

FPGA

数字电子系统设计与开发实例导航

◆ 刘 韬 楼兴华 编著

- 第1章 可编程逻辑器件与FPGA
- 第2章 VHDL与Verilog HDL
- 第3章 使用ISE进行FPGA开发
- 第4章 模拟I²C协议
- 第5章 UART控制器
- 第6章 USB接口控制器
- 第7章 数字视频信号处理器
- 第8章 VGA/LCD显示控制器
- 第9章 CAN总线控制器
- 第10章 以太网控制器(MAC)



人民邮电出版社
POSTS & TELECOM PRESS



源代码光盘
CD-ROM

计算机接口技术系列

FPGA

数字电子系统设计与开发实例导航

◆ 刘 韬 楼兴华 编著



人民邮电出版社
POSTS & TELECOM PRESS

图书在版编目 (CIP) 数据

FPGA 数字电子系统设计与开发实例导航 / 求是科技编著. —北京: 人民邮电出版社, 2005.6

ISBN 7-115-13189-9

I. F... II. 求... III. 可程序逻辑器件—基本知识 IV. TP332.1

中国版本图书馆 CIP 数据核字 (2005) 第 039839 号

内 容 提 要

本书首先介绍了 FPGA 的相关基础知识, 然后分别通过 7 个在实际工程应用中的案例详细介绍了通过 FPGA 实现 I²C 协议要求的接口、UART 控制器、USB 接口控制器、数字视频信号处理器、VGA/LCD 显示控制器、CAN 总线控制器、以太网控制器的方法。

本书所介绍的案例立足于工程实践, 符合实际应用中的开发过程, 在案例介绍过程中结合了作者大量的开发经验。

本书适合 FPGA 的系统开发与设计人员阅读。

计算机接口技术系列

FPGA 数字电子系统设计与开发实例导航

- ◆ 编 著 刘 韬 楼兴华
责任编辑 张立科
- ◆ 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
北京鸿佳印刷厂印刷
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16
印张: 27
字数: 655 千字
印数: 1—5 000 册
- 2005 年 6 月第 1 版
2005 年 6 月北京第 1 次印刷

ISBN 7-115-13189-9/TP · 4519

定价: 48.00 元 (附光盘)

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

前 言

可编程逻辑器件 FPGA (Field Programmable Gate Array, 现场可编程门阵列) 和 CPLD (Complex Programmable Logic Device, 复杂可编程逻辑器件) 由于具有开发简单、静态可重复编程和动态在系统编程 (In System Programmable) 的特点, 已经成为当今应用最广泛的两类可编程专用集成电路 (ASIC)。随着微电子技术的迅猛发展, 速度更快、集成度更高的 FPGA 不断出现。随着结构和工艺的改进, FPGA 芯片上包含的资源越来越丰富, 可实现的功能越来越强, 这使得 FPGA 在电子电路设计中扮演的角色越来越重要。

目前介绍 FPGA 基本技术、开发工具的书籍较多, 但立足于工程实践、介绍实际应用案例的书籍并不多见。在这种情况下, 作者结合自己的项目开发经验编写了此书, 希望能为 FPGA 开发、学习人员提供一定的参考。

本书分为两个部分, 第 1 部分为第 1~3 章, 着重介绍 FPGA 的基本知识、编程语言、开发工具和开发的基本流程; 第 2 部分为第 4~10 章, 通过 7 个取材于实际工程项目的案例介绍如何使用 FPGA 实现电子电路设计中的各种应用, 具体内容如下:

- 第 1 章介绍 FPGA 的技术特点;
- 第 2 章讲解硬件描述语言 VHDL 和 Verilog HDL 的基本语法;
- 第 3 章结合 Xilinx 公司的 FPGA 开发工具 ISE 详细介绍进行 FPGA 开发的流程;
- 第 4 章介绍使用 FPGA 实现满足 I²C 协议要求的接口的方法;
- 第 5 章介绍使用 FPGA 实现 UART 控制器, 实现串行通信的方法;
- 第 6 章详细介绍使用 FPGA 实现 USB 接口控制器的方法;
- 第 7 章介绍使用 FPGA 实现数字视频信号处理器, 完成视频信号从 A/D 转换、格式控制到数据保存多个过程的方法;
- 第 8 章介绍使用 FPGA 实现 VGA/LCD 显示控制器的方法;
- 第 9 章介绍使用 FPGA 实现 CAN 总线控制器的方法;
- 第 10 章介绍使用 FPGA 实现以太网控制器的方法。

