

Verilog
VHDL

FPGA / CPLD

边练边学

——快速入门Verilog/VHDL

(第2版)

吴厚航 编著



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

FPGA/CPLD 边练边学 ——快速入门 Verilog/VHDL (第2版)

吴厚航 编著

北京航空航天大学出版社

内 容 简 介

本书面向广大的 FPGA/CPLD 初学者,从零开始讲述可编程逻辑器件(FPGA/CPLD)以及相关的基础知识,并以一个入门级的学习套件为实验平台,12 个应用实例贯穿其中,不仅有基本的 Verilog/VHDL 语法讲解,而且有设计思路和背景知识的详细描述;手把手地将开发工具(Quartus II+ModelSim)的使用图文并茂地展示给读者。书中例程源程序可在北京航空航天大学出版社网站的“下载中心”免费下载。

本书内容重基础,文字诙谐幽默,适合广大 FPGA/CPLD 的学习者作为入门之选,也可作为具有一定专业知识背景的电子工程师、电子信息类在校本科生、研究生等的参考资料。

图书在版编目(CIP)数据

FPGA/CPLD 边练边学 : 快速入门 Verilog/VHDL / 吴厚航编著. -- 2 版. -- 北京 : 北京航空航天大学出版社, 2017.12

ISBN 978 - 7 - 5124 - 2594 - 1

I. ①F… II. ①吴… III. ①可编程序逻辑器件—系统设计②VHDL 语言—程序设计 IV. ①TP332.1②TP312

中国版本图书馆 CIP 数据核字(2017)第 271950 号

版权所有,侵权必究。

FPGA/CPLD 边练边学 ——快速入门 Verilog/VHDL(第 2 版)

吴厚航 编著

责任编辑 王慕冰

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(邮编 100191) <http://www.buaapress.com.cn>

发行部电话:(010)82317024 传真:(010)82328026

读者信箱:emsbook@gmail.com 邮购电话:(010)82316936

北京泽宇印刷有限公司印装 各地书店经销

*

开本:710×1 000 1/16 印张:17.25 字数:368 千字

2018 年 1 月第 2 版 2018 年 1 月第 1 次印刷 印数:3 000 册

ISBN 978 - 7 - 5124 - 2594 - 1 定价:45.00 元

若本书有倒页、脱页、缺页等印装质量问题,请与本社发行部联系调换。联系电话:(010)82317024

第2版前言

2010年6月和2011年10月,笔者(网名:特权同学)先后出版了图书《深入浅出玩转FPGA》和《爱上FPGA开发——特权和你一起学NIOS II》。这两本书都受到了广大读者的好评,前者在一年内两次重印且于2013年7月推出了第2版;后者也成为了当年Altera公司所举办的大学年会上的指定赠书。此外,笔者也在著名电子网站EDNChina和ChinaAET(《电子技术应用》杂志和电子发烧友网站)建立了专门的FPGA/CPLD助学小组,分享大量的学习资料,也提供了一些帮助初学者快速入门和进阶的开发套件。

1. 为什么写这本书

FPGA技术在国内的发展相对较晚,但FPGA强大的功能,其固有的灵活性和并行性使得很多应用场合非它不可。因此,越来越多的电子工程师和电子专业在校学生都希望能够掌握这门技术。笔者经常收到读者或者开发套件使用者的邮件,所提问题五花八门,笔者虽然尽力答复大家,但着实有些难以招架。其中很大一部分初学者的问题都非常简单和基础,鉴于此,笔者在对这些不断重复的问题的回答过程中,也萌生了写一本基础的FPGA入门教程的想法。

2. 本书的侧重点和读者对象

由于笔者已经出版的两本书中,《深入浅出玩转FPGA(第2版)》重点在FPGA设计的经验和技巧的分享,《爱上FPGA开发——特权和你一起学NIOS II》重点在片上系统的入门开发,因此,你所见到的现在这本书,它的重点则是HDL语法的掌握、软件工具(主要是Quartus II+ModelSim)的使用以及整个开发流程的体验。可以说,它是前面两本书的基础。

需要提醒读者注意的是,本书是基于一个CPLD学习套件来设计各种基本入门例程的。很多人会嗤之以鼻,这不是偷换概念吗?非也,FPGA和CPLD的区别笔者不想过多讨论,仁者见仁智者见智;笔者认为FPGA和CPLD在代码设计、工具使用和流程上几乎是一致的,虽然略有差别。但是对于初学者,你尽可以将CPLD当

作入门级的 FPGA 来用,这样有两个好处:一是降低学习成本(毕竟 FPGA 的价格较高);二是降低学习门槛(CPLD 无论从硬件设计还是内部逻辑上都要比 FPGA 简单得多)。可以肯定地说,如果你通过本书很好地掌握了 HDL 代码设计、工具使用和基本流程,那么再学 FPGA 将变得异常轻松。甚至使用其他的开发工具也会很快上手,因为它们之间都有很大的相通性,很多知识点都是换汤不换药的。

总之,如果你是一个 FPGA 或 CPLD 的初学者,没问题,选择此书,有条件的话再配以书中的 CPLD 套件进行实践,那么入门 FPGA/CPLD 会变得非常容易。

