

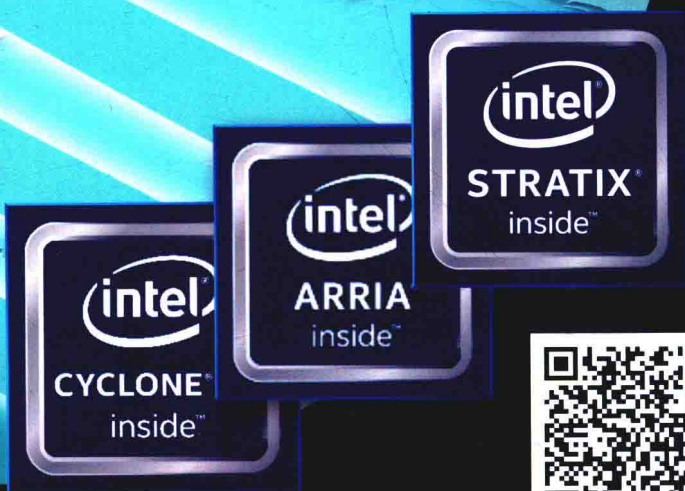
Intel 公司推荐 FPGA/CPLD 培训教材

Quartus Prime  
Design Suite

# Intel FPGA/CPLD 设计 (基础篇)

王欣 王江宏 蔡海宁 王诚 吴继华 编著

Intel 公司 审校



 中国工信出版集团

 人民邮电出版社  
POSTS & TELECOM PRESS

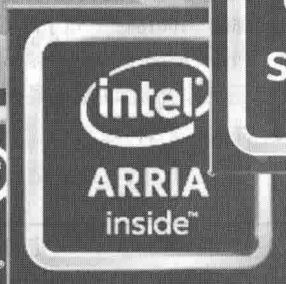
扫描二维码  
下载图书配套资源

Intel 公司推荐 FPGA/CPLD 培训教材

Quartus Prime  
Design Suite

# Intel FPGA/CPLD 设计 (基础篇)

王欣 王江宏 蔡海宁 王诚 吴继华 编著  
Intel 公司 审校



人民邮电出版社  
北京

## 图书在版编目(CIP)数据

Intel FPGA/CPLD设计. 基础篇 / 王欣等编著. —  
北京: 人民邮电出版社, 2017.9  
ISBN 978-7-115-46684-6

I. ①I… II. ①王… III. ①可编程序逻辑阵列—系  
统设计 IV. ①TP332.1

中国版本图书馆CIP数据核字(2017)第219141号

## 内 容 提 要

本书结合作者多年工作经验,系统地介绍了FPGA/CPLD的基本设计方法。在介绍FPGA/CPLD概念的基础上,介绍了Intel主流FPGA/CPLD的结构与特点,并通过丰富的实例讲解Quartus II与ModelSim、Synplify Pro等常用EDA工具的开发流程。

本书所有实例的完整工程、源代码和使用说明文件,都以云存储的方式存放在云端,读者可以通过扫描二维码的方式进行下载。

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

---

◆ 编 著 王 欣 王江宏 蔡海宁 王 诚 吴继华

审 校 Intel 公司

责任编辑 李永涛

责任印制 焦志炜

◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号

邮编 100164 电子邮件 315@ptpress.com.cn

网址 <http://www.ptpress.com.cn>

三河市海波印务有限公司印刷

◆ 开本: 787×1092 1/16

印张: 18.5

字数: 460千字

2017年9月第1版

2017年9月河北第1次印刷

---

定价: 59.00元

读者服务热线: (010)81055410 印装质量热线: (010)81055316

反盗版热线: (010)81055315

广告经营许可证: 京东工商广登字 20170147号

## 序

Altera 公司成立于 1983 年，后于 2016 年成为 Intel 公司可编程解决方案事业部 (Programmable Solutions Group)，总部位于美国硅谷，自从 1984 年发明世界上第一款可编程逻辑器件以来，一直为客户提供业界领先的定制逻辑解决方案。今天，分布在全球 20 多个地区的 3000 多名员工将为 12000 多家用户提供更巧妙的定制逻辑解决方案及广泛的技术支持，包括了 FPGA、SoC、CPLD 及电源管理产品等。

Intel 的全系列定制逻辑解决方案解决了很多系统级难题，包括性能、功耗、总体拥有成本、电路板面积、产品及时面市和设计团队效能等。很多不同行业的业界领先公司都采用了 Intel FPGA 产品，包括数据中心、通信、汽车、广播、工业、医疗、军事、测试和测量、消费类等行业。

Intel 公司可编程解决方案事业部的产品包括：

- Stratix 系列 FPGA 和 SoC (支持实现性能最好的系统，用于数据处理和算法加速)；
- Arria 系列 FPGA 和 SoC (以很低的功耗预算实现了高性能系统)；
- Cyclone 系列 FPGA 和 SoC (以最低功耗实现了需要较高性能的大批量应用系统)；
- MAX 系列 FPGA 和 CPLD (在大批量系统中实现了非易失、单芯片集成)；
- Enpirion 电源产品 (为 Intel FPGA、SoC 和 CPLD 提供支持，在集成产品中同时实现了高效、小外形封装、低噪声性能)；
- 软件开发工具、IP、参考设计和开发套件 (方便了 Intel FPGA、SoC 和 CPLD 的开发，保证了极高的设计团队效能)。