本书的主要作者包括刘韬、楼兴华等, 特别感谢骆娟、李友凡、刘思媛等人对本书顺利完成所付出的辛勤汗水和心血。

由于时间仓促, 加之水平有限, 书中的缺点和不足之处在所难免, 敬请读者批评指正, 本书责任编辑的邮箱是 zhanglike@ptpress.com.cn。

编者
2005.04

目 录

第 1 章 可编程逻辑器件与 FPGA	1
1.1 可编程逻辑器件概述	1
1.1.1 可编程逻辑器件简介	1
1.1.2 发展历史	1
1.1.3 可编程逻辑器件与设计软件	2
1.2 FPGA 概述	2
1.2.1 FPGA 简介	2
1.2.2 FPGA 的优点	3
1.2.3 FPGA 的分类和使用	4
1.2.4 主要 FPGA 供应商一览	5
1.3 总结	6
第 2 章 VHDL 与 Verilog HDL	7
2.1 VHDL 简介	7
2.1.1 VHDL 的历史	7
2.1.2 VHDL 的特点	7
2.2 VHDL 程序基本结构	8
2.2.1 实体的申明方法	9
2.2.2 结构体的描述方法	10
2.2.3 程序包和程序包体	15
2.2.4 配置的申明方法	16
2.2.5 VHDL 程序的库	17
2.3 VHDL 语言的数据类型和运算符	19
2.3.1 VHDL 语言的数据对象	20
2.3.2 VHDL 语言的数据类型	21
2.3.3 VHDL 语言的运算符	25
2.4 VHDL 语言的描述语句	27
2.4.1 VHDL 顺序语句描述方法	27
2.4.2 VHDL 并行语句描述方法	32
2.5 VHDL 语言的预定义属性	34
2.5.1 值类预定义属性	34
2.5.2 函数类预定义属性	34
2.5.3 信号类预定义属性	35
2.5.4 数据类型类预定义属性	36
2.5.5 数据范围类预定义属性	36
2.6 Verilog HDL 简介	36
2.6.1 Verilog HDL 的历史	36
2.6.2 Verilog HDL 的特点	36

2.7 Verilog HDL 程序基本结构	37
2.8 Verilog HDL 语言的数据类型和运算符	38
2.8.1 标识符	38
2.8.2 数据类型	38
2.8.3 模块端口	39
2.8.4 值集合	39
2.8.5 表达式	41
2.9 Verilog HDL 语言的描述语句	42
2.9.1 门级建模形式	42
2.9.2 数据流建模形式	43
2.9.3 行为建模形式	44
2.10 小结	50
第3章 使用 ISE 进行 FPGA 开发	51
3.1 ISE 简介	51
3.2 ISE 和 ModelSim 的安装	52
3.2.1 系统要求	52
3.2.2 ISE 的安装	52
3.2.3 ModelSim 安装	55
3.2.4 关联 ISE 和 ModelSim	56
3.3 ISE 的使用方法	57
3.3.1 ISE 界面介绍	57
3.3.2 使用 ISE 的开发流程	60
3.3.3 使用 ISE 编写 HDL 模块	62
3.3.4 使用语言模版辅助设计	68
3.3.5 使用 ISE 编写测试平台	69
3.3.6 使用 ISE 编写顶层模块	75
3.3.7 使用 ISE 编写约束文件	79
3.3.8 设计的综合和实现	81
3.3.9 硬件编程和下载	82
3.4 小结	83
第4章 模拟 I ² C 协议	84
4.1 I ² C 总线解析	84
4.1.1 I ² C 总线概述	84
4.1.2 I ² C 协议的基本概念	84
4.1.3 I ² C 协议的时序要求	85
4.2 模拟 I ² C 接口程序的基本框架	87
4.3 I ² C 协议的具体实现	89
4.3.1 位传输的实现	89
4.3.2 字节传输的实现	99
4.3.3 程序主体的实现	106

