

Xilinx 公司推荐 FPGA/CPLD 培训教材



# FPGA/CPLD 设计工具

# Xilinx ISE 5.x 使用详解

EDA 先锋工作室  
王诚 薛小刚 钟信潮 编著

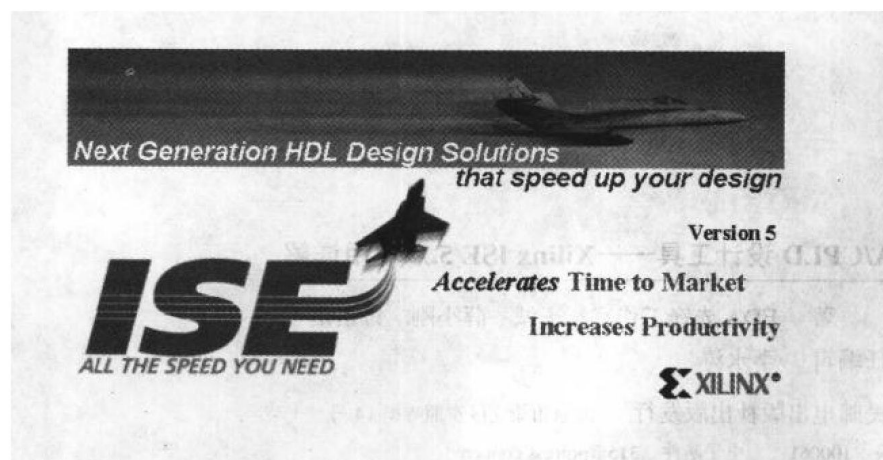
人民邮电出版社  
POSTS & TELECOM PRESS



# FPGA/CPLD 设计工具

## ——Xilinx ISE 5.x 使用详解

王 诚  
EDA 先锋工作室 薛小刚 编著  
钟信潮



人民邮电出版社

## 图书在版编目 (CIP) 数据

FPGA / CPLD 设计工具: Xilinx ISE 5.x 使用详解 / EDA 先锋工作室编著.

—北京: 人民邮电出版社, 2003.6

ISBN 7-115-11266-5

I. F... II. E... III. 可编程序逻辑器件—基本知识 IV. TP332.1

中国版本图书馆 CIP 数据核字 (2003) 第 036545 号

## 内 容 提 要

本书以 FPGA/CPLD 设计流程为主线, 阐述了如何合理利用 ISE 设计平台集成的各种设计工具, 高效地完成 FPGA/CPLD 的设计方法与技巧。全书在介绍 FPGA/CPLD 概念和设计流程的基础上, 依次论述工程管理与设计输入、仿真、综合、约束、实现与布局布线、配置调试等主要设计步骤在 ISE 集成环境中的实现方法与技巧。

本书立足工程实践, 结合作者多年工作经验, 选用大量典型实例, 并配有一定数量的练习题。本书配套光盘收录了所有实例的完整工程目录、源代码、详细操作步骤和使用说明, 利于读者边学边练, 提高实际应用能力。

本书可作为高等院校通信工程、电子工程、计算机、微电子与半导体学等专业的教材, 也可作为硬件工程师和 IC 工程师的实用工具书。

### **FPGA/CPLD 设计工具——Xilinx ISE 5.X 使用详解**

◆ 编 著 EDA 先锋工作室 王诚 薛小刚 钟信潮  
责任编辑 李永涛

◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号

邮编 100061 电子函件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

读者热线 010-67132692

北京汉魂图文设计有限公司制作

北京顺义振华印刷厂印刷

新华书店总店北京发行所经销

◆ 开本: 787×1092 1/16

印张: 24

字数: 585 千字

2003 年 6 月第 1 版

印数: 1-5 000 册

2003 年 6 月北京第 1 次印刷

ISBN 7-115-11266-5/TP · 3447

定价: 45.00 元 (附光盘)

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223



## EDA 先锋工作室

主 编: 王 诚

副主编: 薛小刚 钟信潮

编 委: 李 楠 伊贵业 吴义涛 张世卓  
王书松 侯小辉 于春华 寿开宇

# 序

美国赛灵思公司 (Xilinx) 是全球领先的可编程逻辑完整解决方案的供应商, 赛灵思研发、制造并销售应用范围广泛的高级集成电路、软件设计工具以及作为预定义系统级功能的 IP (Intellectual Property) 核。

赛灵思公司成立于 1984 年, 首创了现场可编程逻辑阵列 (FPGA) 这一创新性的技术, 并于 1985 年首次推出商业化产品。至今, 赛灵思公司已经推出从 Spartan 到 Virtex-II 等系列的 FPGA 产品以及 CoolRunner 等系列的 CPLD 产品, 满足了全世界对可编程逻辑 (PLD) 产品一半以上的需求, 同时还提供了互联网软件解决方案与核心解决方案等软件产品。ISE 是赛灵思公司最主要的设计软件之一, 目前已经推出的最新版本是 ISE 5.2i。该软件提供了可加强现有可编程设计流程, 并可适应客户特有设计方法的全面、丰富的设计选项。

