHz 的时钟频率。直观地讲, 加法器这样的逻辑是由大量的 CLB 来构成的, 例如一个 32bit 的加法器要用掉 62 个 CLB。

1.1.4 门阵列

门阵列属于半定制的集成电路, 可分为有信道和无信道两种。

有信道门阵列是在一个芯片上把门排列成阵列形式, 严格地讲是把单元 (含有若干个器件) 排列成阵列形式。单元被排列成行, 行与行之间留有作为连线用的信道区, 信道的宽度是固定的。这就是“有信道门阵列”这一名称的由来。为了保证单元之间的布线具有 100% 的布通率, 需要有较宽的信道, 但这样会导致无用的走线区域, 因而浪费了硅面积。

为了不大量浪费硅面积以及克服常规门阵列常有的门利用率低的缺点, 1982 年提出了门海的概念, 也就是指无信道门阵列。门海技术并无实现确定的布线信道区, 宏单元之间的连线将在无用的有效器件区上进行。门海技术提高了电路的集成密度, 在一个晶片 (Die) 上可以集成超过 10 万个门。在 20 世纪末, 在一个晶片上已经能够设计门延迟为 0.4ns 的 50 ~ 70 万门的电路。

1.1.5 标准单元法

标准单元法属于定制设计方法, 也是库单元设计方法的一种。库单元设计方法的一个基本思想就是通过重用 (Reuse) 一定的库单元来降低设计的成本, 这样对某种工艺, 库单元只需要设计验证一次, 就可以多次应用。缺点是用户无法精细 (Refine) 设计, 因为用户看到的只是

一些可重用的库单元。标准单元法提供的库包含各种扇入扇出的逻辑门,除了基本的逻辑功能外,还包括反相器、与/与非门、或/或非门、异或/异或非门以及触发器,而且通常也包括一些复杂的功能实现,如与-或-非门、多路复用器、全加器、比较器、计数器、解码器和编码器。用标准单元法设计的电路的原理图只由这些库单元组成。

显然标准单元法提供的库单元毕竟有限,而且设计的电路也可能不是最优的,因此出现了另一种库单元设计方法,即编译单元(Compiled Cell)法。编译单元法允许对实现各种电路模块的晶体管尺寸进行编译优化。和标准单元法不一样,编译单元法中的每个单元可根据最合理的情况单独进行版图设计,因而能够获得最佳的性能。

以上简要介绍了包括半定制和定制在内的各种电路设计方法。它们不像全定制设计方法那样可以针对每个晶体管进行电路参数和版图优化,并达到最快的速度、最低的功耗以及最小的芯片面积,但是由于它们的方便易用,还是受到了人们的广泛欢迎。可以说全定制的设计方法比半定制方法的专业知识要求高,而半定制又比定制的要求高。尤其是 FPGA 的出现,改变了整个芯片设计市场的划分,芯片设计也跨入一个新的领域。

1.2 FPGA 的应用及设计要求

1.2.1 FPGA 的应用

在使用 PLD 或者其他各种中小规模芯片的场合,都可使用 FPGA。下面简要列举一些 FPGA 可能应用的领域。

1. 专用集成电路(ASIC)

FPGA 是实现各种逻辑电路的一个完整的途径,尤其适合于 ASIC 的实现。一些应用 FPGA 设计的例子有 1Mbit 的 FIFO 控制器、IBM PS/2 的微信道接口、带纠错的 DRAM 控制器、打印机控制器、图形引擎、T1 网络收发器以及其他电信方面的各种应用。

2. 随机逻辑(Random Logic)电路

以前,随机逻辑电路通常采用 PAL 来实现。虽然 PAL 通常比 FPGA 速度要快,但是如果对电路的速度要求不是很高或者不是很关键,这样的电路也可以用 FPGA 来实现。因为 1 片 FPGA 实现的电路要用约 20 片 PAL 才能实现,因此采用 FPGA 的好处是提高了系统的可靠性,系统架构也更加紧凑。

3. 抛弃小规模电路芯片

由很多片小规模电路芯片构成的电路功能都能在 1 片 FPGA 中实现,节省了电路板的面积,提高了系统可靠性。

4. 原型(Prototyping)

FPGA 可能是进行原型设计最理想的载体了。实现成本低、开发周期短等等好处使得 FPGA 完全能够取代传统的原型硬件设计方法。原型机的最初框架和实现可以通过 FPGA 来验证,而且,随着项目进行的逐渐深入,在原型设计上的任何修改都是非常容易并且快速的。

5. 基于 FPGA 的计算引擎

一种全新的计算机通过在电路板上就可重新配置的 FPGA 来实现。这种计算机由很多插有 FPGA 的电路板构成,通常 FPGA 之间也是通过铜线连通的。基本的思想是软件或程序都可以通过“编译”的方式(使用高级的综合技术或人工完成)转化成硬件实现。然后再把这些硬件实现下载到 FPGA 中。这样会带来两个好处:第一,不存在传统微处理器不可避免的取指令时间,因为 FPGA 直接就实现了指令,程序的运行速度可以提高 100 倍;第二,FPGA 本身就具有并行处理的能力,这样又能提高软件的运行速度。

Quickturn 公司就开发了这样的一种系统用来进行大规模的电路仿真。Algotronix 公司也为 IBM PC 提供这样的电路板以支持这样的功能。巴黎的 DEC 公司也应用这样的思想设计了每秒完成 250 次到 2 640 次操作的系统,用于 RSA 密码学、离散余弦变换、Ziv-Lempel 编码、二维卷积等。