随着最终市场需求的发展，用户产品越来越复杂，我们的定制逻辑容量也随之快速增长。结果，设计工程师没有足够的设计方法和知识来满足需求的增长。设计人员在采用最新定制逻辑解决方案及 Intel FPGA 前沿产品和技术时，可以利用我们在中国本地开发的参考材料和指南。

我非常荣幸地向您推荐《Intel FPGA/CPLD 设计 (基础篇)》和《Intel FPGA/CPLD 设计 (高级篇)》。与前面版本图书相比，本版对内容进行了更新以反映 Intel 最新 FPGA 器件和设计工具。这两本书不仅介绍了传统 PLD 技术和设计技巧，而且还大量阐述了目前业界流行的硬浮点 DSP、片外高速存储器、HMC (Hybrid Memory Controller)、JESD204B 和片上系统 FPGA，以及 Intel 推出的适用于异构平台并通过 FPGA 做加速处理的 OpenCL 软件开发套件。

这两本书以独特的视角解释了设计方法，帮助您掌握高级 PLD 设计技巧，还介绍了 Intel FPGA 器件和 Quartus II 设计软件。这些书有丰富的设计实例，通过实际练习，能帮助您深入理解概念，养成良好的设计习惯。

希望您能够从这些优秀的书中受益，预祝您的可编程逻辑设计获得成功!

庄秉翰

副总裁

亚太区可编程解决方案事业部

Intel 公司

# 关于本书

## 内容和特点

FPGA/CPLD、GPU 和 CPU 被称为未来数字电路系统的 3 块基石，也是目前硬件设计研究的热点。与传统电路设计方法相比，FPGA/CPLD 具有功能强大，开发过程投资小、周期短，可反复编程修改，保密性能好，开发工具智能化等特点，特别是随着电子工艺的不断改进，低成本 FPGA/CPLD 器件推陈出新，这一切促使 FPGA/CPLD 成为当今硬件设计的首选方式之一。可以说 FPGA/CPLD 设计技术是当今高级硬件工程师与 IC 工程师的必备技能。

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缺乏。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《Altera FPGA/CPLD 设计（基础篇）》和《Altera FPGA/CPLD 设计（高级篇）》。这两本书出版以来，广受读者好评，但随着技术的不断发展，器件型号和软件版本的不断更新，原有图书的内容和知识体系已经不适应目前的读者需求，为此我们根据 Intel（注：原 Altera 已于 2016 年被 Intel 收购）推出的一系列新型 FPGA，以及新版 Quartus II 软件的特性，对上述两本书进行了改版升级。

新版的《Intel FPGA/CPLD 设计（基础篇）》（以下简称“本书”）涵盖了所有 Intel 主流 FPGA/CPLD 硬件结构与特点，详尽地讨论了 Quartus II 与第三方 EDA 工具的设计方法，系统地阐述了 Intel 可编程逻辑设计优化技术。

本书共 7 章，各章内容简要介绍如下。

- 第 1 章：在介绍可编程逻辑设计技术基本概念的基础上讨论了 FPGA/CPLD 的基本结构、设计流程和开发工具，并展望了下一代可编程逻辑设计技术。
- 第 2 章：介绍了 Intel 高密度、低成本 FPGA 与 CPLD 等主流器件的结构特点。
- 第 3 章：重点介绍了 Quartus II 的软件设计流程，并且结合了具体的滤波器设计案例，同时使用前面所学到的工具，详细介绍了优化、分析、查看、验证设计方法。
- 第 4 章：介绍了 Intel FPGA IP 设计工具与使用方法。
- 第 5 章：介绍了 I/O 分配验证工具、功率估计和分析工具、RTL 阅读器、SignalProbe（信号探针）、SignalTap II 逻辑分析器、Chip Editor（底层编辑器）、Timing Closure Floorplan（时序收敛平面布局规划器）和 ECO（工程更改管理）等工具的使用方法。
- 第 6 章：介绍了 Intel FPGA/CPLD 的常用编程与配置方法。
- 第 7 章：在综述第三方综合工具的基础上，重点介绍了 ModelSim 仿真工具与 Synplify/Synplify Pro 综合工具的使用方法。

本书的主要特点如下。

- 全面系统：涵盖了 Intel FPGA 软、硬件设计技术，基础与高级设计工具，全面系统地论述了 Intel 可编程设计技术。
- 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 FPGA 器件制造公司和 EDA 软件设计公司联系紧密，所以有幸能够在第一时间使用最新版本的 FPGA/CPLD 设计工具。书中涉及的所有工具均根据较新资料撰写，使图书介绍的内容新颖。
- 剖析深刻：书中对 FPGA/CPLD 设计的基本原理、方法有较为详尽的论述，对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。