在此, 我们郑重推荐由人民邮电出版社出版的《FPGA/CPLD 设计工具——Xilinx ISE 5.x 使用详解》, 希望通过本书使对在设计中应用 ISE 解决方案或对其感兴趣的读者能更深入的了解基于 ISE 进行 FPGA/CPLD 设计的基本原理、方法。也希望通过本书的编写, 能大力促进中国的数字系统现场集成技术的应用普及和推广。

XILINX ASIA PACIFIC

2003 年 4 月

# 关于本书

## 本书内容和特点

FPGA/CPLD 以其功能强大, 开发过程投资少、周期短, 可反复修改, 保密性能好, 开发工具智能化等特点成为当今硬件设计的首选方式之一。目前全国约有数百万的硬件工程师在自己的设计中运用着各种型号的 FPGA/CPLD。可以说 FPGA/CPLD 设计技术是当今合格硬件工程师与 IC 工程师的必备技能之一。

我国可编程逻辑器件设计技术落后于国外, 目前立足工程实践, 系统地介绍最新 FPGA/CPLD 设计工具的中文书籍较为贫乏。在这种情况下, 为了满足广大工科在校生了解业界流行的高效 FPGA/CPLD 设计技术的需要, 提高硬件工程师与 IC 工程师的工程实践技巧, 我们编写了本书。

全书以 FPGA/CPLD 设计流程为主线, 阐述了如何合理地利用 ISE 设计平台集成的各种设计工具, 高效地设计 FPGA/CPLD 的方法与技巧。

本书主要内容如下。

- 第 1 章 在介绍 FPGA/CPLD 基本原理与特点的基础上重点讲述 FPGA/CPLD 的设计开发流程, 并介绍 ISE 5.x 集成工具的基本功能。
- 第 2 章 介绍 ISE 5.x 中的工程管理方法与 6 种设计输入方式。
- 第 3 章 重点介绍 ModelSim 仿真工具的使用方法与技巧。
- 第 4 章 介绍 ISE 中集成的各种综合工具, 重点讲述业界最流行的高效综合工具 Synplify/Synplify Pro 的综合方法与技巧。
- 第 5 章 集中阐述约束的基本概念, 并介绍 ISE 集成的两个约束设计工具: 约束编辑器 (Constraints Editor) 和引脚与区域约束编辑器 (PACE) 的使用方法。
- 第 6 章 介绍 ISE 中集成的高级实现工具: 时序分析器 (Timing Analyzer)、底层布局器 (Floorplanner) 和 FPGA 底层编辑器 (FPGA Editor) 的使用方法。
- 第 7 章 介绍 ISE 中集成的 FPGA/CPLD 的调试工具: 功耗仿真器 (XPower)、配置工具 (iMPACT) 和在线逻辑分析仪 (ChipScope Pro) 的使用方法。
- 第 8 章 介绍大型复杂设计的推荐设计方法 Modular Design 设计方法。
- 第 9 章 利用一个综合实例来贯穿全书, 以提高读者的实际工作能力。

本书主要特点如下。

- 实用价值高: 本书的作者都有丰富的 FPGA/CPLD、数字 ASIC 设计经验, 本书立足于工程实践的需要, 对工程设计有显著的指导意义。
- 内容新颖: 本书的作者长期工作在可编程逻辑设计的最前沿, 与 EDA 软件制造商联系紧密, 所以有幸能够在第一时间使用最新版本的 FPGA/CPLD 设计工具。本书涉及的所有工具均根据最新资料撰写, 使本书介绍的内容很新颖。
- 剖析深刻: 本书对 FPGA/CPLD 设计的基本原理、方法有较为详尽的论述。对 ISE 集成的各种设计工具的介绍并不局限于操作方法, 而是结合作者多年的工作经验与心得, 从较深的层面对各个工具的特点进行了较为详细的剖析。

## 读者对象

本书可作为高等院校通信工程、电子工程、计算机、微电子与半导体学等理工专业的教材，也可作为硬件工程师和 IC 工程师的实用工具书。

## 本书附盘内容

本书配套光盘提供了书中所有示例的完整工程文件、设计源文件和说明文件。示例按照章节编号和出现的先后顺序排列，例如“Example-2-3”表示第 2 章的第 3 个示例。

每个工程示例都包括了该工程的项目文件、源文件、报告文件和生成结果等文件，读者可以用 ISE 5.x 或相应的软件直接打开。设计源文件根据设计输入类型分为源代码或原理图等。请读者将设计源文件拷贝到计算机硬盘上，并按照书中的操作步骤自行操作练习。示例说明文件包含了示例的详细信息和操作指南。配套光盘中的“Xilinx\_Lib.exe”文件是供 ModelSim 仿真器使用的 Xilinx 器件库自解压文件，在使用 ModelSim 仿真器时使用，具体使用方法参见书中 9.3.2 小节介绍。