3. 本书的基本内容

本书共 7 章。第 1 章是基础中的基础,讲述可编程器件的发展历史并阐述相关的基本概念,第 2 版主要的改动在第 1 章,补充了一些初学者所关心的基本概念和知识。第 2 章则对后续例程中将要使用到的 CPLD 开发平台进行介绍,深入详细地剖析了整个实验平台的硬件板级设计;第 3 章从最基础的 0 和 1 开始回顾数字电路的基础,也会深入探讨读者所关心的可编程器件的内部架构和原理;第 4 章是 HDL 语言基础,介绍使用最广泛的 Verilog 和 VHDL 语言的基本语法;第 5 章中手把手教大家建立 PC 机上的开发环境,并且图文并茂地让大家体验一个完整的工程创建流程;第 6 章通过 10 个基本实例来掌握 HDL 语言,同时也熟悉工具的使用(Quartus II 和 ModelSim);第 7 章则是 CPLD 内部特有资源的应用案例。本书有对基础理论知识专门的讲解,也有对实践的演练和讲解,更多的是在实践中传递实用的设计技巧和方法,非常适合初学者。

在此要感谢广大的 FPGA/CPLD 爱好者们,你们的学习热情是我编写此书的源动力。非常感谢为书中学习套件的前期设计和推广做出过努力的陈卫东、余国峰和黄娜。感谢就职于骏龙科技的“师弟”张亚峰一直以来给予的技术支持。另外,也要特别感谢著名电子网站 EDNChina 的王志华和庞玉婷,ChinaAET 的杨晖、曾超宇、乐卫平、陈颖莹和贾志梅,电子发烧友的高美笛和钱姗姗,你们为广大网友们提供了一个非常友好的交流互动平台。

非常荣幸能够邀请到 Altera 公司的大学计划中国区经理陈卫中先生作序,在此表示由衷的感谢。最后,要向我的家人致敬,你们是我前进路上的最坚强后盾。尤其要感谢我的妻子一直以来所给予我的鼓励和支持。

吴厚航

2017 年 11 月于上海

序 言

很高兴能为吴厚航(网络大名:特权同学)的《FPGA/CPLD 边练边学——快速入门 Verilog/VHDL》一书写序言。其实我一直 是特权同学的粉丝,很欣赏他的风格、他的执着,诚恳和幽默诙谐的语言很能打动我。特权同学能在这么短的时间就把 FPGA 给玩儿“转”了,可见他是着实下了一番苦功的。

这本书从讲故事开始,历数可编程器件的发展。即便你对 FPGA/CPLD 一窍不通,在你看完这本书之后,也会对当今非常火爆的 FPGA/CPLD 有一个全面的了解。哪怕把它当小说看,也会把一个“菜鸟”引领到神奇的数字世界里。

本书非常适合初学者,尤其是在大学里刚刚开始学数字逻辑课程的学生们。首先,在对可编程器件一无所知的情况下,这本书会给你一个清晰的概览。其次,只有走过一遍流程,才会对硬件描述语言 Verilog/VHDL 和 Quartus II 软件有一个概念上的了解,也因之特权同学才会给它起书名叫:边练边学。

我极力推荐这本书,同时也希望特权同学能再接再厉,不断地写出更优秀的作品。

陈卫中

Altera 公司大学计划中国区经理

2013 年 8 月于成都

目 录

第 1 章 可编程器件发展简史与基本概念	1
1.1 可编程器件的由来与发展	1
1.2 FPGA/CPLD 与 Verilog/VHDL	3
1.3 设计方式与工具链	4
1.4 应用领域和发展趋势	6
第 2 章 实验平台板级设计	8
2.1 FPGA/CPLD 板级电路设计五要素	8
2.1.1 能量供应——电源电路	9
2.1.2 心脏跳动——时钟电路	11
2.1.3 状态初始——复位电路	12
2.1.4 灵活定制——配置电路	13
2.1.5 自由扩展——外设电路(I/O 应用)	14
2.2 CPLD 实验板 DIY	14
2.2.1 读懂器件手册	14
2.2.2 CPLD 核心电路设计	19
2.2.3 外设扩展电路设计	25
2.2.4 I/O 引脚分配	30
第 3 章 数字电路基础	36
3.1 0 和 1——精彩世界由此开始	36
3.2 表面现象揭秘——逻辑关系	39
3.3 内里本质探索——器件结构	43
第 4 章 Verilog 与 VHDL 语法基础	48
4.1 语法学习的经验之谈	48