## 配套资源

配套资源中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（读者可扫描封面上的二维码进行下载）。

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 Quartus II 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复制到计算机硬盘上，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。

## 本书约定

为了方便读者阅读，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。

全书的各章节分别由王欣、王江宏、蔡海宁、王诚和吴继华等作者执笔，由 Intel 公司可编程解决方案事业部（Programmable Solutions Group）资深现场应用工程师王江宏和资深应用工程师王欣统筹定稿。

Intel 公司亚太区可编程解决方案事业部现场应用工程总监邓海涛先生、亚太区应用工程总监罗小锋先生、中国区大客户销售总监吕家龙先生、现场应用工程经理赵敏先生对本书提出了许多建设性意见，并给予作者多方面的帮助。在这里要特别感谢 Intel 公司亚太区可编程解决方案事业部副总裁庄秉翰先生在百忙之中亲自为本书撰写序言。感谢所有关心并支持本书的同仁佳友！

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。  
电子邮件：[adeli.wang@intel.com](mailto:adeli.wang@intel.com)（作者），[liyongtao@ptpress.com.cn](mailto:liyongtao@ptpress.com.cn)（责任编辑）。  
如果您需要得到 Intel 更全面的服务与技术支持，请访问 <http://www.altera.com.cn>。

2017 年 5 月

# 目 录

第 1 章 FPGA/CPLD 简介 .....	1
1.1 可编程逻辑设计技术简介 .....	1
1.1.1 可编程逻辑器件发展简史 .....	1
1.1.2 可编程逻辑器件分类 .....	2
1.2 FPGA 带来的优势 .....	2
1.3 FPGA/CPLD 的基本结构 .....	3
1.3.1 FPGA 的基本结构 .....	3
1.3.2 CPLD 的基本结构 .....	7
1.3.3 FPGA 和 CPLD 的比较 .....	9
1.4 FPGA/CPLD 的设计流程 .....	10
1.5 FPGA/CPLD 的常用开发工具 .....	14
1.6 下一代可编程逻辑设计技术展望 .....	18
1.6.1 下一代可编程逻辑器件硬件上的四大发展趋势 .....	18
1.6.2 下一代 EDA 软件设计方法发展趋势 .....	23
1.7 小结 .....	27
1.8 问题与思考 .....	27
第 2 章 Altera FPGA/CPLD 的结构 .....	28
2.1 Altera 高密度 FPGA .....	28
2.1.1 主流高端 FPGA——Stratix V E/GX/GT/GS .....	28
2.1.2 内嵌 10Gbit/s 高速串行收发器的 FPGA——Stratix IV GT .....	50
2.1.3 内嵌高速串行收发器的中端 FPGA——ArriaII GX .....	53
2.2 Altera 低成本 FPGA .....	53
2.3 Altera 的 CPLD 器件 .....	59
2.4 小结 .....	62
2.5 问题与思考 .....	62
第 3 章 Altera Quartus II 开发流程 .....	63
3.1 Quartus II 软件综述 .....	63
3.1.1 Quartus II 软件的功能 .....	63
3.1.2 Quartus II 软件的用户界面 .....	65
3.2 设计输入 .....	68
3.2.1 设计输入方式 .....	70
3.2.2 设计规划 .....	71
3.2.3 设计输入文件实例 .....	71
3.2.4 设计约束 .....	73
3.3 综合 .....	79
3.3.1 使用 Quartus II 软件集成综合 .....	79
3.3.2 控制综合 .....	79
3.3.3 综合实例 .....	84



3.3.4	第三方综合工具.....	85
3.4	布局布线.....	86
3.4.1	设置布局布线参数.....	86
3.4.2	布局布线实例.....	90
3.4.3	增量布局布线.....	91
3.4.4	反标保留分配.....	91
3.5	仿真.....	92
3.5.1	建立矢量源文件.....	92
3.5.2	仿真实例.....	95
3.6	编程与配置.....	96
3.6.1	建立编程文件.....	96
3.6.2	器件编程和配置.....	98
3.7	案例分析：滤波器的设计.....	100
3.7.1	第 1 个滤波器.....	100
3.7.2	第 2 个滤波器：插入流水.....	103
3.7.3	第 3 个滤波器：对称结构.....	105
3.7.4	第 4 个滤波器：脉动滤波器（Systolic Filter）.....	106
3.7.5	第 5 个滤波器：对称反向结构脉动滤波器.....	107
3.7.6	案例总结.....	109
3.8	小结.....	109
3.9	问题与思考.....	109
<b>第 4 章</b>	<b>Altera 的 IP 工具.....</b>	<b>110</b>
4.1	IP 的概念和 Altera 的 IP.....	110
4.1.1	IP 的概念.....	110
4.1.2	Altera 可提供的 IP.....	111
4.1.3	Altera IP 在设计中的作用.....	113
4.2	使用 Altera 的基本宏功能.....	114
4.2.1	定制基本宏功能.....	115
4.2.2	实现基本宏功能.....	119
4.2.3	设计实例.....	122
4.3	使用 Altera 的 IP 核.....	125
4.3.1	定制 IP 核.....	125
4.3.2	实现 IP 核.....	129
4.3.3	设计实例.....	130
4.4	小结.....	131
4.5	问题与思考.....	131
<b>第 5 章</b>	<b>Quartus II 的常用辅助设计工具.....</b>	<b>132</b>
5.1	I/O 分配验证.....	132
5.1.1	I/O 分配验证功能简介.....	133
5.1.2	I/O 分配验证流程.....	133
5.1.3	用于 I/O 分配验证的输入.....	136