## 本书约定

为了方便读者阅读，书中所有命令中的关键字采用正常体，非关键字采用斜体，需要在命令行输入的内容采用了黑体。并且书中设计了 4 个小图标，它们代表的含义如下。

 行家指点：用于介绍使用经验和心得，或罗列重要的概念。

 注意事项：用于提醒读者应该注意的问题。

 多学一招：用于介绍实现同一功能的不同方法。

 操作实例：用于引出一个操作题目和相应的一组操作步骤。

本书第 1、2、4、8、9 章由王诚编写，第 3、6 章及第 4 章第 2 节由薛小刚编写，第 5、7 章由钟信潮编写。

Xilinx 公司亚太地区软件经理邹志雄、资深应用工程师刘晖对全书进行了审校，Xilinx 公司亚太地区公关经理柯春玲，Hoffman 公司高级顾问刘涛，华创科技有限公司资深应用工程师李焰等对本书编写提出了很多建设性的意见，并给予作者多方面的帮助，在此表示衷心的感谢。本书还得到了其他很多 EDA 公司的领导和工程师的帮助，在此一并表示衷心的感谢。

感谢您选择了本书，如果您对书中内容有任何困惑和建议，请与我们联系。

电子函件：book@edacn.com（作者），liyongtao@ptpress.com.cn（责任编辑）。

如果您需要得到 Xilinx 更全面的服务与技术支持，请访问 <http://support.xilinx.com/china>。

EDA 先锋工作室

2003 年 4 月

# 目 录

|                                          |    |
|------------------------------------------|----|
| 第 1 章 ISE 系统简介.....                      | 1  |
| 1.1 FPGA/CPLD 简介.....                    | 1  |
| 1.1.1 FPGA/CPLD 的基本原理.....               | 2  |
| 1.1.2 FPGA/CPLD 的特点.....                 | 2  |
| 1.2 FPGA/CPLD 的设计流程.....                 | 3  |
| 1.3 ISE 系列产品的新特点.....                    | 5  |
| 1.3.1 特点综述.....                          | 5  |
| 1.3.2 ISE 5.x 的新增特性.....                 | 6  |
| 1.4 ISE 5.x 支持的器件.....                   | 8  |
| 1.5 ISE 5.x 的系统配置与安装.....                | 9  |
| 1.5.1 推荐的系统配置.....                       | 9  |
| 1.5.2 ISE 5.x 的安装.....                   | 10 |
| 1.6 ISE 5.x 的集成工具及其基本功能.....             | 12 |
| 1.7 常用专有名词解释.....                        | 15 |
| 1.8 小结.....                              | 17 |
| 1.9 问题与思考.....                           | 18 |
| 第 2 章 工程管理器与设计输入工具.....                  | 19 |
| 2.1 ISE 工程管理器——Project Navigator.....    | 19 |
| 2.1.1 Project Navigator 综述.....          | 19 |
| 2.1.2 Project Navigator 的用户界面.....       | 20 |
| 2.1.3 使用 Project Navigator 创建并管理工程.....  | 27 |
| 2.2 HDL 语言的输入工具——HDL Editor.....         | 32 |
| 2.2.1 HDL Editor 综述.....                 | 32 |
| 2.2.2 源代码输入的好助手——Language Templates..... | 32 |
| 2.3 状态机输入工具——StateCAD.....               | 34 |
| 2.3.1 StateCAD 综述.....                   | 34 |
| 2.3.2 StateCAD 的用户界面.....                | 35 |
| 2.3.3 使用 StateCAD 设计状态机.....             | 40 |
| 2.4 原理图输入工具——ECS.....                    | 51 |
| 2.4.1 ECS 综述.....                        | 52 |
| 2.4.2 ECS 的用户界面.....                     | 52 |
| 2.4.3 使用 ECS 完成原理图输入设计.....              | 55 |
| 2.4.4 使用 ECS 进行混合设计的方法.....              | 62 |
| 2.5 IP 核生成工具——CORE Generator.....        | 63 |