4.2	可综合的语法子集	50
4.2.1	可综合的 Verilog 语法	51
4.2.2	可综合的 VHDL 语法	56
4.3	代码风格与书写规范	60
4.3.1	代码书写规范	61
4.3.2	代码风格	63
第 5 章	第一个完整的工程实践案例	81
5.1	软件开发平台搭建	81
5.1.1	软件下载和 License 申请	81
5.1.2	Quartus II 的安装	85
5.1.3	ModelSim 的安装	88
5.2	基本开发流程概述	90
5.3	第一个工程实例	91
5.3.1	工程创建与设计输入	92
5.3.2	行为仿真	97
5.3.3	引脚分配与编译	104
5.3.4	门级仿真	106
5.3.5	板级调试	109
第 6 章	基础实验与拓展练习	112
6.1	基于时钟分频的 PWM 发生器	112
6.1.1	实验原理分析	112
6.1.2	Verilog 参考实例	113
6.1.3	VHDL 参考实例	114
6.1.4	仿真验证与板级调试	115
6.1.5	实验流程与注意事项	117
6.1.6	拓展练习	120
6.2	经典的按键消抖实例	121
6.2.1	实验原理分析	121
6.2.2	Verilog 参考实例	123
6.2.3	VHDL 参考实例	124
6.2.4	仿真验证与板级调试	126
6.2.5	实验流程与注意事项	130
6.2.6	拓展练习	131
6.3	基于 Johnson 计数器的流水灯实验	131

6.3.1	实验原理分析	131
6.3.2	Verilog 参考实例	132
6.3.3	VHDL 参考实例	135
6.3.4	仿真验证	138
6.3.5	实验流程与注意事项	143
6.3.6	拓展练习	143
6.4	数码管驱动显示实验	144
6.4.1	实验原理分析	144
6.4.2	Verilog 参考实例	146
6.4.3	VHDL 参考实例	148
6.4.4	仿真验证	149
6.4.5	实验流程与注意事项	152
6.4.6	拓展练习	154
6.5	乘法器设计实验	155
6.5.1	实验原理分析	155
6.5.2	Verilog 参考实例	156
6.5.3	VHDL 参考实例	157
6.5.4	仿真验证	158
6.5.5	实验流程与注意事项	161
6.5.6	拓展练习	161
6.6	VGA 显示驱动实验	161
6.6.1	实验原理分析	161
6.6.2	Verilog 参考实例	163
6.6.3	VHDL 参考实例	165
6.6.4	仿真验证	168
6.6.5	实验流程与注意事项	169
6.6.6	拓展练习	170
6.7	UART 串口收发实验	170
6.7.1	实验原理分析	170
6.7.2	Verilog 参考实例	171
6.7.3	VHDL 参考实例	180
6.7.4	仿真验证	188
6.7.5	实验流程与注意事项	190
6.7.6	拓展练习	191
6.8	PS/2 键盘解码实验	192
6.8.1	实验原理分析	192

6.8.2	Verilog 参考实例	194
6.8.3	VHDL 参考实例	198
6.8.4	仿真验证	203
6.8.5	实验流程与注意事项	207
6.8.6	拓展练习	207
6.9	基于 I ² C 通信的 EEPROM 读/写实验	208
6.9.1	实验原理分析	208
6.9.2	Verilog 参考实例	210
6.9.3	VHDL 参考实例	222
6.9.4	仿真验证	235
6.9.5	实验流程与注意事项	241
6.9.6	拓展练习	241
6.10	SRAM 读/写测试实验	242
6.10.1	实验原理分析	242
6.10.2	Verilog 参考实例	244
6.10.3	VHDL 参考实例	247
6.10.4	仿真验证	251
6.10.5	实验流程与注意事项	253
6.10.6	拓展练习	254
第 7 章	器件资源应用实例	255
7.1	MAX II 内部振荡时钟使用实例	255
7.2	MAX II 的 UFM 模块使用实例	258
	参考文献	263

第 1 章

可编程器件发展简史与基本概念

1.1 可编程器件的由来与发展

20 世纪 60 年代中期, TI 公司设计制造了各式各样的实现基本逻辑门电路功能的芯片, 相信今天很多的工程师仍然很熟悉这些主要面对军工应用的 54xx 和面对商业应用的 74xx 芯片。据说早期的工程师甚至能够单凭着这些芯片就可架构出简单的 CPU 功能。还真别小瞧这些基本逻辑门电路, 俗话说“万丈高楼平地起”, 如果说今天在嵌入式领域呼风唤雨的各种功能强大的 ARM7、ARM9、DSP 是万丈高楼, 那么称这些基本的逻辑门电路为一砖一瓦倒是一点也不为过。

从 1971 年 Intel 公司的第一颗 4 位微处理器 Intel 4004 到 20 世纪 80 年代初被奉为经典的 8051 单片机, 再到今天各大嵌入式处理器厂商竞相使用的由 ARM 公司推出的各种 Cortex 内核, 嵌入式处理器的发展不可不说是翻天覆地。不过, 如果深入处理器的底层结构, 会发现它们最本质的东西并没有太大的改变。而处理器再强大, 一颗芯片尽可以将各种外设嵌入其中, 但对于任何一颗已经批量生产的芯片而言, 它的功能是固定的, 若想在既有外设功能的基础上有任何的扩展, 或许不是遇到电气特性不支持就是遇到 I/O 太少的尴尬, 而这些问题也就催生了可编程逻辑器件的诞生。今天的 CPU 周围已很难看见 54 或 74 字样的 ASIC 了, 取而代之的可能是引脚密集的 CPLD 或 FPGA, 因为在系统的可扩展性和灵活性方面, FPGA/CPLD 有着得天独厚的优势。当然, 今天动辄上百万门的 FPGA 器件可不是为干这点活而制造的, 它更多地被应用到了通信、数据采集、网络等对数据传输速率和吞吐量有更高要求的场合。