4.4 程序的仿真与测试	110
4.4.1 主节点的仿真	111
4.4.2 从节点的仿真	113
4.4.3 仿真主程序	119
4.4.4 仿真结果	125
4.5 小结	126
第5章 UART 控制器	127
5.1 计算机接口技术简介	127
5.1.1 常用的计算机接口简介	127
5.1.2 计算机接口的控制方式	128
5.2 RS-232 串口通信简介	129
5.2.1 串口通信概述	129
5.2.2 RS-232 协议	130
5.2.3 RS-232 通信时序和 UART	133
5.2.4 串口通信实现方案	134
5.3 使用 FPGA 实现 UART	134
5.3.1 UART 实现原理	134
5.3.2 UART 工作流程	136
5.3.3 信号监测器模块的实现	137
5.3.4 波特率发生器模块的实现	139
5.3.5 移位寄存器模块的实现	143
5.3.6 奇偶校验器模块的实现	145
5.3.7 总线选择器模块的实现	147
5.3.8 计数器模块的实现	149
5.3.9 UART 内核模块的实现	151
5.3.10 UART 顶层模块的实现	164
5.3.11 测试平台的编写和仿真	172
5.4 小结	178
第6章 USB 接口控制器	179
6.1 USB 接口简介	179
6.1.1 USB 发展历史	179
6.1.2 USB 特点和应用	179
6.2 USB 体系结构	180
6.2.1 USB 系统描述	180
6.2.2 USB 物理接口	181
6.2.3 USB 电源管理	182
6.2.4 USB 设备	183
6.2.5 USB 主机	183
6.3 USB 通信原理	183
6.3.1 USB 传输模型	184

6.3.2	USB 设备检测过程	188
6.4	USB 系统开发	196
6.4.1	USB 硬件系统	196
6.4.2	USB 接口芯片 PDIUSB12	197
6.5	USB 设计实例	210
6.5.1	设计需求分析	210
6.5.2	设计方案	211
6.6	FPGA 固件开发	211
6.6.1	固件模块划分	211
6.6.2	自定义包编写	212
6.6.3	分频器模块的实现	215
6.6.4	沿控制模块的实现	216
6.6.5	输入/输出切换模块的实现	217
6.6.6	请求处理模块的实现	219
6.6.7	设备收发器模块的实现	222
6.6.8	测试平台的编写	233
6.7	USB 驱动和软件开发	235
6.7.1	USB 驱动编写	235
6.7.2	USB 软件编写	241
6.8	小结	243
第 7 章	数字视频信号处理器	245
7.1	视频信号概述	245
7.1.1	视频信号处理的基本过程	246
7.1.2	数字图像处理技术概念	246
7.1.3	数字视频信号的格式	247
7.2	视频信号处理的框架	248
7.2.1	视频信号转换模块	248
7.2.2	视频数据计算模块	249
7.2.3	通信模块	249
7.3	视频信号处理的电路	250
7.3.1	中央控制器 FPGA	250
7.3.2	电视信号转换模块电路	250
7.3.3	图像缓存部分电路	253
7.4	视频处理程序的具体实现	254
7.4.1	主体程序的实现	254
7.4.2	视频图像数据采集程序的实现	256
7.4.3	SRAM 的读写控制	263
7.5	程序测试与运行	266
7.5.1	测试程序	266
7.5.2	测试结果	273

7.6 小结	274
第8章 VGA/LCD 显示控制器	275
8.1 VGA 显示原理	275
8.1.1 图像显示原理	275
8.1.2 常用视频显示标准	276
8.2 VGA/LCD 显示控制器的基本框架	276
8.3 VGA/LCD 显示控制器程序的实现	277
8.3.1 顶层程序——vga_top	277
8.3.2 颜色查找表——Color Lookup Table	278
8.3.3 颜色处理器——Color Processor	285
8.3.4 光标处理器——Cursor Processor	295
8.3.5 视频定时产生器——Video Timing Generator	301
8.3.6 输出 FIFO	306
8.4 程序的仿真与测试	309
8.5 小结	322
第9章 CAN 总线控制器	323
9.1 CAN 总线协议解析	323
9.1.1 CAN 总线通信模型	324
9.1.2 CAN 总线协议中的基本概念	324
9.1.3 报文的数据结构	325
9.1.4 位时序 (Bit Timing)	327
9.1.5 同步 (Synchronization)	328
9.2 CAN 通信控制器程序基本框架	328
9.2.1 SJA1000CAN 通信控制器	328
9.2.2 CAN 通信控制器程序框架	329
9.3 CAN 通信控制器的具体实现	330
9.3.1 顶层控制程序——TOP	331
9.3.2 寄存器控制	332
9.3.3 位时序逻辑——Bit Timing Logic	339
9.3.4 位数据流处理器——Bit Stream Processor	342
9.3.5 CRC 校验	357
9.3.6 FIFO	357
9.4 程序的仿真与测试	360
9.5 小结	365
第10章 以太网控制器 (MAC)	366
10.1 以太网基本原理	366
10.1.1 以太网协议的参考模型	366
10.1.2 MAC 子层	368
10.1.3 媒体无关接口 (Media Independent Interface)	369
10.2 以太网控制器 (MAC) 的基本框架	370