5.1.4	运行 I/O 分配验证 .....	137
5.2	功率分析 .....	139
5.2.1	Excel-based 功率计算器 .....	139
5.2.2	Simulation-based 功率估算 .....	141
5.3	RTL 阅读器 .....	142
5.3.1	RTL 阅读器简介 .....	142
5.3.2	RTL 阅读器用户界面 .....	143
5.3.3	原理图的分页和模块层次的切换 .....	144
5.3.4	过滤原理图 .....	145
5.3.5	将原理图中的节点定位到源设计文件 .....	147
5.3.6	在原理图中查找节点或网线 .....	147
5.3.7	使用 RTL 阅读器分析设计中的问题 .....	148
5.4	SignalProbe 及 SignalTap II 逻辑分析器 .....	148
5.4.1	SignalProbe .....	148
5.4.2	SignalTap II .....	151
5.5	时序收敛平面布局规划器 (Timing Closure Floorplan) .....	157
5.5.1	使用 Timing Closure Floorplan 分析设计 .....	158
5.5.2	使用 Timing Closure Floorplan 优化设计 .....	164
5.6	Chip Editor 底层编辑器 .....	164
5.6.1	Chip Editor 功能简介 .....	164
5.6.2	使用 Chip Editor 的设计流程 .....	165
5.6.3	Chip Editor 视图 .....	166
5.6.4	资源特性编辑器 .....	168
5.6.5	Chip Editor 的一般应用 .....	172
5.7	工程更改管理 (ECO) .....	172
5.7.1	ECO 简介 .....	173
5.7.2	ECO 的应用范围 .....	173
5.7.3	ECO 的操作流程 .....	174
5.7.4	使用 Change Manager 查看和管理更改 .....	175
5.7.5	ECO 验证 .....	176
5.8	小结 .....	176
5.9	问题与思考 .....	176
<b>第 6 章</b>	<b>编程与配置 .....</b>	<b>177</b>
6.1	配置 Altera FPGA .....	177
6.1.1	配置方式 .....	177
6.1.2	主动串行 (AS) .....	183
6.1.3	被动串行 (PS) .....	186
6.1.4	快速被动并行 (FPP) .....	188
6.1.5	被动并行异步 (PPA) .....	189
6.1.6	JTAG 配置方式 .....	191
6.1.7	ByteBlaster II 下载电缆 .....	192

6.1.8	配置芯片 .....	194
6.2	配置文件和软件支持 .....	194
6.2.1	软件支持 .....	194
6.2.2	配置文件 .....	196
6.3	单板设计及调试注意事项 .....	199
6.3.1	配置的可靠性 .....	200
6.3.2	单板设计要点 .....	200
6.3.3	调试建议 .....	201
6.4	小结 .....	203
6.5	问题与思考 .....	203
<b>第 7 章</b>	<b>第三方 EDA 工具 .....</b>	<b>204</b>
7.1	第三方 EDA 工具综述 .....	204
7.1.1	NativeLink 与 WYSIWYG .....	204
7.1.2	3 种 EDA 工具的使用流程 .....	205
7.1.3	Quartus II 支持的第三方工具 .....	205
7.2	仿真的概念与 ModelSim 仿真工具 .....	207
7.2.1	仿真简介 .....	207
7.2.2	仿真的切入点 .....	208
7.2.3	ModelSim 仿真工具的不同版本 .....	210
7.2.4	ModelSim 的图形用户界面 .....	210
7.2.5	ModelSim 的基本仿真步骤 .....	221
7.2.6	使用 ModelSim 进行功能仿真 .....	226
7.2.7	使用 ModelSim 进行时序仿真 .....	230
7.2.8	ModelSim 仿真工具高级应用 .....	232
7.3	综合的概念与 Synplify/Synplify Pro 综合工具 .....	242
7.3.1	Synplify/Synplify Pro 的功能与特点 .....	242
7.3.2	Synplify Pro 的用户界面 .....	248
7.3.3	Synplify Pro 综合流程 .....	251
7.3.4	Synplify Pro 的其他综合技巧 .....	272
7.4	小结 .....	284
7.5	问题与思考 .....	284

# 第1章 FPGA/CPLD 简介

本章内容侧重于一般性可编程设计理论，综合讨论目前的主流器件和 EDA 开发工具。本章在引入 FPGA/CPLD 基本概念的基础上，重点论述了 FPGA/CPLD 的完整设计流程，简要介绍了 FPGA/CPLD 的常用开发工具，最后展望了 FPGA/CPLD 设计技术的新发展。