6. 可重配置的硬件

使用 FPGA 构成的机器的具体功能在使用过程中也可以改变,这也是 FPGA 吸引人的原因之一。一个例子是,在远地的计算机设备由于某个设计错误可能需要修改,如果该设备由 FPGA 来实现,那么这种修改就可以很容易地完成。对于一块采用了几片 FPGA 的电路板而言,如果 FPGA 之间的连线也是由可编程网络构成的,那么通过配置这个可编程网络和 FPGA 来修改电路板的行为功能就变得容易了。但要求这样的 FPGA 是可以重新配置的。

1.2.2 FPGA 设计要求

什么样的设计才算是优秀的 FPGA 设计呢?简单地说,就是优秀的 FPGA 设计与系统需求文档(SRD, System Requirement Document)和客户的规范保持一致。但这句话不能简单地理解为完成一个优秀的 FPGA 设计仅仅要做的就是达到客户和系统的基本需求。除了必须达到这些规范和需求外,可读性、可重复性和可测性三个方面也很重要。只有满足了上述的所有特征和要求,一个 FPGA 设计才算真正地完成,才是优秀的 FPGA 设计。下面展开说明优秀的 FPGA 设计的这三个重要的特征。

1. 可读性

可读性好的 FPGA 设计的原理图和硬件描述语言设计应该包含有足够详细的注释。很显然,每张原理图之间的关系以及硬件描述的模块之间的互连关系的说明是很重要的,但是各个模块本身的说明也是不能忽视的。例如,状态机的文档应当包含状态图或功能描述。布尔方程的实现过程也应该写在文档中,甚至应当写在源代码里面,包括简化前的或简化后的布尔方程。FPGA 的设计文档也应该包含用户自己创建的约束文件,还应该说明在设计、实现和验证阶段使用的各个输出文件。例如在综合后,应当说明网表文件的硬件描述语言类型、目的等。贝尔实验室以及其他大的研究机构都发现,写好文档也许花不了很多的时间,但是却可以在调试、测试和维护设计上节省大量的时间。写好设计的文档的另一个好处是一个已经经过验证的电路很容易地就能重用,为开发节省时间。

2. 可重复性

可重复性指 FPGA 设计应该保证如果不同的人从不同部位开始,并重新进行布局布线等,应该得到同样的结果。没有这个保证,验证以及其他形式的设计测试就毫无价值。因为设计师显然不希望在设计里出现这样的情况,器件具有相同的输入输出管脚和功能,但是由于布局布线的差异,最后时序却不一样。但是如果在实现的过程中,没有让系统设计软件参数或选项保持一致,这种情况就会发生。因此 FPGA 的文档就应包括必要的信息,即软件开发系统的版本号、软件的各个选项及参数设置。

关于 FPGA 设计的可重复性,有两点应该提到:一是 16 位的随机数种子,二是布局布线编辑情况。随机数种子是一个由系统时钟生成的 16 位数,用来初始化自动布局布线进程(APR, Automatic Place and Route)。如果在执行 APR 过程前没有指定这个随机数种子,那么每次运行 APR 就会得到不同的结果。同样,在 APR 之后,可能需要人工进行修改或完善,这些人工修改的过程或参数都应该以文档的方式记录下来,包括布局布线编辑器的选项和参数设置。如果不这样做,最终的实现就会因人而异,整个系统的性能也变得不稳定,甚至无法评估。

3. 可测性

可测性是优秀的 FPGA 设计的最后一个特征。FPGA 的开发商以及系统工程师都应该考虑这个因素。从 FPGA 厂商的角度来看,由于 FPGA 是基于静态 RAM 来研制的,加上专为测试而开发的专用功能,通常可以保证 FPGA 本身 100% 的故障覆盖率。系统级的测试要求工程师对整个设计流程以及系统架构都要很清楚,由于这不属于本书的覆盖范围,就不再赘述。

1.3 EDA 技术简介及软件

本节专门介绍 EDA(电子设计自动化, Electronic Design Automation)技术的发展以及 FPGA 领域常用的 EDA 工具。

1.3.1 EDA 技术

在过去的三十多年里,计算机技术迅猛发展,也给 EDA 行业带来了巨大的变化。CAD 程序在最开始用来设计批输入、针式打印机的输出,形成了 CAD 工具的基础。后来随着多任务操作系统和 PC 的出现,软件工具的概念被引入到电子系统设计中,从 20 世纪 80 年代早期开始,就具备了 EDA 的雏形。进入 20 世纪 90 年代后,电子系统已经从电路板级系统集成发展成为包括 ASIC、FPGA 和嵌入系统的多种模式。全球的 IC 的 EDA 市场已经增长到 30 亿美元,可以说 EDA 产业已经成为电子信息类产品的支柱产业。EDA 的蓬勃发展离不开设计方法学的进步。下面回顾一下过去几十年内 IC 设计方法学的演进,帮助不太熟悉 EDA 的读者建立 EDA 的概念。

(1) 手工设计(Hand Design) 手工设计的目标是要进行组合逻辑的优化和化简。手工设计是 20 世纪 60 年代以前主要的电路设计方法。

(2) 电路仿真(Circuit Simulation) 这里的电路指 IC 中的晶体管电路。在 20 世纪 70 年代早期,像 Calma 这样的公司就把数字化系统推向了市场,并采用 Spice 来进行电路仿真。Applicon、VC 和 RacalRedac 等公司也纷纷开发了印刷电路板的自动布线工具。IBM 和 WE 等公司也开发