|              |                                       |            |
|--------------|---------------------------------------|------------|
| 2.5.1        | CORE Generator 综述.....                | 63         |
| 2.5.2        | CORE Generator 的用户界面.....             | 64         |
| 2.5.3        | 使用 CORE Generator 生成 IP 核的方法与技巧.....  | 67         |
| 2.6          | 测试激励生成器——HDL Bencher.....             | 72         |
| 2.6.1        | HDL Bencher 综述.....                   | 72         |
| 2.6.2        | 使用 HDL Bencher 生成测试激励.....            | 74         |
| 2.7          | 设计结构向导——Architecture Wizard.....      | 80         |
| 2.7.1        | Architecture Wizard 综述.....           | 80         |
| 2.7.2        | Architecture Wizard 使用方法.....         | 80         |
| 2.8          | 小结.....                               | 83         |
| 2.9          | 问题与思考.....                            | 83         |
| <b>第 3 章</b> | <b>ModelSim 仿真工具.....</b>             | <b>85</b>  |
| 3.1          | ModelSim 的用户接口.....                   | 86         |
| 3.2          | ModelSim 仿真窗口综述.....                  | 88         |
| 3.3          | 仿真环境的建立.....                          | 94         |
| 3.3.1        | 仿真库的命名.....                           | 95         |
| 3.3.2        | 仿真库文件的手动建立.....                       | 96         |
| 3.4          | 一个简单的仿真示例.....                        | 97         |
| 3.4.1        | 在 ModelSim 环境下进行仿真.....               | 97         |
| 3.4.2        | 在 ISE 集成环境中进行仿真.....                  | 100        |
| 3.5          | 混合 VHDL/Verilog 仿真.....               | 102        |
| 3.6          | ModelSim 中的调试方法.....                  | 104        |
| 3.7          | 使用批处理方式进行仿真.....                      | 107        |
| 3.8          | 波形比较与 WLF 文件.....                     | 109        |
| 3.8.1        | 创建一个参考的数据集合.....                      | 109        |
| 3.8.2        | 修改源文件重新运行仿真.....                      | 110        |
| 3.8.3        | 进行波形比较.....                           | 110        |
| 3.9          | SDF 时序标注.....                         | 115        |
| 3.10         | 仿真中的代码覆盖率.....                        | 116        |
| 3.11         | VCD 文件.....                           | 118        |
| 3.11.1       | 创建 VCD 文件.....                        | 118        |
| 3.11.2       | 使用一个 VCD 文件重新进行仿真.....                | 119        |
| 3.12         | 问题与思考.....                            | 120        |
| <b>第 4 章</b> | <b>ISE 中集成的综合工具.....</b>              | <b>121</b> |
| 4.1          | 新兴的高效综合工具——Synplify/Synplify Pro..... | 121        |
| 4.1.1        | Synplify/Synplify Pro 的功能与特点.....     | 121        |
| 4.1.2        | Synplify Pro 的用户界面.....               | 126        |

|              |                                     |            |
|--------------|-------------------------------------|------------|
| 4.1.3        | Synplify Pro 综合流程 .....             | 130        |
| 4.1.4        | Synplify Pro 的其他综合技巧 .....          | 146        |
| 4.2          | Xilinx 最早的合作伙伴——Synopsys 综合工具 ..... | 158        |
| 4.2.1        | 设计流程 .....                          | 159        |
| 4.2.2        | FE 综合优化过程 .....                     | 161        |
| 4.2.3        | FST 操作说明 .....                      | 170        |
| 4.3          | Xilinx 内嵌的综合工具——XST .....           | 173        |
| 4.3.1        | XST 综述 .....                        | 173        |
| 4.3.2        | XST 综合属性设置 .....                    | 173        |
| 4.3.3        | 使用 XST 的综合流程 .....                  | 178        |
| 4.4          | 全局时钟与第二全局时钟资源 .....                 | 182        |
| 4.4.1        | 全局时钟资源简介 .....                      | 182        |
| 4.4.2        | 常用的与全局时钟资源相关的 Xilinx 器件原语 .....     | 182        |
| 4.4.3        | Xilinx 全局时钟资源的使用方法 .....            | 185        |
| 4.4.4        | 使用 Xilinx 全局时钟资源的注意事项 .....         | 186        |
| 4.4.5        | 第二全局时钟资源 .....                      | 188        |
| 4.5          | 小结 .....                            | 189        |
| 4.6          | 问题与思考 .....                         | 189        |
| <b>第 5 章</b> | <b>约束 .....</b>                     | <b>191</b> |
| 5.1          | 概述 .....                            | 191        |
| 5.2          | 时序约束 .....                          | 193        |
| 5.2.1        | 周期约束 (PERIOD 约束) .....              | 193        |
| 5.2.2        | 偏移约束 (OFFSET 约束) .....              | 196        |
| 5.2.3        | 专门约束 .....                          | 199        |
| 5.3          | 分组约束 .....                          | 202        |
| 5.3.1        | TNM 约束 .....                        | 202        |
| 5.3.2        | TNM_NET 约束 .....                    | 205        |
| 5.3.3        | TIMEGRP 约束 .....                    | 205        |
| 5.3.4        | TPTHURU 约束 .....                    | 206        |
| 5.3.5        | TPSYNC 约束 .....                     | 206        |
| 5.4          | 约束编辑器——Constraints Editor .....     | 207        |
| 5.4.1        | Constraints Editor 的用户界面 .....      | 207        |
| 5.4.2        | 附加全局约束 .....                        | 208        |
| 5.4.3        | 附加端口约束 .....                        | 210        |
| 5.4.4        | 附加分组约束和时序约束 .....                   | 211        |
| 5.4.5        | 附加专用约束 .....                        | 214        |
| 5.5          | 引脚与区域约束编辑器——PACE .....              | 215        |
| 5.5.1        | PACE 的用户界面 .....                    | 216        |