10.3 以太网控制器 (MAC) 程序的实现	371
10.3.1 顶层程序——eth_top	371
10.3.2 媒体无关接口模块 (Media Independent Interface Module)	373
10.3.3 数据发送模块	379
10.3.4 数据接收模块	392
10.3.5 控制模块	399
10.4 程序的仿真与测试	405
10.4.1 顶层程序	405
10.4.2 外部 PHY 芯片模拟程序	408
10.4.3 仿真结果	418
10.5 小结	419

第 1 章 可编程逻辑器件与 FPGA

1.1 可编程逻辑器件概述

1.1.1 可编程逻辑器件简介

可编程逻辑器件 (Programmable Logic Device) 可以完全由用户通过软件进行配置和编程, 从而完成某种特定的逻辑功能。它是 20 世纪 70 年代在 ASIC 设计的基础上发展起来的新型逻辑器件, 经过 80 年代的发展, PLD 行业初步形成, 而进入 90 年代, 可编程逻辑器件成为半导体领域中发展最快的产品之一。

使用可编程逻辑器件不但能够节省设计面积, 而且使得设计具有很强的灵活性。随着可编程逻辑器件设计技术的发展, 每个逻辑器件中门电路的数量愈来愈多, 一个逻辑器件就可以完成本来要由很多分立逻辑器件和存储芯片完成的功能。这样做减少了系统的功耗和成本, 提高了它的性能和可靠性。

可编程逻辑器件又细分为简单 PLD (SPLD)、复杂 PLD (CPLD) 以及 FPGA。它们内部结构的实现方法各不相同。在本章中“可编程逻辑器件”这个术语是指所有的这类产品, 不同类型的器件则用英文缩写来表示。

1.1.2 发展历史

可编程逻辑器件的发展可以分为 3 个阶段。

早期的可编程逻辑器件只有可编程只读存储器 (PROM)、紫外线可擦除只读存储器 (EPROM) 和电可擦除只读存储器 (EEPROM) 3 种。由于结构的限制, 它们只能完成简单的数字逻辑功能。

其后, 出现了一类结构上稍复杂的可编程芯片, 正式被称为可编程逻辑器件 (PLD), 它能够完成各种逻辑运算功能。典型的 PLD 由一个“与”门和一个“或”门阵列组成, 任意一个组合逻辑都可以用“与或”表达式来描述, 所以, PLD 能以乘积和的形式完成大量的组合逻辑运算。这一阶段的产品主要有 PAL (可编程阵列逻辑) 和 GAL (通用阵列逻辑)。

PAL 由一个可编程的“与”平面和一个固定的“或”平面构成, 或门的输出可以通过触发器有选择地被置为寄存状态。PAL 器件是现场可编程的, 它的实现工艺有反熔丝技术、EPROM 技术和 EEPROM 技术。还有一类结构更为灵活的逻辑器件是可编程逻辑阵列 (PLA), 它也由一个“与”平面和一个“或”平面构成, 但是这两个平面的连接关系是可编程的。

PAL 器件既有现场可编程的, 也有掩膜可编程的。后来在 PAL 的基础上, 又发展出了一种通用阵列逻辑 GAL (Generic Array Logic), 如 GAL16V8、GAL22V10 等。它采用了 EEPROM 工艺, 实现了电可擦除、电可改写, 其输出结构是可编程的逻辑宏单元, 因而它的设计具有很强的灵活性, 至今仍有许多人使用。图 1-1 所示是典型 PLD 的部分结构图。