本章主要内容如下。

- 可编程逻辑设计技术简介。
- FPGA/CPLD 的基本结构。
- FPGA/CPLD 的设计流程。
- FPGA/CPLD 的常用开发工具。
- 下一代可编程逻辑设计技术展望。

## 1.1 可编程逻辑设计技术简介

本节在讨论可编程逻辑器件发展简史的基础上简述目前常用的可编程逻辑器件的分类。

### 1.1.1 可编程逻辑器件发展简史

随着微电子设计技术与工艺的发展，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，缩小了设计的物理尺寸，推动了社会的数字化进程。但是，ASIC 设计周期长、改版投资大、灵活性差等缺陷，制约了 ASIC 的应用范围。硬件工程师希望有一种更灵活的设计方法，可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，这是提出可编程逻辑器件的基本思想。

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（E<sup>2</sup>PROM），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复杂可编程逻辑器件（CPLD）和现场可编程逻辑器件（FPGA）。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（System On Programmable Chip, SOPC）提供了强大的硬件支持。



## 1.1.2 可编程逻辑器件分类

广义上讲, 可编程逻辑器件是指一切通过软件手段更改、配置器件内部连接结构和逻辑单元, 完成既定设计功能的数字集成电路。目前常用的可编程逻辑器件主要有简单的逻辑阵列 (PAL/GAL)、复杂可编程逻辑器件 (CPLD) 和现场可编程逻辑阵列 (FPGA) 等 3 大类。

**PAL/GAL。** PAL 是 Programmable Array Logic 的缩写, 即可编程阵列逻辑; GAL 是 Generic Array Logic 的缩写, 即通用可编程阵列逻辑。PAL/GAL 是早期可编程逻辑器件的发展形式, 其特点是大多基于  $E^2$ CMOS 工艺, 结构较为简单, 可编程逻辑单元多为与阵列和或阵列, 可编程单元密度较低, 仅能适用于某些简单的数字逻辑电路。虽然 PAL/GAL 密度较低, 但是它们一出现即以其低功耗、低成本、高可靠性、软件可编程、可重复更改等特点引发了数字电路领域的巨大震动。虽然目前较复杂的逻辑电路一般使用 CPLD 甚至 FPGA 完成, 但是对于很多简单的数字逻辑, GAL 等简单的可编程逻辑器件仍然被大量使用。目前, 国内外很多对成本十分敏感的设计都在使用 GAL 等低成本可编程逻辑器件, 越来越多的 74 系列逻辑电路被 GAL 取代。GAL 等器件发展至今已经 20 多年, 新一代的 GAL 以功能灵活、小封装、低成本、重复可编程、应用灵活等优点仍然在数字电路领域扮演着重要的角色。目前, 比较大的 GAL 器件供应商主要是 Lattice 半导体。

### (1) CPLD。

CPLD 是 Complex Programmable Logic Device 的缩写, 即复杂的可编程逻辑器件。Altera 为了突出特性, 曾将自己的 CPLD 器件称为 EPLD (Enhanced Programmable Logic Device), 即增强型可编程逻辑器件。其实 EPLD 和 CPLD 属于同等性质的逻辑器件, 目前 Altera 为了遵循称呼习惯, 已经将其 EPLD 统称为 CPLD。CPLD 是在 PAL、GAL 的基础上发展起来的, 一般采用  $E^2$ CMOS 工艺, 也有少数厂商采用 Flash 工艺, 其基本结构由可编程 I/O 单元、基本逻辑单元、布线池和其他辅助功能模块构成。CPLD 可实现的逻辑功能比 PAL 和 GAL 有了大幅度提升, 一般可以完成设计中较复杂、较高速度的逻辑功能, 如接口转换、总线控制等。CPLD 的主要器件供应商有 Altera、Lattice 和 Xilinx 等。

### (2) FPGA。

FPGA 是 Filed Programmable Gate Array 的缩写, 即现场可编程逻辑阵列。FPGA 是在 CPLD 的基础上发展起来的新型高性能可编程逻辑器件, 一般采用 SRAM 工艺, 也有一些专用器件采用 Flash 工艺或反熔丝 (Anti-Fuse) 工艺等。FPGA 的集成度很高, 其器件密度从数万系统门到数千万系统门不等, 可以完成极其复杂的时序与组合逻辑电路功能, 适用于高速、高密度的高端数字逻辑电路设计领域。FPGA 的基本组成部分有可编程输入/输出单元、基本可编程逻辑单元、嵌入式块 RAM、丰富的布线资源、底层嵌入功能单元、内嵌专用硬核等。FPGA 的主要器件供应商有 Xilinx、Altera、Lattice、Actel 和 Atmel 等。

## 1.2 FPGA 带来的优势

从可编程逻辑器件诞生的初始阶段, 一直被当作芯片验证器件来使用。因为它不能达到很快的速度, 同时逻辑的规模也有限, 价格又昂贵, 所以很少被使用到最终的产品中。导致