今天大家熟知的 FPGA/CPLD 也不是一开始就有的, 第一款可编程逻辑器件 (PLD) 最初是在 1970 年以 PROM 的形式进入人们视野的, 这种 PROM 结构的可编程逻辑器件可以实现简单的逻辑功能, 很容易替代当时流行的 54 系列或 74 系列逻辑门电路。

受限于 PROM 的结构, 第一款可编程逻辑器件输入相对较少, 因此, 可编程逻辑阵列 (PAL) 便应运而生。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。PAL的问题在于其实现方式使得信号通过可编程连线的�时间相对较长。在PAL的基础上,又发展了一种通用阵列逻辑GAL,要比PAL速度快许多。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,因而其设计具有很强的灵活性,至今仍有许多人使用。

这些早期的PLD器件的一个共同特点是可以实现速度特性较好的逻辑功能,但其过于简单的结构也使它们只能实现规模较小的电路。电子领域的发展趋势总是朝着速度更快、功能更强、体积更小、成本更低的方向迈进。复杂可编程逻辑器件(CPLD)的诞生也就顺理成章了。Altera公司于1984年发明了CMOS和EPROM技术相结合的CPLD。CPLD可实现的逻辑功能相比PAL和GAL有了大幅度的提升,已经可以胜任设计中复杂性较高、速度也较快的逻辑功能,尤其在接口转换、总线控制和扩展方面有着较多的应用。经过几十年的发展,今天的CPLD功能和性能也得到了进一步的提升,其基本结构由可编程I/O单元、基本逻辑单元、布线池以及其他相关辅助功能块组成。Altera、Xilinx和Lattice是主要的CPLD供应商。

其实无论是前面提到的PAL、GAL,还是CPLD,要实现大规模的复杂逻辑电路都显得无能为力。而ASIC的设计既耗时又费钱,而且功能固定,在流片后很难随意更改。鉴于此,Xilinx创始人之一Ross Freeman发明了现场可编程门阵列FPGA,FPGA是一块全部由“开放式门”组成的计算机芯片。采用该芯片,工程师可以根据需要进行灵活编程,添加各种新功能,以满足不断发展的协议标准或规范,甚至可以在设计的最后阶段对它进行修改和升级。Freeman先生当时就推测低成本、高灵活性的FPGA将成为各种应用中定制芯片的替代品。也正是由于此项伟大的发明,让Freeman先生于2009年荣登美国发明家名人堂。

从发明FPGA至今,伴随着制造工艺的不断进步,FPGA在深亚微米甚至深亚纳米时代一直走在了创新第一线。如今的FPGA器件,其组成不限于基本的可编程I/O单元、可编程逻辑单元、丰富的布线资源,而且还拥有灵活的时钟管理单元、嵌入式块RAM以及各种通用的内嵌功能单元,很多器件还顺应市场需求内嵌专用的硬件模块。近些年来,可编程器件的龙头老大Xilinx公司和Altera公司更是相继推出了硬核CPU+FPGA的产品(Xilinx公司的Zynq和Altera公司的Soc FPGA),此举大有单芯片横扫千军的架势。

电子行业在继续挑战摩尔定律的征程中,不仅是可编程器件继续大放光彩,而且ASIC也能够重获新生。可编程器件,尤其是FPGA/CPLD的发明和大量应用已经足够让我们肃然起敬。相信对于很多即将或者已经走上电子硬件设计道路的同仁们,对可编程器件的了解、熟悉甚至精通是提升自身技术能力的基本技能之一。

1.2 FPGA/CPLD 与 Verilog/VHDL

1. FPGA 与 CPLD

尽管很多人听说过 FPGA 和 CPLD,但是关于 FPGA 与 CPLD 之间的区别,了解的人可能不是很多。虽然 FPGA 与 CPLD 都是“可反复编程的逻辑器件”,但在技术上却有一些差异。简单地说,FPGA 就是将 CPLD 的电路规模、功能、性能等方面强化之后的产物。

一般而言,FPGA 与 CPLD 之间的区别如表 1.1 所列。

表 1.1 FPGA 和 CPLD 的比较

项 目	CPLD	FPGA
组合逻辑实现方式	乘积项(product term)和查找表(LUT, look up table)	查找表(LUT,look up table)
存储结构	非易失性(Flash,EEPROM)	易失性(SRAM)
特点	<ul style="list-style-type: none"> ● 非易失性:即使切断电源,电路上的数据也不会丢失。 ● 立即上电:上电后立即开始工作。 ● 可在单芯片上工作 	<ul style="list-style-type: none"> ● 内建高性能硬核功能。 <ul style="list-style-type: none"> - PLL、DLL; - 存储器模块; - DSP 模块; - 高速串行收发器。 ● 用最先进的技术实现高集成度和高性能。 ● 需要外部配置 ROM
应用领域	偏向于简单的控制应用以及粘合逻辑	偏向于较复杂且高速的控制通道应用以及数据处理
集成度	中小规模	中大规模