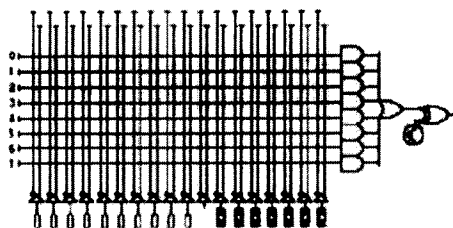


图 1-1 典型 PLD 的部分结构

这些早期的 PLD 器件的一个共同特点是提高了逻辑运算的速度，但其过于简单的结构也使它们只能实现规模较小的电路。为了弥补这一缺陷，20 世纪 80 年代中期，Altera 和 XILINX 分别推出了类似于 PAL 结构的扩展型 CPLD（Complex Programmable Logic Dvice）和与标准门阵列类似的 FPGA（Field Programmable Gate Array），它们都具有体系结构和逻辑单元灵活、集成度高以及适用范围宽等特点。这两种器件兼容了 PLD 和通用门阵列的优点，可实现较大规模的电路，编程也很灵活。与门阵列等其他 ASIC（Application Specific IC）相比，它们又具有设计开发周期短、设计制造成本低、开发工具先进、标准产品无需测试、质量稳定以及可实时在线检验等优点，因此被广泛应用于产品的原型设计和产品生产（一般在 10000 件以下）之中。几乎所有应用门阵列、PLD 和中小规模通用数字集成电路的场合均可应用 FPGA 和 CPLD 器件。

1.1.3 可编程逻辑器件与设计软件

使用高度复杂的可编程逻辑器件，在很大程度上要依靠电子设计自动化（EDA）软件才能求得最佳的系统性能参数，而这些参数的优先排序则取决于应用。如果目标分别是低功耗、高性能或者尽量减少门电路数量，则设计结果就会大不相同。理想的电子设计自动化软件应当做到：

- 设计人员无需过问器件的内部结构细节；
- 设计人员能够确定各个设计目标的优先次序，并根据这一优先次序来优化软件的操作；
- 能够高效率地利用器件的资源；
- 很少甚至根本无需人工干预；
- 能够快速编译或者重新编译一项设计；
- 在每次编译时，有关定时和引出脚的修改应当减少到最低程度，甚至完全无需修改。

可编程逻辑器件厂商硅片产品的技术领先程度及其文档的全面程度，并不是决定他们成败的惟一因素。由可编程逻辑器件公司内部开发或由第三方提供软件工具支持的深度和广度，也是同样重要的。

1.2 FPGA 概述

1.2.1 FPGA 简介

FPGA 是英文 Field Programmable Gate Array 的缩写，即现场可编程门阵列，它是在 PAL、

GAL、EPLD 等可编程器件的基础上进一步发展的产物。它是作为专用集成电路（ASIC）领域中的一种半定制电路而出现的，既解决了定制电路的不足，又克服了原有可编程器件门电路数有限的缺点。

目前 FPGA 的品种很多，有 XILINX 公司的 Spartan、Vertex 系列、ALTERA 公司的 FIEX 系列、Actel 公司的 ProASIC 系列以及 TI 公司的 TPC 系列等。

FPGA 是由存放在片内 RAM 中的程序来设置其工作状态的，因此，工作时需要对片内的 RAM 进行编程。用户可以根据不同的配置模式，采用不同的编程方式。

FPGA 的使用非常灵活。目前，大部分的 FPGA 在使用时都需要外接一个 EPROM 保存其程序，加电时，FPGA 芯片将 EPROM 中的数据读入片内编程 RAM 中，配置完成后，FPGA 进入工作状态。掉电后，FPGA 恢复成白片，内部逻辑关系消失，因此，FPGA 能够反复使用。FPGA 的编程无需专用的 FPGA 编程器，只需用通用的 EPROM、PROM 编程器即可。当需要修改 FPGA 功能时，只需换一片 EPROM 即可。这样，同一片 FPGA，不同的编程数据，可以产生不同的电路功能。FPGA 市场占有率最高的两大公司 XILINX 和 Altera 生产的 FPGA 都是此类型的，另外还有一种基于 Flash 的 FPGA，由 Actel 公司提供。此种 FPGA 其芯片内部自带了 Flash ROM，可以存储程序而不需要外接 ROM，可以节省了成本和制板面积。