产能有限,需求量也并不大,和 ASIC 无法相提并论。但是从本世纪开始,情况慢慢发生了变化。首先是 FPGA 的工艺提升,使得器件能支持比较快速的性能,同时逻辑规模也大规模提升。业界慢慢发现 FPGA 也可以使用在最终产品当中并且具备以下一些独特的优势。

- 价格优势。这似乎和 FPGA 相对昂贵的价格有一些矛盾,但确实是事实。一颗 ASIC 芯片,只有在大规模使用的情况下才能获得利润,否则无法覆盖前期付出的流片的成本。而 FPGA 的单价确实比较贵一些,但是不需要厂家付出流片的成本,而且可以保证 100%的成品率。因而对很多供货规模并不庞大的 ASIC 而言,具有价格优势,可以帮助厂家降低成本。
- 时间优势。时间是产品占领市场的关键因素。FPGA 相对 ASIC 具有显而易见的时间优势。一般规模的 FPGA 开发速度可以在半年左右,甚至可以更快,只要研发能力强。而常规芯片限制于生产厂家的时间表,通常会需要一年到两年的时间。这对于很多急需占据市场的产品而言,无法接受。
- 更新优势。可编程逻辑器件是可以反复配置的,甚至可以远程修改配置文件。这是这种器件的独特优势,类似于 CPU 的软件可以升级。厂家同样可以对芯片硬件进行升级,比如一些协议还没有成熟,或者在后面的升级中需要添加协议、功能的产品中,可以得到很好的利用。甚至有些厂家为了尽早占领市场,可以先完成主要功能提供给客户,然后在后面慢慢改进,甚至可以为客户定制。
- 自主优势。在一些行业,比如视频,没有什么秘密可言,几乎所有的厂家都使用类似的芯片,而同类芯片的解决方案也几乎是一样的。为了获得市场差异化的优势,有些厂家希望能开发出一些特别的功能,同时又不希望花很大的代价来生产自己的芯片,这个时候 FPGA 就成为它们的选择。

因此, FPGA 已经不再是简单的验证芯片,由于功能不断加强,已经成为业界各个领域里重要的解决方式之一。所以学习 FPGA,并不只是为了学习数字电路之类的基础知识,来获得一些感官上的收获,更多的是成为未来工作的砝码。

## 1.3 FPGA/CPLD 的基本结构

本节在讨论 FPGA 与 CPLD 的通用结构的基础上,比较两者的异同,加深读者对这两种通用的可编程逻辑器件的认识。

### 1.3.1 FPGA 的基本结构

简化的 FPGA 基本由 6 部分组成,分别为可编程输入/输出单元、基本可编程逻辑单元、嵌入式块 RAM、丰富的布线资源、底层嵌入功能单元和内嵌专用硬核等,如图 1-1 所示。

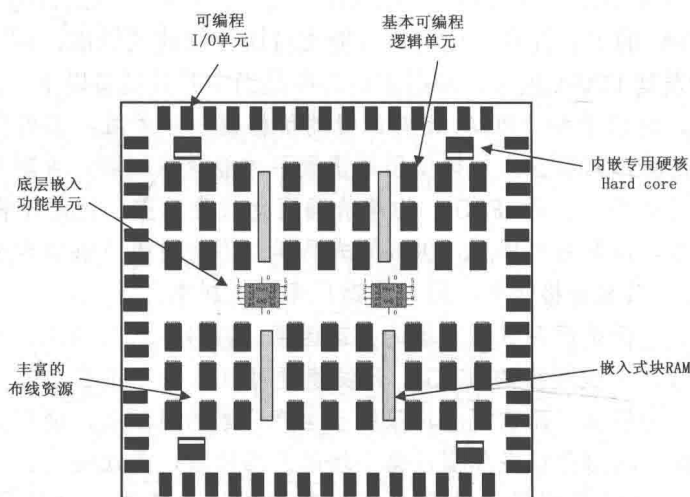


图1-1 可编程逻辑器件的结构原理图

每个单元的基本概念介绍如下。

#### (1) 可编程输入/输出单元。

输入/输出 (Input/Output) 单元简称 I/O 单元，它们是芯片与外界电路的接口部分，完成不同电气特性下对输入/输出信号的驱动与匹配需求。为了使 FPGA 有更灵活的应用，目前大多数 FPGA 的 I/O 单元被设计为可编程模式，即通过软件的灵活配置，可以适配不同的电气标准与 I/O 物理特性；可以调整匹配阻抗特性，上/下拉电阻；可以调整输出驱动电流的大小等。

可编程 I/O 单元支持的电气标准因工艺而异，不同器件商不同器件族的 FPGA 支持的 I/O 标准也不同，一般说来，常见的电气标准有 LVTTTL、LVCMOS、SSTL、HSTL、LVDS、LVPECL 和 PCI 等。值得一提的是，随着 ASIC 工艺的飞速发展，目前可编程 I/O 支持的最高频率越来越高，一些高端 FPGA 通过 DDR 寄存器技术，甚至可以支持高达 2Gbit/s 的数据速率。