|              |                                            |            |
|--------------|--------------------------------------------|------------|
| 5.5.2        | 附加区域约束 .....                               | 219        |
| 5.5.3        | 附加 I/O 引脚约束 .....                          | 220        |
| 5.6          | 约束文件 .....                                 | 221        |
| 5.6.1        | 约束文件的概念 .....                              | 221        |
| 5.6.2        | UCF、NCF 文件的基本语法规则 .....                    | 222        |
| 5.7          | 小结 .....                                   | 224        |
| 5.8          | 问题与思考 .....                                | 224        |
| <b>第 6 章</b> | <b>辅助设计工具 .....</b>                        | <b>225</b> |
| 6.1          | 时序分析器——Timing Analyzer .....               | 225        |
| 6.1.1        | 时序分析器的用户界面 .....                           | 226        |
| 6.1.2        | 时序分析器的作用及设计流程 .....                        | 227        |
| 6.1.3        | 基本时序路径 .....                               | 228        |
| 6.1.4        | 时序分析器的使用方法 .....                           | 236        |
| 6.2          | 布局规划器——Floorplanner .....                  | 240        |
| 6.2.1        | 布局规划器的用户界面 .....                           | 240        |
| 6.2.2        | 布局规划器的特点及作用 .....                          | 242        |
| 6.2.3        | 布局规划设计流程 .....                             | 243        |
| 6.2.4        | 设计示例 .....                                 | 246        |
| 6.3          | FPGA 底层编辑器——FPGA Editor .....              | 251        |
| 6.3.1        | FPGA 底层编辑器的用户接口 .....                      | 251        |
| 6.3.2        | FPGA 底层编辑器的作用 .....                        | 252        |
| 6.3.3        | FPGA 底层编辑器输入输出文件 .....                     | 254        |
| 6.3.4        | FPGA 底层编辑器的工作流程 .....                      | 255        |
| 6.3.5        | 使用 FPGA 底层编辑器的预备知识 .....                   | 255        |
| 6.3.6        | 设计示例 .....                                 | 257        |
| 6.4          | 小结 .....                                   | 262        |
| 6.5          | 问题与思考 .....                                | 262        |
| <b>第 7 章</b> | <b>XPower、iMPACT 和 ChipScope Pro .....</b> | <b>263</b> |
| 7.1          | XPower .....                               | 263        |
| 7.1.1        | XPower 综述 .....                            | 263        |
| 7.1.2        | XPower 的用户界面 .....                         | 264        |
| 7.1.3        | 用 XPower 分析功耗 .....                        | 267        |
| 7.2          | iMPACT .....                               | 269        |
| 7.2.1        | iMPACT 综述 .....                            | 269        |
| 7.2.2        | iMPACT 的用户界面 .....                         | 270        |
| 7.2.3        | 用 iMPACT 下载配置文件 .....                      | 273        |
| 7.3          | ChipScope Pro .....                        | 281        |

|              |                                               |            |
|--------------|-----------------------------------------------|------------|
| 7.3.1        | ChipScope Pro 综述 .....                        | 281        |
| 7.3.2        | ChipScope Pro Core Inserter .....             | 283        |
| 7.3.3        | ChipScope Pro Analyzer .....                  | 289        |
| 7.4          | 小结 .....                                      | 294        |
| 7.5          | 问题与思考 .....                                   | 294        |
| <b>第 8 章</b> | <b>模块化设计方法 .....</b>                          | <b>295</b> |
| 8.1          | 模块化设计方法的基本概念 .....                            | 295        |
| 8.2          | 模块化设计方法的设计流程 .....                            | 296        |
| 8.2.1        | Modular Design 的设计输入与综合步骤 .....               | 297        |
| 8.2.2        | Modular Design 的实现步骤 .....                    | 298        |
| 8.3          | 模块化设计方法的注意事项与设计技巧 .....                       | 302        |
| 8.3.1        | Modular Design 的目录管理 .....                    | 302        |
| 8.3.2        | Modular Design 的常用约束 .....                    | 303        |
| 8.3.3        | Modular Design 的设计规模与性能表现 .....               | 303        |
| 8.3.4        | Modular Design 的报告查看 .....                    | 304        |
| 8.3.5        | 使用 XFLOW 自动进行模块化设计 .....                      | 304        |
| 8.4          | 模块化设计方法的设计实例 .....                            | 306        |
| 8.5          | 小结 .....                                      | 315        |
| 8.6          | 问题与思考 .....                                   | 316        |
| <b>第 9 章</b> | <b>融会贯通——“运动计时表”设计 .....</b>                  | <b>317</b> |
| 9.1          | 示例背景 .....                                    | 318        |
| 9.2          | 多元混合设计输入方法 .....                              | 319        |
| 9.2.1        | 新建工程 “watch_sc” .....                         | 319        |
| 9.2.2        | 使用 ECS 绘制 “cnt60” 和 “outs3” 模块原理图 .....       | 320        |
| 9.2.3        | 使用 Core Generator 生成 “tenths” IP 核 .....      | 326        |
| 9.2.4        | 使用 StateCAD 设计 “stmach_v” 状态机 .....           | 329        |
| 9.2.5        | 使用 Architecture Wizard 生成时钟管理模块 “dcml” .....  | 337        |
| 9.2.6        | 使用语言模板设计 “hex2led” 和 “decode” 的 HDL 源代码 ..... | 339        |
| 9.2.7        | 使用 ECS 设计顶层原理图 .....                          | 342        |
| 9.3          | 测试激励与行为级功能仿真 .....                            | 343        |
| 9.3.1        | 使用 HDL Bencher 生成测试激励 .....                   | 344        |
| 9.3.2        | 调用 ModelSim 进行行为级功能仿真 .....                   | 346        |
| 9.4          | Synplify Pro 和 XST 综合方法 .....                 | 347        |
| 9.4.1        | 使用 XST 综合整个设计 .....                           | 348        |
| 9.4.2        | 使用 Synplify Pro 的特色工具分析、优化设计 .....            | 350        |
| 9.5          | 设计用户约束文件与实现结果的分析 .....                        | 356        |
| 9.5.1        | 使用 Constraints Editor 设计 UCF 文件 .....         | 356        |