FPGA 有如下几种配置模式：

- 并行主模式为一片 FPGA 加一片 EPROM 的方式；
- 主从模式可以支持一片 PROM 编程多片 FPGA；
- 串行模式可以采用串行 PROM 编程 FPGA；
- 外设模式可以将 FPGA 作为微处理器的外设，由微处理器对其编程。

1.2.2 FPGA 的优点

FPGA 是在 PAL、GAL 等逻辑器件的基础之上发展起来的，同以往的 PAL、GAL 等相比较，FPGA 的规模比较大，它可以替代几十甚至几千块通用 IC 芯片。可以说，FPGA 芯片是小批量系统提高系统集成度、可靠性的最佳选择之一。

FPGA 芯片可以被称为可订制的特殊 ASIC 芯片，它除了具有 ASIC 的特点之外，还具有以下几个优点：

- 随着 VLSI（Very Large Scale IC，超大规模集成电路）工艺的不断提高，单一芯片内部已可以容纳上百万个晶体管，这使得 FPGA 芯片所能实现的功能也越来越强，同时也可以实现系统集成。

- FPGA 芯片在出厂之前都做过百分之百的测试，不需要设计人员承担投片风险和费用，设计人员只需在自己的实验室里通过相关的软硬件环境来完成芯片的最终功能设计，所以，FPGA 的资金投入小，节省了许多潜在的花费。

- 用户可以反复地编程、擦除、使用或者在不动外围电路的情况下用不同软件即可实现不同的功能。

- FPGA 软件包中有各种输入工具、仿真工具、版图设计工具和编程器等全线产品，电路设计人员在很短的时间内就可完成电路的输入、编译、优化、仿真，直至最后芯片的制作。

- 电路设计人员使用 FPGA 进行电路设计时，不需要具备专门的 IC（集成电路）深层次的知识，FPGA 软件易学易用。

1.2.3 FPGA 的分类和使用

根据 FPGA 基本结构的不同,可以将其分为基于乘积项 (Product-Term) 技术的 FPGA 和基于查找表 (Look-Up-Table) 技术的 FPGA 两种。

基于乘积项技术的 FPGA 主要由 3 个模块组成——逻辑单元阵列 (Logic Cell Array)、可编程连线 (PIA) 和 I/O 控制块。逻辑单元阵列是 FPGA 的基本结构,由它来实现基本的逻辑功能。可编程连线负责信号传递,连接所有的宏单元。I/O 控制块负责输入/输出的电气特性控制,比如可以设定集电极开路输出、摆率控制、三态输出等。图 1-2 所示是基于乘积项技术 FPGA 的基本原理图。

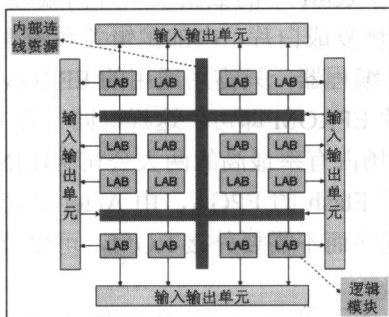


图 1-2 基于乘积项技术 FPGA 的基本原理图

基于查找表技术的 FPGA 也是目前的一种主流产品。查找表简称为 LUT,其本质就是一个 RAM。目前 FPGA 中多使用 4 输入的 LUT,所以每一个 LUT 可以看成是一个有 4 位地址线的 16×1 的 RAM。当用户通过原理图或 HDL 语言描述了一个逻辑电路以后, FPGA 开发软件会自动计算逻辑电路的所有可能的结果,并把结果事先写入 RAM,这样每输入一个信号进行逻辑运算就等于输入一个地址进行查表,找出地址对应的内容,然后输出即可。

在完成 FPGA 开发以后,软件会生成一个最终的编程文件,不同类型的 FPGA 使用不同的方法将编程文件加载到 FPGA 中。

基于乘积项 (Product-Term) 技术的 FPGA 一般采用 EEPROM (或 Flash) 保存程序。此类 FPGA 一般由厂家提供编程电缆,如 Altera 提供的编程电缆叫 Byteblaster。电缆一端接在计算机的并行打印口上,另一端接在 PCB 板上的一个 10 芯插头上, FPGA 芯片有 4 个管脚 (编程脚) 与插头相连。图 1-3 所示是这类电缆使用的示例图。