总而言之,FPGA 和 CPLD 最大的区别是其存储结构不同,这同时也决定了它们的规模不同。但是从使用和实现的角度来看,其实它们所使用的语言以及开发流程的各个步骤几乎是一致的。对于大多数的初学者来说,是学 FPGA 还是学 CPLD 都是没有问题的,只要掌握它们的开发设计流程,就会发现它们的应用方法和技巧都是相通的。对于大多数电子工程师来说,将来不一定都有机会用 FPGA 做一些高性能的产品,但是用一颗小小的 CPLD 来实现一些辅助开发倒是很有可能的。

2. Verilog 与 VHDL

Verilog 和 VHDL,它们的历史渊源、孰优孰劣这里不再介绍。美国和中国台湾地区的逻辑设计公司大都以 Verilog 语言为主,国内目前学习和使用 Verilog 的人数也在逐渐超过 VHDL。从高等校院刚毕业的大学生大都熟悉 VHDL,估计是因为国

内高等院校的老教师们接触 VHDL 早一些,所以学校里开设 VHDL 课程也多一些。但是从学习的角度来讲,Verilog 相对于 VHDL 有着快速上手、易于使用的特点,因此获得了更多工程师的青睐。即使是学校里没接触过 VHDL 的初学者,只要凭着一点 C 语言的底子加上一些硬件基础,三两个月就可以熟悉 Verilog 语法。当然,仅仅入门还是远远不够的,要想真正掌握 Verilog 必须花很多时间和精力,再加上一些项目的实践,才会慢慢对可编程逻辑器件的设计有更深入的理解和认识。

1.3 设计方式与工具链

FPGA/CPLD 的设计方式和大多数的嵌入式微处理器类似,首先要有一台电脑,其次要搭建一块有 FPGA/CPLD 的电路板并且预留出供代码烧录的调试接口。通俗来理解 FPGA/CPLD 的开发,也就是在 PC 机的各种工具上做好设计,最终产生一个 bit 流(就是一个充满 0 和 1 的只有我们的 FPGA/CPLD 器件才能够读懂的“机器语言”),通过下载线把这个 bit 流再送给 FPGA/CPLD 运行。说着简单,其实也着实不易,首先是 FPGA/CPLD 的设计输入有自己专门的语言(HDL,即硬件描述语言),目前应用最广泛的是 VHDL 和 Verilog。无论是 VHDL 还是 Verilog,尤其 Verilog 语法和 C 很相似,整个思想上完全是对过往软件语言的一种颠覆。软件编程是顺序思想,即便是操作系统根本上也还是顺序执行的,而 HDL 完全不同,在 FPGA/CPLD 器件中实现的功能是并行的(当然,也可以让它顺序地干活,毕竟软件都是在硬件基础上实现的,单纯软件能干的活,硬件没有干不了的,因为本来就是硬件在干;而硬件能干的活,软件就不见得能胜任了)。

再来看看 FPGA/CPLD 一般的开发流程。如图 1.1 所示,这也许算不上一个完全意义上完整的开发流程,但是这些步骤绝对都是 FPGA 开发流程中不可缺少的部分(行为仿真和时序仿真有时只是作为可选步骤),其他的设计方法、技巧都是贯穿于其中的。

此外,FPGA 设计的最大特点就是迭代性很强,并不是一个简单的顺序流程。在开发设计过程中,设计者在测试验证中一旦发现问题,往往需要回到前面的步骤重新审查、修改,然后重新综合、实现、仿真验证,直到最终的设计符合需求。

看到这些错综复杂的框图和连线,希望不会吓到曾经热情高涨的初学者们。路是一步一步走出来的,只要用心,通过本书后面各个章节的学习,相信入门绝不是难事。

接下来,我们会讨论一些关于 FPGA 工具链的话题。这里所谓的工具,指的是针对 FPGA 开发流程所需要涉及的各种 EDA 工具。这部分知识只是作为一个参考,希望有志于从事可编程器件相关工作的同仁们对本行业内的生态链也有所了解。下文所列出的很多供应商以及他们提供的 EDA 工具很有可能随着时代的变迁逐渐淡出历史舞台——消失在人们的视野中;也很可能在读者阅读本书时,更多新的