|                                  |     |
|----------------------------------|-----|
| 9.5.2 使用 PACE 设计 UCF.....        | 359 |
| 9.5.3 实现步骤与实现结果分析 .....          | 361 |
| 9.6 使用 ModelSim 进行布线后仿真 .....    | 366 |
| 9.7 使用 iMPACT 配置 FPGA/CPLD ..... | 367 |
| 9.8 小结.....                      | 372 |
| 9.9 问题与思考.....                   | 372 |

# 第1章 ISE 系统简介

本章在介绍 FPGA/CPLD 基本理论的基础上，引入 FPGA/CPLD 的完整设计流程，重点介绍 Xilinx 的 ISE 集成开发环境的特点与 ISE 所集成的工具包的基本功能。

本章主要内容如下：

- FPGA/CPLD 简介；
- FPGA/CPLD 设计开发流程；
- ISE 5.x 系列产品的特点；
- ISE 5.x 支持的器件；
- ISE 的安装与配置；
- ISE 5.x 中集成的工具及其基本功能；
- 常用专有名词解释。

## 1.1 FPGA/CPLD 简介

在数字化、信息化的时代，数字集成电路应用得非常广泛。随着微电子技术与工艺的发展，数字集成电路从电子管、晶体管、中小规模集成电路、超大规模集成电路（VLSIC）逐步发展到今天的专用集成电路（ASIC）。ASIC 的出现降低了产品的生产成本，提高了系统的可靠性，减少了产品的物理尺寸，推动了社会的数字化进程。但是 ASIC 因其设计周期长，改版投资大，灵活性差等缺陷制约着它的应用范围。硬件工程师希望有一种更灵活的设计方法，根据需要，在实验室就能设计、更改大规模数字逻辑，研制自己的 ASIC 并马上投入使用。这就是可编程逻辑器件提出的基本思想。

可编程逻辑器件随着微电子制造工艺的发展取得了长足的进步。从早期的只能存储少量数据，完成简单逻辑功能的可编程只读存储器（PROM）、紫外线可擦除只读存储器（EPROM）和电可擦除只读存储器（EEPROM），发展到能完成中大规模的数字逻辑功能的可编程阵列逻辑（PAL）和通用阵列逻辑（GAL），今天已经发展成为可以完成超大规模的复杂组合逻辑与时序逻辑的现场可编程逻辑器件（FPGA）和复杂可编程逻辑器件（CPLD）。随着工艺技术的发展与市场需要，超大规模、高速、低功耗的新型 FPGA/CPLD 不断推陈出新。新一代的 FPGA 甚至集成了中央处理器（CPU）或数字处理器（DSP）内核，在一片 FPGA 上进行软硬件协同设计，为实现片上可编程系统（SOPC, System On Programmable Chip）提供了强大的硬件支持。

### 1.1.1 FPGA/CPLD 的基本原理

简化的 FPGA/CPLD 的结构由 4 部分组成：输入/输出模块、二维逻辑阵列模块、连线资源和内嵌存储器结构，如图 1-1 所示。输入/输出模块是芯片与外界接口，完成不同电气特性下的输入/输出功能要求；二维逻辑阵列模块是可编程逻辑的主体，可以根据设计灵活地改变连接与配置，完成不同的逻辑功能；连线资源连接所有的二维逻辑阵列模块和输入/输出模块，连线长度和工艺决定着信号在连线上的驱动能力和传输速度；内嵌存储器结构可以在芯片内部存储数据。