#### (2) 基本可编程逻辑单元。

基本可编程逻辑单元是可编程逻辑的主体，可以根据设计灵活地改变其内部连接与配置，完成不同的逻辑功能。FPGA 一般是基于 SRAM 工艺的，其基本可编程逻辑单元几乎都是由查找表 (LUT, Look Up Table) 和寄存器 (Register) 组成的。FPGA 内部查找表一般为 4 输入 (注: Altera Stratix II 的自适应逻辑模块 ALM 结构比较特殊)，查找表一般完成纯组合逻辑功能。FPGA 内部寄存器结构相当灵活，可以配置为带同步/异步复位或置位、时钟使能的触发器 (FF, Flip Flop)，也可以配置成为锁存器 (Latch)。FPGA 一般依赖寄存器完成同步时序逻辑设计。一般来说，比较经典的基本可编程单元的配置是一个寄存器加一个查找表，但是不同厂商的寄存器和查找表的内部结构有一定的差异，而且寄存器和查找表的组合模式也不同。例如，Altera 可编程逻辑单元通常被称为 LE (Logic Element, 逻辑单元)，由一个 Register 加一个 LUT 构成。Altera 的大多数 FPGA 将 10 个 LE 有机地组合起来，构成更大的功能单元——逻辑阵列模块 (LAB, Logic Array Block)，LAB 中除了 LE 还包含 LE 间的进位链、LAB 控制信号、局部互联线资源、LUT 级联链、寄存器级联链等连



线与控制资源。Xilinx 可编程逻辑单元叫 Slice，它由上下两个部分构成，每个部分都由一个 Register 加一个 LUT 组成，被称为 LC (Logic Cell, 逻辑单元)，两个 LC 之间有一些共用逻辑，可以完成 LC 之间的配合与级联。Lattice 的底层逻辑单元叫 PFU (Programmable Function Unit, 可编程功能单元)，它由 8 个 LUT 和 8~9 个 Register 构成。当然这些可编程单元的配置结构随着器件的发展也在不断更新，最新的一些可编程逻辑器件常常根据设计需求推出一些新的 LUT 和 Register 的配置比率，并优化其内部的连接构造。

学习底层配置单元的 LUT 和 Register 比率的一个重要意义在于器件选型和规模估算。很多器件手册上用器件的 ASIC 门数或等效的系统门数表示器件的规模，但目前 FPGA 内部除了基本可编程逻辑单元外，还包含有丰富的嵌入式 RAM、PLL 或 DLL，以及专用 Hard IP Core (硬知识产权功能核) 等。这些功能模块也会等效出一定规模的系统门，所以用系统门权衡基本可编程逻辑单元的数量是不准确的，常常混淆设计者。比较简单科学的方法是用器件的 Register 或 LUT 的数量衡量 (一般来说两者比例为 1:1)。例如，Xilinx 的 Spartan-III 系列的 XC3S1000 有 15360 个 LUT，而 Lattice 的 EC 系列 LFEC15E 也有 15360 个 LUT，所以这两款 FPGA 的可编程逻辑单元数量基本相当，属于同一规模的产品。同样道理，Altera 的 Cyclone 器件族的 EP1C12 的 LUT 数量是 12060 个，就比前面提到的两款 FPGA 规模略小。需要说明的是，器件选型是一个综合性问题，需要将设计的需求、成本压力、规模、速度等级、时钟资源、I/O 特性、封装、专用功能模块等诸多因素综合考虑。

### (3) 嵌入式块 RAM。

目前大多数 FPGA 都有内嵌的块 RAM (Block RAM)。FPGA 内部嵌入可编程 RAM 模块，大大地拓展了 FPGA 的应用范围和使用灵活性。FPGA 内嵌的块 RAM 一般可以灵活配置为单端口 RAM (SPRAM, Single Port RAM)、双端口 RAM (DPRAM, Double Ports RAM)、伪双端口 RAM (Pseudo DPRAM)、CAM (Content Addressable Memory)、FIFO (First In First Out) 等常用存储结构。RAM 的概念和功能读者应该非常熟悉，在此不再赘述。FPGA 中其实并没有专用的 ROM 硬件资源，实现 ROM 的思路是对 RAM 赋予初值，并保持该初值。所谓 CAM，即内容地址存储器。CAM 这种存储器在其每个存储单元都包含了一个内嵌的比较逻辑，写入 CAM 的数据会和其内部存储的每一个数据进行比较，并返回与端口数据相同的所有内部数据的地址。概括地讲，RAM 是一种根据地址读、写数据的存储单元；而 CAM 和 RAM 恰恰相反，它返回的是与端口数据相匹配的内部地址。CAM 的应用也非常广泛，比如在路由器中的地址交换表等。FIFO 是“先进先出队列”式存储结构。FPGA 内部实现 RAM、ROM、CAM、FIFO 等存储结构都可以基于嵌入式块 RAM 单元，并根据需求自动生成相应的粘合逻辑 (Glue Logic) 以完成地址和片选等控制逻辑。