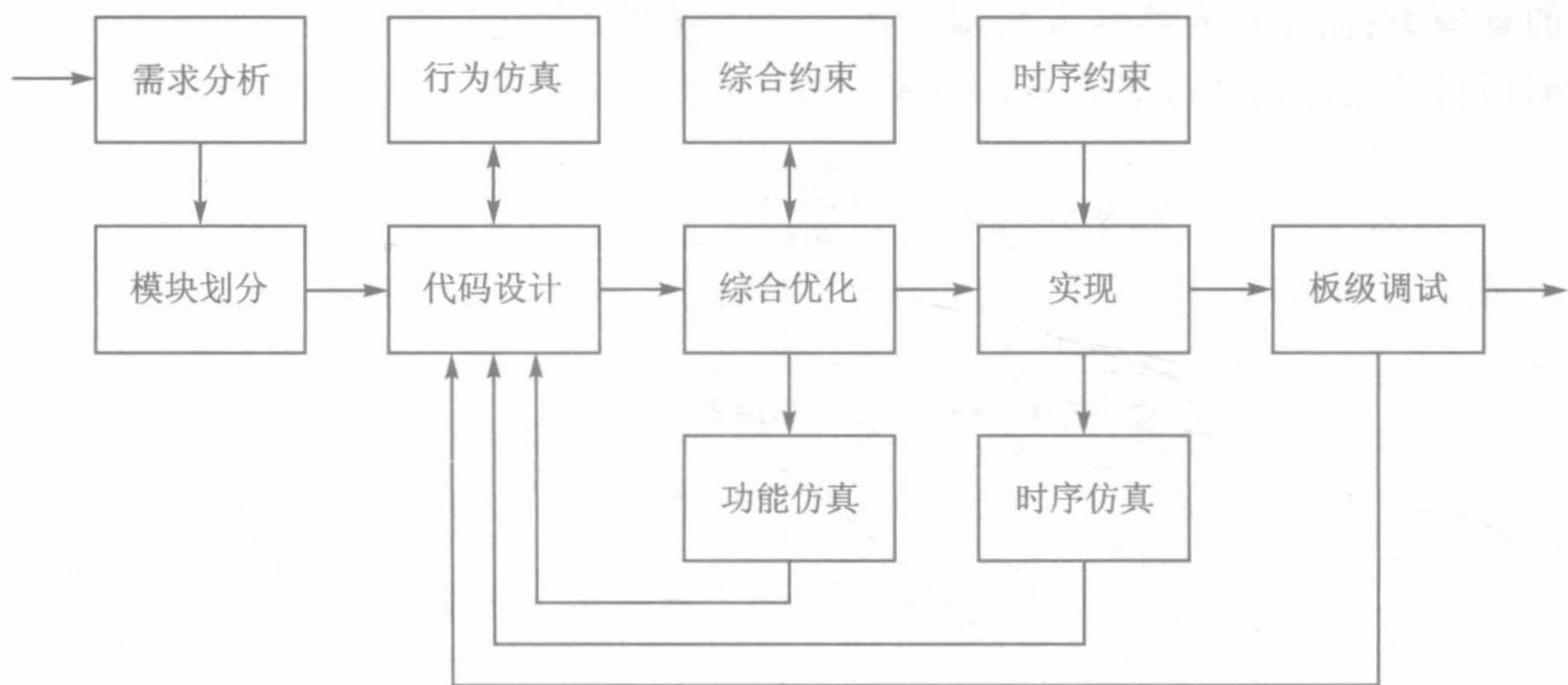


图 1.1 FPGA/CPLD 开发流程

EDA 工具和厂商不断地涌现。这些都很正常,因为从已往几十年 EDA 工具的更迭来看,能够长期稳定地在这份榜单中占有一席之地真的不是一件容易的事。

首先,我们需要罗列从事和制造可编程逻辑器件(这里主要针对 FPGA/CPLD 而言)的供应商,这些供应商通常都会有针对自己器件的比较完备的集成的 EDA 开发工具。如表 1.2 所列,这些应该是目前大家比较熟悉并且网络上搜索得到的 FPGA/CPLD 供应商,而至于那些网络上都很难寻觅到的供应商(这样的 FPGA/CPLD 供应商还是有不少的),笔者也实在无法一一罗列。

表 1.2 FPGA/CPLD 主要供应商

公司	网址	产品	工具
Actel	www.actel.com	FPGA	Libero IDE
Altera	www.altera.com	FPGA/CPLD	Quartus II
Lattice	www.latticesemi.com	FPGA/CPLD	Diamond
Xilinx	www.xilinx.com	FPGA/CPLD	ISE

在这份列表中,四家公司的产品各有千秋。众所周知,Altera 和 Xilinx 占据了这个市场的大半壁江山,应该可以毫不夸张地说,是他们一直在引领着这个行业的前进和发展。而余下的厂商们也各有千秋,Actel 主打非易失性、低功耗及混合信号的 FPGA 器件,Actel 的反熔丝技术是赫赫有名的,而且在航天航空领域有着非常好的客户群;Lattice 在中低端应用中蚕食了很大一部分的市场份额。

如图 1.2 所示,我们可以看到 FPGA/CPLD 开发过程中涉及多个步骤并会有不同的网表生成,这些不同的网表使得使用官方既有开发工具的过程中也能够支持不同的第三方 EDA 工具。综合、布局布线、仿真、时序设计,虽然这几个步骤在一般的开发过程中都能够使用器件原厂提供的集成开发工具完成,但是也可以交给不同的 EDA 工具来完成。一般而言,尤其是综合、仿真和时序设计,第三方的一些工具有更

强的竞争力;而布局布线这方面则是器件原厂来得更优化一些。无论如何,用户可以根据项目的实际需要或者自身的开发习惯,选择不同的工具帮助完成设计。

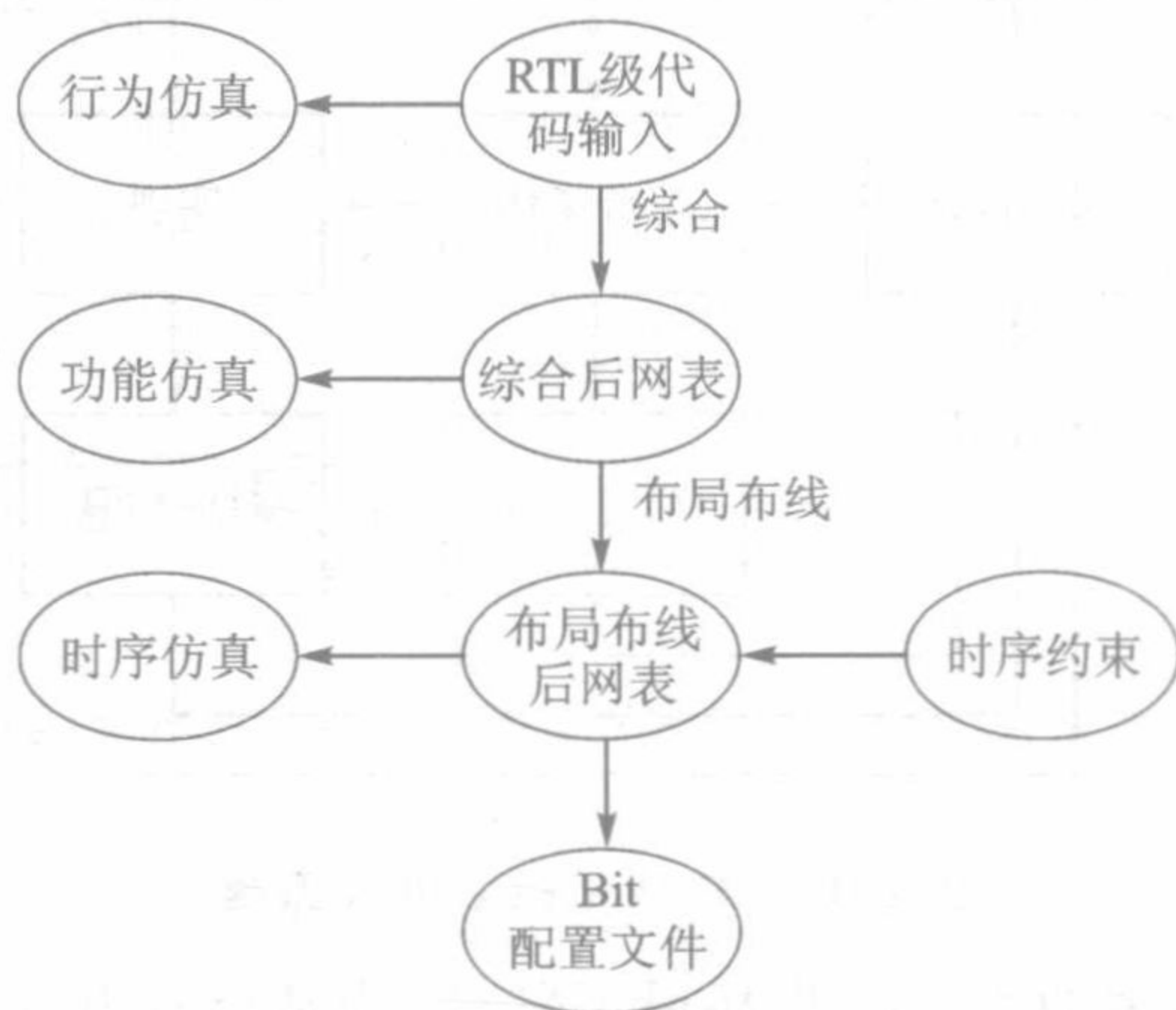


图 1.2 FPGA/CPLD 网表生成

表 1.3 罗列出了目前比较常见的 FPGA/CPLD 开发使用的第三方综合、仿真工具。

表 1.3 FPGA/CPLD 主要第三方 EDA 工具

公司	网址	产品	功能
Aldec	www.aldec.com	Active-HDL 等	仿真工具
Model Tech	www.model.com	ModelSim	仿真工具
Synopsys	www.synopsys.com	VCS	仿真工具
Synopsys	www.synopsys.com	Synplify Pro 等	综合工具
Mentor	www.mentor.com	LeonardoSpectrum	综合工具

1.4 应用领域和发展趋势

FPGA/CPLD 目前虽然还受制于较高的开发门槛以及器件本身高昂的价格,从应用的普及率上看和 ARM、DSP 还是有一定的差距的,但是在非常多的应用场合,工程师们还是会别无选择地使用它。FPGA/CPLD 所固有的灵活性和并行性是其其他芯片所不具备的,所以它的应用领域涵盖得很广,如图 1.3 所示。从技术角度来看,主要有以下需求的应用场合:

- 逻辑粘合,如一些嵌入式处理常常需要地址或外设扩展,CPLD 器件尤其适合。已经少有项目会选择一颗 FPGA 器件专门用于逻辑粘合的应用,但是在已经使用的 FPGA 器件中顺便做些逻辑粘合的工作倒是非常普遍。

- 实时控制,如液晶屏或电机等设备的驱动控制,此类应用也以 CPLD 或低端 FPGA 为主。
- 高速信号采集和处理,如高速 A/D 前端或图像前端的采集和预处理,近年来持续升温的机器视觉应用也几乎是无一例外地都使用了 FPGA 器件。
- 协议实现,如更新较快的各种有线和无线通信标准、广播视频及其编解码算法、各种加密算法等,使用 FPGA 比 ASIC 更有竞争力。
- 各种原型验证系统,由于工艺的提升,流片成本也不断攀升,而在流片前使用 FPGA 做前期的验证已成为非常流行的做法。
- 片上系统,如 Altera 公司的 Soc FPGA 和 Xilinx 公司的 Zynq,这类 FPGA 器件,既有成熟的 ARM 硬核处理器,又有丰富的 FPGA 资源,大有单芯片一统天下的架势。

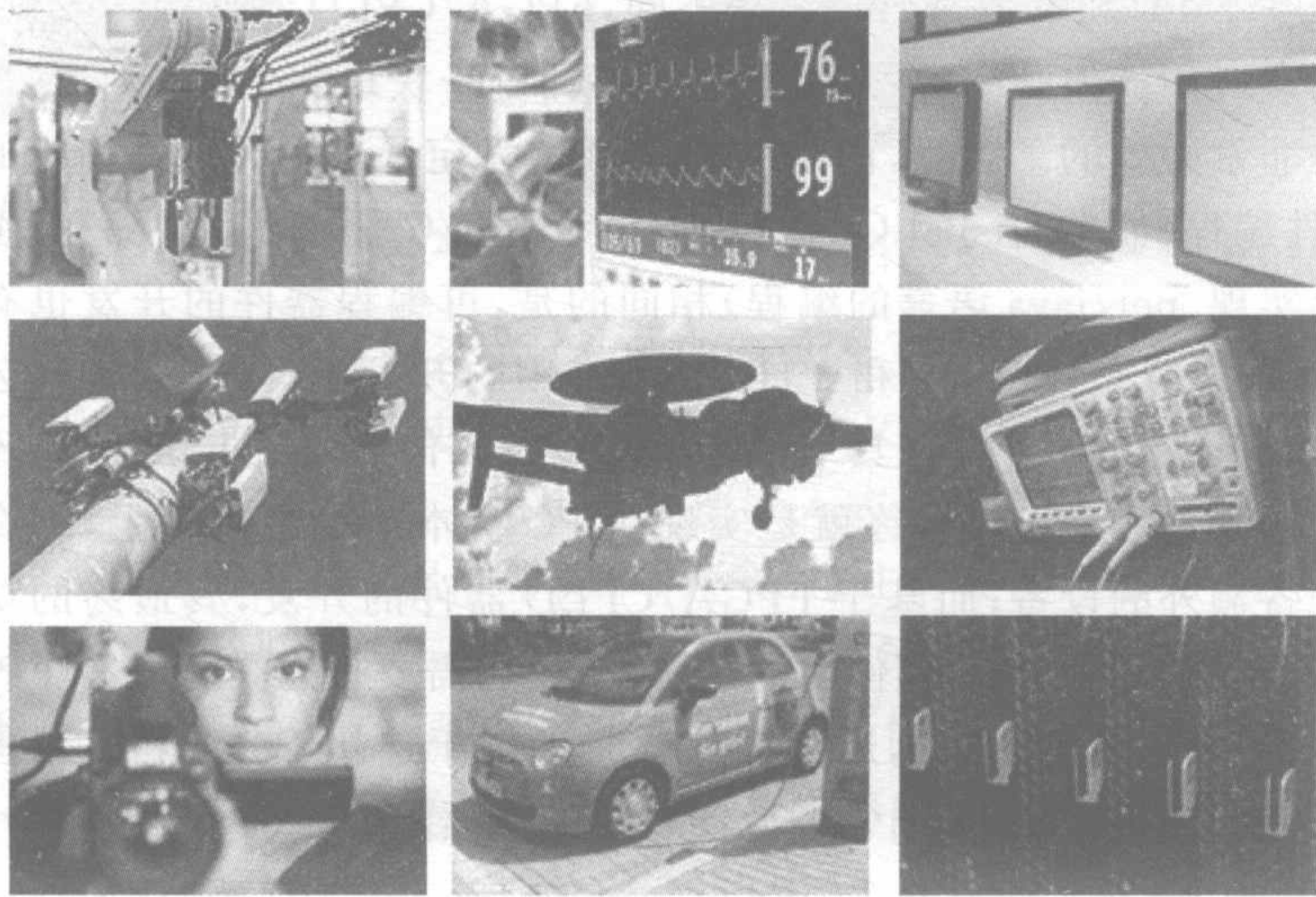


图 1.3 FPGA 应用精彩纷呈

当然,若从具体的应用领域来看,FPGA 在电信、无线通信、有线通信、消费电子产品、视频和图像处理、车载、航空航天和国防、ASIC 原型开发、测试测量、存储、数据安全、医疗电子、高性能计算以及各种定制设计中都有涉猎。总而言之,FPGA 所诞生并发展的时代是一个好时代,与生俱来的一些特性也注定了它将会在这个时代的大舞台上大放光彩。

思考

1. FPGA 和 CPLD 到底有何异同呢?
2. 使用 FPGA/CPLD 器件需要具备怎样的一个软件开发环境?