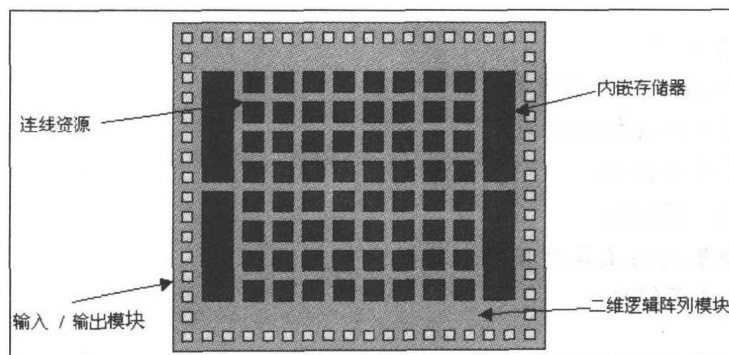


图1-1 可编程逻辑器件的结构原理图

FPGA 和 CPLD 的内部结构稍有不同，FPGA 的二维逻辑阵列模块是基于查找表（Look-Up-Table）结构的，如 Xilinx 的 Spartan、Virtex、Virtex-E、Virtex-II、Virtex-II Pro 系列和 ALTERA 的 FLEX、ACEX、APEX 系列等。而 CPLD 的二维逻辑阵列模块是基于乘积项（Product-Term）结构的，如 Xilinx 的 XC9500 系列（Flash 工艺），ALTERA 的 MAX7000（EEPROM 工艺）和 Lattice 的部分产品（EEPROM 工艺）。

通常，FPGA 中寄存器资源比较丰富，适合做同步时序电路较多的设计；CPLD 中组合逻辑资源比较丰富，适合做组合电路较多的设计。而且 FPGA 和 CPLD 在规模、价格及速度上略有差异。尽管 FPGA 与 CPLD 在某些方面有一些差异，但是对用户而言，FPGA 和 CPLD 的设计方法是相似的，使用 EDA 软件的设计过程也没有太大的差别。设计时，只需根据所选器件型号充分发挥器件的特性就可以了，所以下面多数情况下的论述，并未加以区分，而统称为可编程逻辑器件。

### 1.1.2 FPGA/CPLD 的特点

FPGA/CPLD 既继承了 ASIC 的大规模、高集成度、高可靠性的优点，又克服了普通 ASIC 设计周期长、投资大、灵活性差的缺点，逐步成为复杂数字硬件电路设计的理想首选。当代 FPGA/CPLD 有以下特点：

- 规模越来越大。随着 VLSI（Very Large Scale IC，超大规模集成电路）工艺的不断提高，单一芯片内部可以容纳上百万个晶体管，FPGA/CPLD 芯片的规模也越来越大。单片逻辑门数已愈百万，如 Xilinx Virtex-II xc2v 8000 已经达

到 800 万门的规模。芯片的规模越大所能实现的功能就越强，同时也更适于实现片上系统（SOC）。

- 开发过程投资小。FPGA / CPLD 芯片在出厂之前都做过百分之百的测试，而且 FPGA / CPLD 设计灵活，发现错误时可直接更改设计，减少了投片风险，节省了许多潜在的花费。所以不但许多复杂系统使用 FPGA / CPLD 完成，甚至设计 ASIC 也要把实现 FPGA / CPLD 功能样机作为必需的步骤。
- FPGA/CPLD 一般可以反复地编程、擦除。在不改变外围电路的情况下，设计不同片内逻辑就能实现不同的电路功能。所以，用 FPGA / CPLD 试制功能样机，能以最快的速度占领市场。甚至在有些领域，因为相关标准协议发展太快，设计 ASIC 可能跟不上技术的更新，只能用 FPGA / CPLD 完成系统的研制与开发。
- 保密性能好。在某些场合下，根据要求选用防止反向技术的 FPGA / CPLD，能很好地保护系统的安全性和设计者的知识产权。
- FPGA / CPLD 开发工具智能化，功能强大。现在，FPGA / CPLD 开发工具种类繁多、智能化高、功能强大。应用各种工具可以完成从输入、综合、实现到配置芯片等一系列功能。还有很多工具可以完成对设计的仿真、优化、约束、在线调试等功能。这些工具易学易用，可以使设计人员更能集中精力进行电路设计，快速将产品推向市场。
- 新型 FPGA 内嵌 CPU 或 DSP 内核，支持软硬件协同设计，可以作为片上可编程系统（SOPC）的硬件平台。

## 1.2 FPGA/CPLD 的设计流程

一个完整的 FPGA/CPLD 设计流程包括电路设计与输入、功能仿真、综合、综合后仿真、实现、布线后仿真和下板调试等主要步骤，如图 1-2 所示。

其中电路设计与输入是根据工程师的设计方法将所设计的功能描述给 EDA 软件。常用的设计输入方法有硬件描述语言（HDL）和原理图设计输入方法。原理图设计输入法在早期应用得比较广泛，它根据设计要求，选用器件、绘制原理图、完成输入过程。这种方法的优点是直观、便于理解、元件库资源丰富。但是在大型设计中，这种方法的维护性较差，不利于模块建设与重用。更主要的缺点是：当所选用芯片升级换代后，所有的原理图都要作相应的改动。目前进行大型工程设计时，最常用的设计方法是 HDL 设计输入法。其中影响最为广泛的 HDL 语言是 VHDL 和 Verilog HDL。它们的共同特点是利于由顶向下设计，利于模块的划分与复用，可移植性好，通用性好，设计不因芯片的工艺与结构的变化而变化，更利于向 ASIC 的移植。波形输入和状态机输入方法是两种常用的辅助设计输入方法。使用波形输入法时，只要绘制出激励波形和输出波形，EDA 软件就能自动地根据响应关系进行设计。而使用状态机输入法时，设计者只需画出状态转移图，EDA 软件就能生成相应的 HDL 代码或者原理图，使用十分方便。Xilinx ISE 工具包中的 StateCAD 就能完成状态机输入的功能。但是需要指出的是，后两种设计方法只能在某些特殊情况下缓解设计者的工作量，并不适合所有的设计。



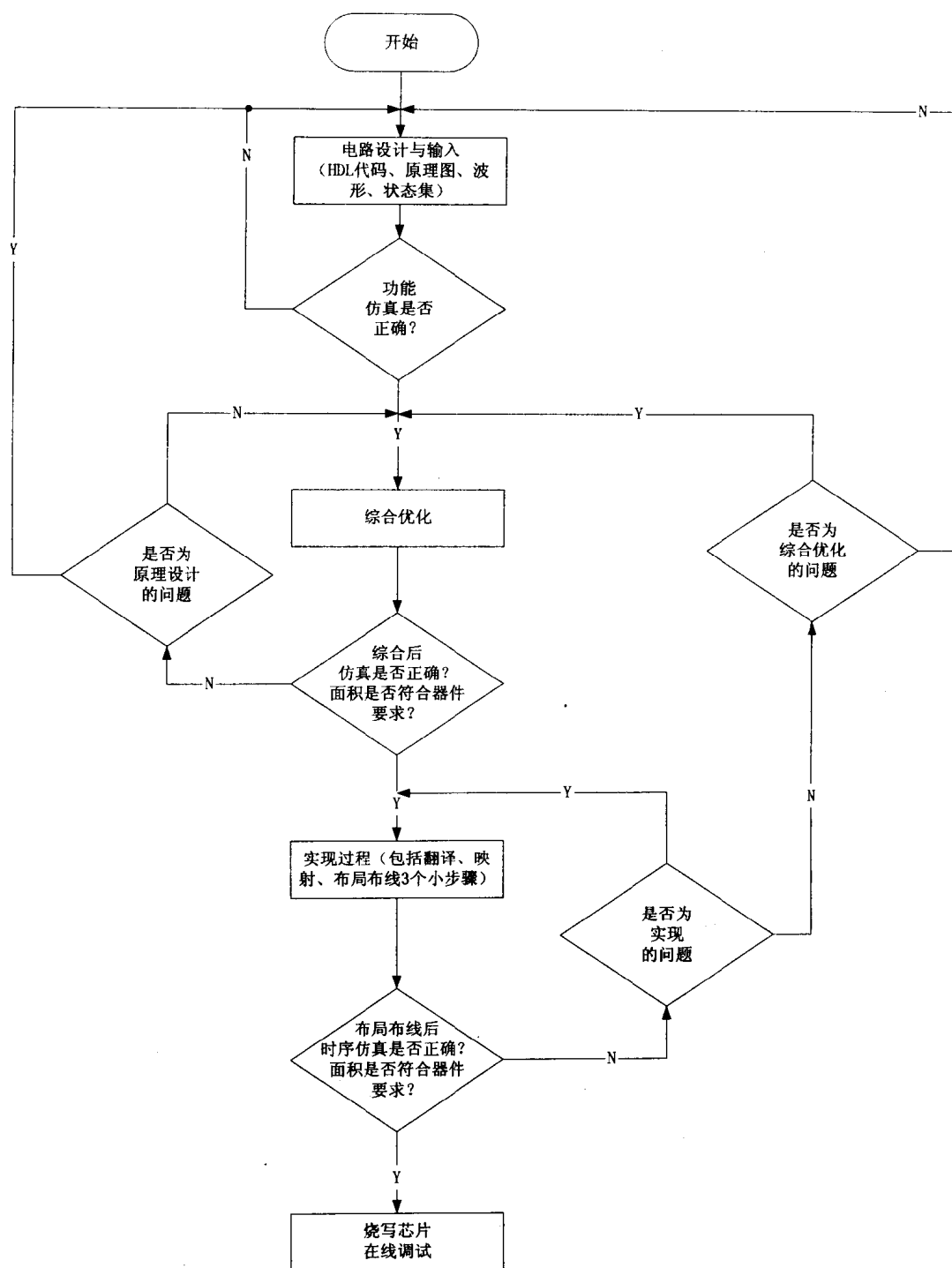


图1-2 完整的FPGA/CPLD设计流程

电路设计完成后, 要用专用的仿真工具对设计进行功能仿真, 验证电路功能是否符合设计要求。功能仿真有时也被称为前仿真。常用的仿真工具有 Model Tech 公司的 ModelSim, Synopsys 公司的 VCS, Cadence 公司的 NC-Verilog 和 NC-VHDL, Aldec 公司的 Active HDL