不同器件商或不同器件族的内嵌块 RAM 的结构不同，Xilinx 常见的块 RAM 大小是 4kbit 和 18kbit 两种结构，Lattice 常用的块 RAM 大小是 9kbit，Altera 的块 RAM 最为灵活，一些高端器件内部同时含有两种块 RAM 结构，分别是 M9K RAM (9kbit) 和 M-144K RAM (144kbit)。

需要补充一点的是，除了块 RAM，Altera、Xilinx 和 Lattice 的 FPGA 还可以灵活地将 LUT 配置成 RAM、ROM、FIFO 等存储结构，这种技术被称为分布式 RAM (Distributed RAM)。根据设计需求，块 RAM 的数量和配置方式也是器件选型的一个重要标准。

### (4) 丰富的布线资源。





布线资源连通 FPGA 内部所有单元，连线的长度和工艺决定着信号在连线上的驱动能力和传输速度。FPGA 内部有着非常丰富的布线资源，这些布线资源根据工艺、长度、宽度和分布位置的不同而被划分为不同的等级，有一些是全局性的专用布线资源，用以完成器件内部的全局时钟和全局复位/置位的布线；一些叫作长线资源，用以完成器件 Bank（分区）间的一些高速信号和一些第二全局时钟信号（有时也被称为 Low Skew 信号）的布线；还有一些叫作短线资源，用以完成基本逻辑单元之间的逻辑互联与布线；另外，在基本逻辑单元内部还有着各式各样的布线资源和专用时钟、复位等控制信号线。

实现过程中，设计者一般不需要直接选择布线资源，而是由布局布线器自动根据输入的逻辑网表的拓朴结构和约束条件选择可用的布线资源连通所用的底层单元模块，所以设计者常常忽略布线资源。其实布线资源的优化与使用和设计的实现结果（包含速度和面积两个方面）有直接关系。

#### (5) 底层嵌入功能单元。

底层嵌入功能单元的概念比较笼统，这里我们指的是那些通用程度较高的嵌入式功能模块，比如 PLL（Phase Locked Loop）、DLL（Delay Locked Loop）、DSP、CPU 等。随着 FPGA 的发展，这些模块被越来越多地嵌入到 FPGA 的内部，以满足不同场合的需求。

目前大多数 FPGA 厂商都在 FPGA 内部集成了 DLL 或 PLL 硬件电路，用以完成时钟的高精度、低抖动的倍频、分频、占空比调整、移相等功能。目前，高端 FPGA 产品集成的 DLL 和 PLL 资源越来越丰富，功能越来越复杂，精度越来越高（一般在 ps 的数量级）。Altera 芯片集成的是 PLL，Xilinx 芯片主要集成的是 DLL，Lattice 的新型 FPGA 同时集成了 PLL 与 DLL 以适应不同的需求。Altera 芯片的 PLL 模块分为增强型 PLL（Enhanced PLL）和快速 PLL（Fast PLL）等。Xilinx 芯片 DLL 的模块名称为 CLKDLL，在高端 FPGA 中，CLKDLL 的增强型模块为 DCM（Digital Clock Manager，数字时钟管理模块）。这些时钟模块的生成和配置方法一般分为两种，一种是在 HDL 代码和原理图中直接实例化，另一种方法是在 IP 核生成器中配置相关参数，自动生成 IP。Altera 的 IP 核生成器叫作 Mega Wizard，Xilinx 的 IP 核生成器叫作 Core Generator，Lattice 的 IP 核生成器叫作 Module/IP Manager。另外，可以通过在综合、实现步骤的约束文件中编写约束属性来完成时钟模块的约束。

越来越多的高端 FPGA 产品将包含 DSP 或 CPU 等软处理核，从而使 FPGA 由传统的硬件设计手段逐步过渡为系统级设计平台。例如，Altera 的 Stratix、Stratix GX、Stratix II 等器件族内部集成了 DSP Core，配合通用逻辑资源，还可以实现 ARM、MIPS、NIOS 等嵌入式处理器系统；Xilinx 的 Virtex II 和 Virtex II Pro 系列 FPGA 内部集成了 Power PC 450 的 CPU Core 和 MicroBlaze RISC 处理器 Core；Lattice 的 ECP 系列 FPGA 内部集成了系统 DSP Core 模块。这些 CPU 或 DSP 处理模块的硬件主要由一些加、乘、快速进位链、Pipelining 和 Mux 等结构组成，加上用逻辑资源和块 RAM 实现的软核部分，就组成了功能强大的软运算中心。这种 CPU 或 DSP 比较适合实现 FIR 滤波器、编码解码、FFT（快速傅里叶变换）等运算密集型应用。FPGA 内部嵌入 CPU 或 DSP 等处理器，使 FPGA 在一定程度上具备了实现软硬件联合系统的能力，FPGA 正逐步成为 SPOC（System On Programmable Chip）的高效设计平台。Altera 的系统级开发工具是 Qsys 和 DSP Builder，通过这些平台用户可以方便地设计标准的 DSP 处理器（如 ARM、NIOS 等）、专用硬件结构与软硬件协同处理模块等。