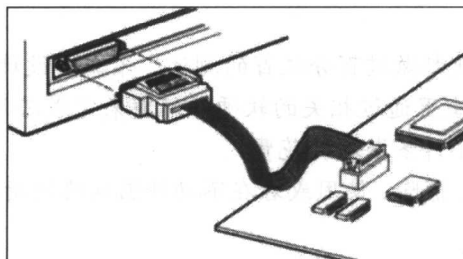


图 1-3 编程电缆使用示例图

基于查找表技术 (Look-Up table) 技术常使用 SRAM 保存程序。由于 SRAM 工艺的特点,掉电后数据会消失,因此调试期间可以通过电缆配置 FPGA 器件,调试完成后,需要将数据固化在一个专用的 EEPROM 中 (用通用编程器烧写),上电时由这片 EEPROM 对 FPGA 加载数

据,十几个毫秒后 FPGA 即可正常工作(亦可由 CPU 配置 FPGA)。此类型的 FPGA 一般不可以进行程序加密。

除了上述的两种 FPGA 外,还有一种反熔丝(Anti-fuse)技术的 FPGA,如 Actel、Quicklogic 的部分产品就采用这种工艺。反熔丝 FPGA 用法与 EEPROM 的 FPGA 一样,但这种 FPGA 不能重复擦写,所以初期开发过程比较麻烦,费用也比较昂贵。但反熔丝技术也有许多优点,如布线能力更强、系统速度更快、功耗更低、同时抗辐射能力强、耐高低温、可以加密,所以在一些有特殊要求的领域,如军事及航空航天中运用较多。

1.2.4 主要 FPGA 供应商一览

随着可编程逻辑器件应用的日益广泛,许多 IC 制造厂家涉足 FPGA 领域。目前世界上有十几家生产 FPGA 的公司,最大的 3 家是 ALTERA、XILINX、Lattice。表 1-1 所示为世界上主要的几家 FPGA 供应商以及其产品特点。

表 1-1 主要 FPGA 供应商

名称	商标	简介
ALTERA		20 世纪 90 年代以后发展很快,是最大可编程逻辑器件供应商之一。主要产品有 MAX3000/7000、FLEX10K、APEX20K、ACEX1K、Stratix、Cyclone 等。开发软件为 MaxplusII 和 QuartusII。用户普遍认为其开发工具 MaxplusII 是最成功的 PLD 开发平台之一,配合 Altera 公司提供的免费 OEM HDL 综合工具使用,可以达到较高的效率
XILINX		FPGA 的发明者,老牌 PLD 公司,是最大可编程逻辑器件供应商之一。产品种类较全,主要有 XC9500/4000、Coolrunner (XPLA3)、Spartan、Virtex 等。开发软件为 Foundation 和 ISE。通常来说,在欧洲用 XILINX 的人多,在日本和亚太地区用 ALTERA 的人多,在美国则是平分秋色。全球 PLD/FPGA 产品 60%以上是由 ALTERA 和 XILINX 提供的。可以讲 ALTERA 和 XILINX 共同决定了 PLD 技术的发展方向
Lattice		Lattice 是 ISP 技术的发明者,ISP 技术极大地促进了 PLD 产品的发展。与 ALTERA 和 XILINX 相比,Lattice 的开发工具比 ALTERA 和 XILINX 略逊一筹,但它在中小规模 PLD 应用上有一定的特色。1999 年收购了 Vantis (原 AMD 子公司)公司,成为第三大可编程逻辑器件供应商。2001 年 12 月收购 agere 公司(原 Lucent 微电子部)的 FPGA 部门。主要产品有 ispLSI2000/5000/8000、MACH4/5、ispMACH4000 等
Actel		反熔丝(一次性烧写)PLD 的领导者,由于反熔丝 PLD 抗辐射、耐高低温、功耗低、速度快,所以在军品和宇航级市场上有较大优势。ALTERA 和 XILINX 则一般不涉足军品和宇航级市场
QuickLogic		专业 PLD/FPGA 公司,以一次性反熔丝工艺为主,有一些集成硬核的 FPGA 比较有特色,但总体上在中国地区销售量不大

需要注意的是,不同厂家对各自器件的叫法不尽相同,XILINX 把基于查找表技术、使用

SRAM 工艺、要外挂配置用 EEPROM 的 PLD 叫 FPGA；把基于乘积项技术，采用 Flash 工艺的 PLD 叫 CPLD。ALTERA 把自己的 PLD 产品 MAX 系列（乘积项技术，. EEPROM 工艺）、FLEX 系列（查找表技术，SRAM 工艺）都叫作 CPLD，即复杂 PLD（Complex PLD）。但由于 FLEX 系列也是 SRAM 工艺、基于查找表技术、要外挂配置用的 EPROM，用法和 XILINX 的 FPGA 一样，所以很多人把 ALTERA 的 FELX 系列产品也叫做 FPGA。

1.3 总结

本章首先介绍了可编程逻辑器件的基本概念和发展历史，接着详细介绍了 FPGA 的优点以及分类，并且列举了目前主要的 FPGA 供应商。FPGA 被广泛地应用于目前的数字系统设计中，所以掌握 FPGA 的开发方法将是当前电子工程师必备的一项技能。

第2章 VHDL 与 Verilog HDL

随着 EDA 技术的发展,使用硬件语言(HDL)进行 FPGA 设计已经成为主流。目前最主要的硬件描述语言是 VHDL 和 Verilog HDL。VHDL 发展得较早并且语法严格,而 Verilog HDL 是在 C 语言的基础上发展起来的一种硬件描述语言,语法较自由。两者相比,VHDL 的书写规则比 Verilog HDL 烦琐一些,但是 Verilog HDL 相对自由的语法也容易让初学者出错。所以,国外很多的电子学专业都会在本科阶段教授 VHDL,在研究生阶段教授 Verilog HDL。从国内来看,使用 VHDL 的开发者相对较多,所以 VHDL 相关的参考书也很多,资料比较充分;而 Verilog HDL 的使用者相对较少,这给学习 Verilog HDL 带来一些困难。目前,已出现用于 FPGA 设计的硬件描述 C 语言 SystemC,虽然还不成熟,各个 FPGA 芯片生产商以及 EDA 软件开发商对它的支持有限,但它有可能会成为继 VHDL 和 Verilog HDL 之后,设计大规模 FPGA 的又一种手段。

本章主要介绍 VHDL 和 Verilog HDL 的基本语法和程序结构,由于目前 SystemC 的使用群体很小,在此不做介绍。

2.1 VHDL 简介

2.1.1 VHDL 的历史

VHDL 的英文全名是 Very-High-Speed Integrated Circuit Hardware Description Language,诞生于 1982 年。1987 年底,VHDL 被 IEEE 和美国国防部确认为标准硬件描述语言。自 IEEE 公布了 VHDL 的标准版本 IEEE-1076(简称 87 版)之后,各 EDA 公司相继推出了自己的 VHDL 设计环境,或宣布自己的设计工具可以提供 VHDL 接口。此后 VHDL 在电子设计领域逐步取代了原有的各种非标准硬件描述语言。1993 年,IEEE 对 VHDL 进行了修订,从更高的抽象层次和系统描述能力上扩展 VHDL 的内容,并公布了新版本的 VHDL,即 IEEE 标准的 1076-1993 版本(简称 93 版)。现在,VHDL 和 Verilog HDL 作为 IEEE 的工业标准硬件描述语言,在电子工程领域已成为事实上的通用硬件描述语言。

2.1.2 VHDL 的特点

VHDL 主要用于描述数字系统的结构、行为、功能和接口。除了含有许多具有硬件特征的语句外,VHDL 在语言形式、描述风格和句法上与一般的计算机高级语言十分相似。VHDL 的程序结构特点是将一项工程设计,或称设计实体(可以是一个元件、一个电路模块或一个系统)分成外部和内部两部分。外部也可称为可视部分,它描述了此模块的端口,而内部可称为不可视部分,它涉及到实体的功能实现和算法完成。在对一个设计实体定义了外部端口后,一旦其内部开发完成,其他的设计就可以直接调用这个实体。这种将设计实体分成内外部分的概念是 VHDL 系统设计的基本点。

应用 VHDL 进行工程设计有以下的优点: