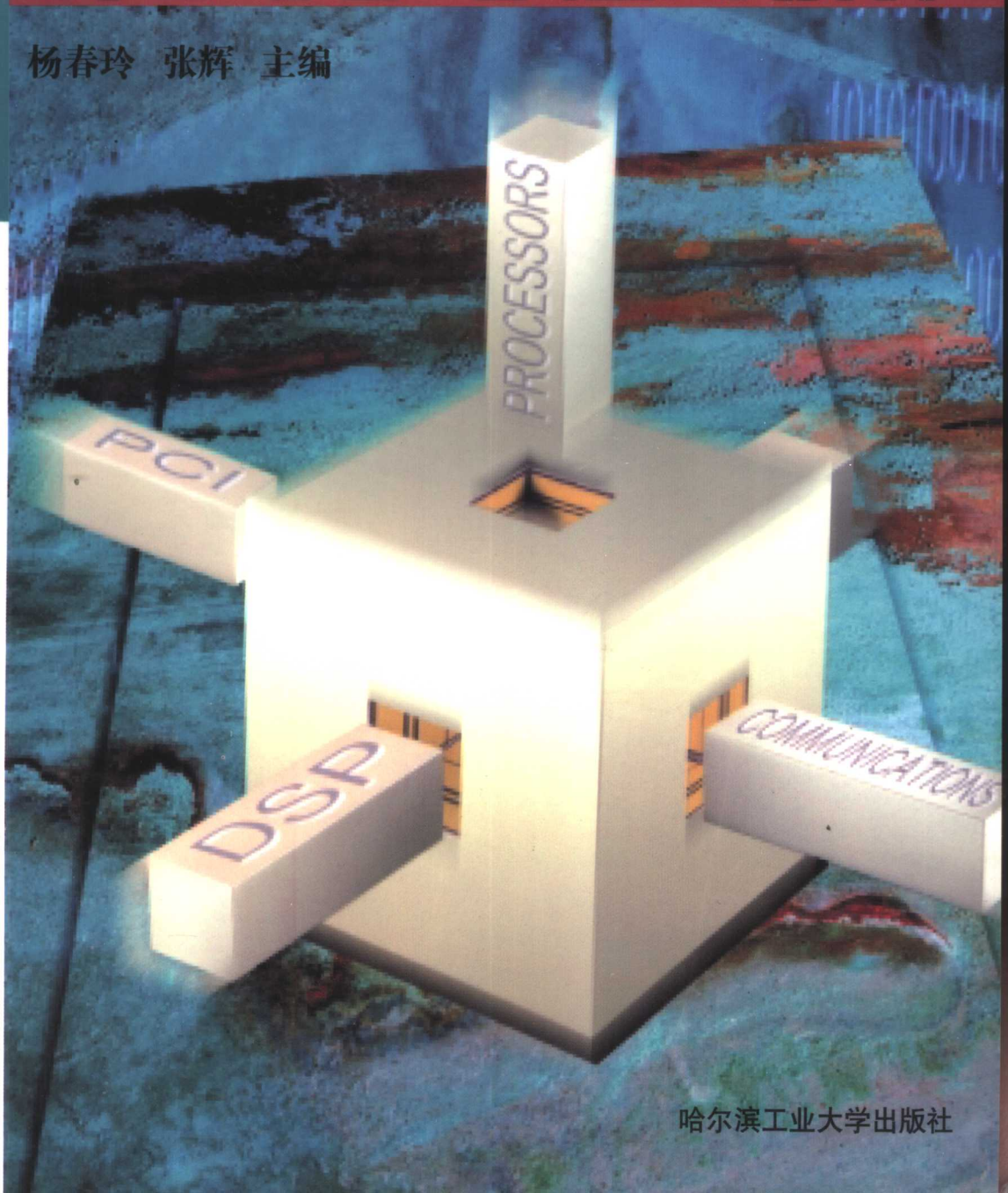


现代可编程逻辑器件 及SOPC应用设计

杨春玲 张辉 主编



哈尔滨工业大学出版社

现代可编程逻辑器件及 SOPC 应用设计

杨春玲 张辉 主编

哈尔滨工业大学出版社
哈尔滨

内 容 简 介

本书根据课堂教学和实验要求,以提高实际工程设计能力为目的,对目前各种 PLD 器件及编程方法做了系统和完整的介绍。

全书共分 8 章,详细介绍了 EDA 设计方法、各种 PLD 器件的结构原理、VHDL 的设计优化、可编程逻辑器件的各种开发环境以及基于 PLD 器件的典型设计项目。另外对集成了 DSP、MCU、存储器的片上可编程系统(SOPC)的结构和应用也做了介绍。

本书可作为高等院校电子工程、通信、工业自动化、计算机应用技术、电子对抗、仪器仪表、数字信号或图像处理等专业的本科生或研究生的电子技术或可编程技术的教材及实验指导书,也可作为相关专业技术人员的参考书。

图书在版编目(CIP)数据

现代可编程逻辑器件及 SOPC 应用设计/杨春玲, 张辉
主编. 哈尔滨: 哈尔滨工业大学出版社, 2005.3

ISBN 7-5603-2137-2

I. 现… II. ①杨… ②张… III. 可编程逻辑器件
IV. TP332.1

中国版本图书馆 CIP 数据核字(2005)第 019690 号

出版发行 哈尔滨工业大学出版社

社 址 哈尔滨市南岗区复华四道街 10 号 邮编 150006

传 真 0451-86414749

印 刷 肇东粮食印刷厂

开 本 787×1092 1/16 印张 20 字数 458 千字

版 次 2005 年 3 月第 1 版 2005 年 3 月第 1 次印刷

书 号 ISBN 7-5603-2137-2/TP·214

印 数 1~3 000

定 价 30.00 元

前 言

可编程逻辑器件(PLD)是 20 世纪 70 年代发展起来的新型逻辑器件,可以完全由用户配置以完成某种特定的逻辑功能。经过 80 年代的发展,PLD 行业初步形成,而进入 90 年代以后,PLD 已成为半导体领域中发展最快的产品之一。

可编程逻辑器件是在 ASIC 设计的基础上发展起来的,在 ASIC 设计方法中,通常采用全定制和半定制电路设计方法,但设计完成后如果不能满足要求,还要重新设计再进行验证。这样不但会导致设计开发周期变长,产品上市时间(Time to Market)难以保证,而且会大大增加产品的开发费用。

这一弊端为可编程逻辑器件的发展提供了动力,从 ASIC 设计的风险来看,可编程逻辑器件正好解决了这一问题。随着工艺、技术及市场的不断发展,PLD 产品的价格越来越低、集成度越来越高、速度越来越快,并在越来越多的领域中取代了 ASIC。Dataquest 数据显示,2002 年 PLD 的年销售额已达 47 亿美元,并成为半导体市场中成长最快的领域。

PLD 产品的优势在于可以缩短开发周期、现场灵活性好、开发风险小,且随着芯片制造工艺的不断进步,单片集成度飞速提高,价格也越来越高,已广泛应用于电子、通信、航天及军事等领域。

集成度的不断提高使得产品的性能不断提高,功能不断增多。最早的 PLD 仅仅能够实现一些简单的逻辑功能,而现在片上可编程系统直接实现系统集成,在速度上可以满足一般系统的要求。其好处是用户把所有关键的功能块放上去后,可以随着标准的改变而重新配置,而且可以降低费用,缩短开发时间。同时,IP(Intelligent Property)核在 PLD 中的使用也使得 SOPC 成为可能。2000 年,Altera 推出了高密度的 APEX 系列器件和开发工具 Quartus II,再加上一系列可重复使用的 IP 核,SOPC 开始有了实质性的发展。之后,Altera 从 MIPS 和 ARM 公司获得了处理器层次结构的许可权,这两种芯核纳入 Altera 的 IP 核,以便将晶体管级处理器内核嵌入自己的 CPLD 层次结构中,并针对 SOPC 推出了一套嵌入式处理器产品,它包括 ARM、MIPS 和 Nios 处理器三种芯核以及相应的开发工具。

不久,另一 PLD 厂商 Xilinx 公司也宣布推出了新型的 SOPC 方案。Xilinx 与 ARCCores 共同为 ARC 的 32 位可配置处理器的用户提供可编程逻辑解决方案。ARC 的 32 位可配置处理器将被用于 Xilinx Virtex 和 Spartan II 系列的 FPGA 中。ARC 处理器与传统 RISC 处理器不同,它的指令集可以根据用户的确切需要进行配置。ARC 工具集允许按处理器规格改制,使其在较低的时钟频率下具有更好的系统性能和更低的功耗,成为理想的以 FPGA 为基础的软件解决方案。

PLD 的飞速发展使得传统设计方法及工具逐渐被抛弃,新器件、新理念、新思路的掌握与理解对于当代有志于 PLD 技术的电子工程师来说是一个新的挑战,掌握 SOPC 开发技术、VHDL 语言与 FPGA 开发技术是每一位电子设计工程师重要而紧迫的任务。为在我国电子设计,电子产品开发,ASIC/SOPC 设计开发,IP Core 应用和开发,具有自主知识产权的电子系统的开发(为适应我国进入 WTO 的新形势),大学生电子设计创新能力

的培养以及电子、信息、通信、电子对抗、工控类高等教学领域深化推广 EDA 和 SOC 技术, 为适应高技术人才市场的需求以及促进高新技术的推广应用, 我们编写了本书。

全书共分 4 篇、10 章, 在编写中力求准确并注意系统性, 具有如下特色:

(1) 覆盖面广, 包括了目前工程实际中应用的可编程逻辑器件及开发环境。

(2) 内容新, 包括了目前最新的可编程逻辑器件和开发环境。

(3) 通俗易懂, 由浅入深, 通过实例讲解了 VHDL 语言以及各种器件的编程方法。

(4) 工程实践性强, 教材的设计样例来自于工程实际, 可以很好地培养学生的动手实践能力。

参加本书编写的有杨春玲、张辉、赵东阳、梁泉, 由于时间仓促, 难免有疏漏之处, 请读者予以指正。

另外, 本书在编写的过程中得到了杭州电子科技大学潘松教授的帮助, 在此深表谢意。

由于可编程逻辑器件的发展, 相应的教学内容和教学方法也应不断改进, 其中一定有许多问题值得深入探讨, 敬请提出宝贵意见。

编 者

2005 年 3 月

目 录

第 1 章 可编程逻辑器件概述	1
1.1 EDA 技术的发展概况	1
1.2 可编程逻辑器件的发展概况	5
1.3 硬件描述语言 VHDL	7
1.4 EDA 与传统电子设计方法	9
1.5 IP 核	19
第 2 章 GAL/CPLD 器件结构与应用	21
2.1 概述	21
2.2 只读存储器 ROM	22
2.3 PLD 的分类和特点	26
2.4 GAL 的结构	26
2.5 CPLD 器件简介	31
2.6 CPLD 的编程与配置	42
第 3 章 现场可编程门阵列	46
3.1 概述	46
3.2 FPGA 的基本结构	46
3.3 FPGA 的配置	62
第 4 章 VHDL 设计初步	66
4.1 VHDL 概述	66
4.2 译码器与编码器	78
4.3 比较器	84
4.4 数码转换电路	88
4.5 算术运算	93
4.6 计数器	100
4.7 移位寄存器	109
4.8 状态机	112

第 5 章 VHDL 语法	123
5.1 VHDL 的结构.....	123
5.2 VHDL 的语言元素.....	137
5.3 VHDL 的基本语句.....	148
第 6 章 可编程逻辑器件的开发环境	179
6.1 可编程数字系统设计的输入.....	179
6.2 Altera 开发环境 MAX+plus II 使用入门.....	180
6.3 Quartus II 软件的使用.....	188
6.4 Xilinx 开发环境 Foundation 使用入门.....	207
6.5 ISE 集成开发环境使用入门.....	221
第 7 章 DSP Builder 及 SOPC	232
7.1 DSP Builder.....	232
7.2 Nios 嵌入式 CPU 核.....	249
第 8 章 数字系统设计实例	286
8.1 4 位十进制频率计设计.....	286
8.2 VGA 图像显示控制器设计.....	293
8.3 模拟波形发生器电路设计.....	297
8.4 通用异步收发器.....	301
8.5 电子琴电路.....	306
参考文献	311

第 1 章 可编程逻辑器件概述

内容提要：EDA 技术的发展概况，可编程逻辑器件的发展概况，硬件描述语言 VHDL，EDA 与传统电子设计方法，IP 核的概念。通过本章的学习，可使大家对现代电子设计有一个初步的了解。

1.1 EDA 技术的发展概况

电子设计自动化（EDA，Electronic Design Automation）是指利用计算机完成电子系统的设计。EDA 技术是以计算机和微电子技术为先导，汇集了计算机图形学、拓扑、逻辑学、微电子工艺与结构学和计算数学等多种计算机应用学科最新成果的先进技术。

EDA 技术以计算机为工具，代替人完成数字系统的逻辑综合、布局布线和设计仿真等工作。设计人员只需完成对系统功能的描述，就可以由计算机软件进行处理，得到设计结果，修改设计如同修改软件一样方便，极大地提高了设计效率。

从 20 世纪 60 年代中期开始，人们就不断地开发出了各种计算机辅助设计工具来帮助设计人员进行电子系统设计。电路理论和半导体工艺水平的提高，对 EDA 技术的发展起到了巨大的推进作用，使 EDA 作用范围从 PCB 板设计延伸到电子线路和集成电路设计，直至整个系统的设计，也使 IC 芯片系统应用、电路制作和整个电子系统生产过程都集成在一个环境之中。根据电子设计技术的发展特征，EDA 技术发展大致分为三个阶段。

1. CAD 阶段（20 世纪 60 年代中期～ 80 年代初期）

这个阶段的特点是一些单独的工具软件，主要有 PCB（Printed Circuit Board）布线设计、电路模拟、逻辑模拟、版图的绘制等，通过计算机的使用，使设计人员从大量繁琐重复的计算和绘图工作中解脱出来。例如，目前常用的 Protel 早期版本 Tango，以及用于电路模拟的 PSPICE 软件和后来产品化的 IC 版图编辑与设计规则检查系统等软件，都是这个阶段的产品。这个时期的 EDA 一般称为 CAD（Computer Aided Design）。

20 世纪 80 年代初，随着集成电路规模的增大，EDA 技术有了较快的发展。许多软件公司如 Mentor、Daisy System 及 Logic System 等进入市场，开始供应带电路图编辑工具和逻辑模拟工具的 EDA 软件。这个时期的软件主要针对产品开发，按照设计、分析、生产、测试等多个阶段，不同阶段分别使用不同的软件包，每个软件只能完成其中的一项工作，通过顺序循环使用这些软件，可完成设计的全过程。但这样的设计过程存在两个方面的问题：第一，由于各个工具软件是由不同的公司和专家开发的，只解决一个领域的问题，若将一个工具软件的输出作为另一个工具软件的输入，就需要人工处理，过程很繁琐，影响

了设计速度；第二，对于复杂电子系统的设计，当时的 EDA 工具由于缺乏系统级的设计考虑，不能提供系统级的仿真与综合，设计错误如果在开发后期才被发现，将给修改工作带来极大不便。

2. CAE 阶段（20 世纪 80 年代初期～ 90 年代初期）

这个阶段在集成电路与电子设计方法学以及设计工具集成化方面取得了许多成果。各种设计工具，如原理图输入、编译与连接、逻辑模拟、测试码生成、版图自动布局以及各种单元库已齐全。由于采用了统一数据管理技术，因而能够将各个工具集成为一个 CAE（Computer Aided Engineering）系统。按照设计方法学制定的设计流程，可以实现从设计输入到版图输出的全程设计自动化。这个阶段主要采用基于单元库的半定制设计方法，采用门阵列和标准单元设计的各种 ASIC 得到了极大的发展，将集成电路工业推入了 ASIC 时代。多数系统中集成了 PCB 自动布局布线软件以及热特性、噪声、可靠性等分析软件，进而可以实现电子系统设计自动化。

3. EDA 阶段（20 世纪 90 年代以来）

20 世纪 90 年代以来，微电子技术以惊人的速度发展，其工艺水平达到深亚微米（Deep-Submicron）级，在一个芯片上可集成数百万乃至上千万只晶体管，工作速度可达到 GHz，这为制造出规模更大、速度更快和信息容量更大的芯片系统提供了条件，但同时也对 EDA 技术提出了更高的要求，并促进了 EDA 技术的发展。此阶段主要出现了以高级语言描述、系统仿真和综合技术为特征的第三代 EDA 技术，不仅极大地提高了系统的设计效率，而且使设计人员摆脱了大量的辅助性及基础性工作，将精力集中于创造性的方案与概念的构思上。下面简单介绍这个阶段 EDA 技术的主要特征。

（1）高层综合（HLS, High Level Synthesis）的理论与方法取得了较大的进展，将 EDA 设计层次由 RT 级提高到了系统级（又称行为级），并划分为逻辑综合和测试综合。逻辑综合就是对不同层次和不同形式的设计描述进行转换，通过综合算法，以具体的工艺背景实现高层目标所规定的优化设计，通过设计综合工具，可将电子系统的高层行为描述转换到低层硬件描述和确定的物理实现，使设计人员无需直接面对低层电路，不必了解具体的逻辑器件，从而把精力集中到系统行为建模和算法设计上。测试综合是以设计结果的性能为目标的综合方法，以电路的时序、功耗、电磁辐射和负载能力等性能指标为综合对象。测试综合是保证电子系统设计结果稳定可靠的必要条件，也是对设计进行验证的有效方法。其典型工具有 Synopsys 公司的 Behavioral Compiler 及 Mentor Graphics 公司的 Monet 和 Renoir。

（2）采用硬件描述语言（HDL, Hardware Description Language）来描述 10 万门以上的设计，并形成了 VHDL（Very high speed integrated circuit HDL）和 Verilog HDL 两种标准硬件描述语言。它们均支持不同层次的描述，使得复杂 IC 的描述规范化，便于传递、交流、保存和修改，也便于重复使用。它们多应用于 FPGA/CPLD/EPLD 的设计中。大多数 EDA 软件都兼容这两种标准。

（3）采用平面规划（Floorplaning）技术对逻辑综合和物理版图设计进行联合管理，

做到在逻辑综合早期设计阶段就考虑到物理设计信息的影响。通过这些信息,设计者能更进一步进行综合与优化,并保证所做的修改只会提高性能而不会给版图设计带来负面影响。这在深亚微米级布线延时已成为主要延时的情况下,加速设计过程的收敛与成功是有所帮助的。在 Synopsys 和 Cadence 等公司的 EDA 系统中均采用了这项技术。

(4) 可测性综合设计。随着 ASIC 的规模与复杂性的增加,测试难度与费用急剧上升,由此产生了将可测性电路结构制作在 ASIC 芯片上的想法,于是开发了扫描插入、BLST (内建自测试)、边界扫描等可测性设计 (DFT) 工具,并已集成到 EDA 系统中。其典型产品有 Compass 公司的 Test Assistant 和 Mentor Graphics 公司的 LBLST Architect、BSD Architect、DFT Advisor 等。

(5) 为带有嵌入 IP 模块 (IP 核) 的 ASIC 设计提供软硬件协同系统设计工具。协同验证弥补了硬件设计和软件设计流程之间的空隙,保证了软硬件之间的同步协调工作。协同验证是当今系统集成核心,它以高层系统设计为主导,以性能优化为目标,融合了逻辑综合、性能仿真、形式验证和可测性设计。产品如 Mentor Graphics 公司的 Seamless CAV。

(6) 建立并行设计工程 (CE, Concurrent Engineering) 框架结构的集成化设计环境,以适应当今 ASIC 的如下一些特点:数字与模拟电路并存;硬件与软件设计并存;产品上市速度要快。在这种集成化设计环境中,使用统一的数据管理系统与完善的通讯管理系统,由若干相关的设计小组共享数据库和知识库,并行地进行设计,而且在各种平台之间可以平滑过渡。

全球 EDA 厂商有近百家之多,大体可分为两类:一类是 EDA 专业软件公司,较著名的有 Mentor Graphics、Cadence Design Systems、Synopsys、Viewlogic Systems、Protel 等;另一类是半导体器件厂商,为了销售自己的产品而开发 EDA 工具,较著名的有 Altera、Xilinx、AMD、TI、Lattice 等。EDA 专业软件公司独立于半导体器件厂商,推出的 EDA 系统具有较好的标准化和兼容性,也比较注意追求技术上的先进性,适合于搞学术性基础研究的单位使用。而半导体厂商开发的 EDA 工具,能针对自己器件的工艺特点做出优化设计,提高资源利用率,降低功耗,改善性能,比较适合于进行产品开发的单位使用。在 EDA 技术发展策略上,EDA 专业软件公司面向应用,提供 IP 模块和相应的设计服务;而半导体厂商则采取三位一体的战略,在器件生产、设计服务和 IP 模块的提供上下工夫。

随着市场需求的增长,集成工艺水平的可行性以及计算机自动设计技术的不断提高,促使单片系统,或称系统集成芯片成为 IC 设计的发展方向,这一发展趋势表现在如下四个方面:

(1) 超大规模集成电路的集成度和工艺水平不断提高,深亚微米工艺,如 $0.18\ \mu\text{m}$ 、 $0.13\ \mu\text{m}$ 和 $90\ \text{nm}$ (2003 年) 已经走向成熟,在一个芯片上完成系统级的集成已成为可能。

(2) 市场对电子产品提出了更高的要求,如必须降低电子系统的成本,减小系统的体积等,从而对系统的集成度不断提出更高的要求。

(3) 高性能的 EDA 工具得到长足的发展,其自动化和智能化程度不断提高,为嵌入式系统设计提供了功能强大的开发环境。

(4) 计算机硬件平台性能大幅度提高, 为复杂的 SOC 设计提供了物理基础。

但现有的 HDL 语言只是提供行为级或功能级的描述, 尚无法完成对复杂的系统级的抽象描述。人们正尝试开发一种新的系统级设计语言来完成这一工作, 现在已开发出更趋于电路行为级的硬件描述语言, 如 System C、Superlog 及系统级混合仿真工具, 可以在同一个开发平台上完成高级语言(如 C / C++等), 与标准 HDL 语言(Verilog HDL、VHDL)或其他更低层次描述模块的混合仿真。虽然用户用高级语言编写的模块尚不能自动转化成 HDL 描述, 但作为一种针对特定应用领域的开发工具, 软件供应商已经为常用的功能模块提供了丰富的宏单元库支持, 可以方便地构建应用系统, 并通过仿真加以优化, 最后自动产生 HDL 代码, 进入下一阶段的 ASIC 实现。

此外, 随着系统开发对 EDA 技术的目标器件各种性能要求的提高, ASIC 和 FPGA 将更大程度地相互融合。这是因为虽然标准逻辑 ASIC 芯片尺寸小、功能强大、耗电省, 但设计复杂, 并且有批量生产要求。可编程逻辑器件开发费用低廉, 能在现场进行编程, 但体积大、功能有限, 而且功耗较大。因此, FPGA 和 ASIC 正在走到一起, 互相融合, 取长补短。由于一些 ASIC 制造商提供了具有可编程逻辑的标准单元, 可编程器件制造商重新对标准逻辑单元发生兴趣, 而有些公司采取两头并进的方法, 从而使市场开始发生变化, 在 FPGA 和 ASIC 之间正在诞生一种“杂交”产品, 以满足成本和上市速度的要求。例如, 将可编程逻辑器件嵌入标准单元。

尽管将标准单元核与可编程器件集成在一起并不意味着使 ASIC 更加便宜, 或使 FPGA 更加省电。但是, 可使设计人员将两者的优点结合在一起, 通过去掉 FPGA 的一些功能, 减少成本和开发时间并增加灵活性。

当然现在也在进行将 ASIC 嵌入可编程逻辑单元的工作。目前, 许多 PLD 公司开始为 ASIC 提供 FPGA 内核。PLD 厂商与 ASIC 制造商结盟, 为 SOC 设计提供嵌入式 FPGA 模块, 使未来的 ASIC 供应商有机会更快地进入市场, 利用嵌入式内核获得更长的市场生命期。

例如, 在实际应用中使用所谓可编程系统级集成电路(FPSLIC), 即将嵌入式 FPGA 内核与 RISC 微控制器组合在一起形成新的 IC, 广泛用于电信、网络、仪器仪表和汽车的低功耗应用系统中。

在新一代产品的电路板上, 尽管空间有限(几乎不能再增加器件), 在 ASIC 器件中仍留下了 FPGA 的空间。如果希望改变设计, 或由于开始的工作中没有条件做足够的验证测试, 稍后也可以根据要求对它编程, 有一定的再修改的自由度。ASIC 设计人员采用这种小的可编程逻辑内核用于修改设计问题, 很好地降低了设计风险。

ASIC 制造商增加可编程逻辑的另一个原因是, 考虑到设计产品的许多性能指标变化太快, 特别是通信协议, 故为已经完成设计并投入应用的 IC 留有多次可自由更改的功能是十分有价值的事, 这对通信领域中的芯片设计尤为重要。

现在, 传统的 ASIC 和 FPGA 之间的界限正变得模糊。系统级芯片不仅集成 RAM 和微处理器, 也集成 FPGA。整个 EDA 和 IC 设计工业都朝着这个方向发展, 这并非是 FPGA 与 ASIC 制造商竞争的产物, 而对于用户来说, 意味着有了更多的选择。

1.2 可编程逻辑器件的发展概况

集成电路的发展大大促进了 EDA 的发展, 先进的 EDA 已从传统的“自下而上”的设计方法改变为“自上而下”的设计方法。ASIC (Application Specific Integrated Circuit) 的设计与制造, 已不再完全由半导体厂商独立承担, 系统设计师在实验室里就可以设计出合适的 ASIC 芯片, 并且立即投入实际应用之中, 这都得益于可编程逻辑器件 (PLD, Programmable Logic Device) 的出现。现在应用最广泛的 PLD 主要是现场可编程门阵列 (FPGA, Field Programmable Gate Array) 和复杂可编程逻辑器件 (CPLD, Complex Programmable Logic Device)。

可编程逻辑器件(Programmable Logic Device)是 ASIC 的一个重要分支, 是厂家作为一种通用型器件生产的半定制电路, 用户可通过对器件编程实现所需要的逻辑功能。PLD 是用户可配置的逻辑器件, 它的成本比较低, 使用灵活, 设计周期短, 而且可靠性高, 风险小, 因而很快得到了普遍应用, 发展非常迅速。

PLD 从 20 世纪 70 年代发展到现在, 已经形成了许多类型的产品, 其结构、工艺、集成度、速度和性能都在不断改进和提高。PLD 又分为简单低密度 PLD 和复杂高密度 PLD。最早的 PLD 是 1970 年制成的可编程只读存储器 (PROM, Programmable Read-Only Memory), 它由固定的与阵列和可编程的或阵列组成。PROM 采用熔丝工艺编程, 只能写一次, 不能擦除和重写。随着技术的发展和应用要求的提高, 此后又出现了 UVEPROM(紫外线可擦除只读存储器)和 EEPROM(电可擦除只读存储器), 由于它们价格低, 易于编程, 速度低, 适合于存储函数和数据表格, 因此主要用做存储器。典型的 EPROM 有 2716、2732 等。

可编程逻辑阵列 (PLA, Programmable Logic Array)于 20 世纪 70 年代中期出现, 它是由可编程的与阵列和可编程的或阵列组成的, 但由于器件的资源利用率低, 价格较贵, 编程复杂, 支持 PLA 的开发软件有一定难度, 因而没有得到广泛应用。

可编程阵列逻辑 (PAL, Programmable Array Logic)器件是 1977 年美国 MMI 公司(单片存储器公司)率先推出的, 它由可编程的与阵列和固定的或阵列组成, 采用熔丝编程方式, 双极性工艺制造, 器件的工作速度很高。由于它的输出结构种类很多, 设计很灵活, 因而成为第一个得到普遍应用的可编程逻辑器件, 如 PAL16L8。

通用阵列逻辑(GAL, Generic Array Logic)器件是 1985 年 Lattice 公司最先发明的可电擦写、可重复编程、可设置加密位的 PLD。GAL 在 PAL 基础上, 采用了输出逻辑宏单元形式 E²CMOS 工艺结构。具有代表性的 GAL 芯片有 GAL16V8 和 GAL20V8, 这两种 GAL 几乎能够仿真所有类型的 PAL 器件。在实际应用中, GAL 器件对 PAL 器件仿真具有 100% 的兼容性, 所以 GAL 几乎完全代替了 PAL 器件, 并可以取代大部分 SSI、MSI 数字集成电路, 如标准的 54/74 系列器件, 因而获得了广泛的应用。

PAL 和 GAL 都属于简单 PLD, 结构简单, 设计灵活, 对开发软件的要求低, 但规模小, 难以实现复杂的逻辑功能。随着技术的发展, 简单 PLD 在集成密度和性能方面的局限性也暴露出来, 其寄存器、I/O 引脚和时钟资源的数目有限, 没有内部互连, 因此包括

EPLD、CPLD 和 FPGA 在内的复杂 PLD 迅速发展起来,并向着高密度、高速度、低功耗以及结构体系更灵活、适用范围更宽广的方向发展。

可擦除可编程逻辑器件 (EPLD, Erasable PLD)是 20 世纪 80 年代中期 Altera 公司推出的基于 UVEPROM 和 CMOS 技术的 PLD,后来发展到采用 E²CMOS 工艺制作的 PLD。EPLD 基本逻辑单元是宏单元。宏单元由可编程的与或阵列、可编程寄存器和可编程 I/O 三部分组成。从某种意义上讲, EPLD 是改进的 GAL,它在 GAL 基础上大量增加输出宏单元的数目,提供更大的与阵列,灵活性较 GAL 有较大改善,集成密度大幅度提高,内部连线相对固定,延时小,有利于器件在高频率下工作,但内部互连能力十分弱。世界著名的半导体器件公司如 Altera、Xilinx、AMD 和 Lattice 均有 EPLD 产品,但结构差异较大。

复杂可编程逻辑器件 (CPLD, Complex PLD)是 20 世纪 80 年代末 Lattice 公司提出的在线可编程(isp, In System Programmability)技术以后,于 20 世纪 90 年代初出现的。CPLD 是在 EPLD 的基础上发展起来的,采用 E²CMOS 工艺制作,与 EPLD 相比,增加了内部连线,对逻辑宏单元和 I/O 单元也有重大的改进。CPLD 至少包含三种结构:可编程逻辑宏单元、可编程 I/O 单元和可编程内部连线。部分 CPLD 器件内部还集成了 RAM、FIFO 或双口 RAM 等存储器,以适应 DSP 应用设计的要求。其典型器件有 Altera 的 MAX7000 系列, Xilinx 的 7000 和 9500 系列, Lattice 的 PLSI/ispLSI 系列和 AMD 的 MACH 系列。

现场可编程门阵列 (FPGA, Field Programmable Gate Array)器件是 Xilinx 公司在 1985 年首家推出的,它是一种新型的高密度 PLD,采用 CMOS-SRAM 工艺制作。FPGA 的结构与门阵列 PLD 不同,其内部由许多独立的可编程逻辑模块(CLB)组成,逻辑块之间可以灵活地相互连接。FPGA 的结构一般分为三部分:可编程逻辑块、可编程 I/O 模块和可编程内部连线。CLB 的功能很强,不仅能够实现逻辑函数,还可以配置成 RAM 等复杂的形式。配置数据存放在片内的 SRAM 或熔丝图上,基于 SRAM 的 FPGA 器件工作前需要从芯片外部加载配置数据。配置数据可以存储在片外的 EPROM 或计算机上,设计人员可以控制加载过程,在现场修改器件的逻辑功能,即所谓现场可编程。FPGA 出现后受到电子设计工程师们的普遍欢迎,发展十分迅速。Xilinx、Altera、Actel 等公司都能提供高性能的 FPGA 芯片。

高密度 PLD 在生产工艺、器件编程、测试技术等方面都有了飞速发展。目前 PLD 的集成度非常高,如 Altera 公司的 APEX II 系列器件,采用 0.15 μm 工艺,各层都是铜金属布线,其中 EP2A90 的密度可达 400 万典型门,可为用户提供 1 140 个 I/O 引脚,1 GB/s 的数据速率。世界各著名半导体器件公司,如 Altera、Xilinx、Lattice、Actel 和 AMD 公司均可提供不同类型的 CPLD 和 FPGA 产品。众多公司的竞争促进了可编程集成电路技术的提高,使其性能不断改善,产品日益丰富,价格逐步下降。可以预计,可编程逻辑器件将在结构、密度、功能、速度、性能等方面得到进一步发展,结合 EDA 技术,PLD 将在现代电子系统设计中得到非常广泛的应用。

20 世纪末出现了 SOPC (片上可编程系统)器件, SOPC 是现代电子技术和电子系统设计的汇聚点和最新发展方向,它将普通 EDA 技术、计算机系统、嵌入式系统、工业自动化控制系统、DSP 及无线电等融为一体,涵盖了嵌入式系统设计技术的全部内容。SOPC 结合了 SOC、PLD 和 FPGA 各自的优点,集成了硬核或软核 CPU、DSP、存储器、外围

I/O 及可编程逻辑, 用户可以利用 SOPC 平台自行设计各种高速高性能的 DSP 处理器或特定功能的 CPU 处理器, 从而使电子系统设计进入了一个全新的模式。在应用的灵活性和价格上 SOPC 有极大的优势, 被称为“半导体产业的未来”。Xilinx 公司和 Altera 公司的新一代 FPGA 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 SOPC 提供了强大的硬件支持。

1.3 硬件描述语言 VHDL

1.3.1 概述

硬件描述语言 VHDL 是 EDA 技术的重要组成部分, 是电子设计主流硬件的描述语言, 本书将重点介绍它的编程方法和使用技术。

VHDL 的英文全名是 VHSIC (Very High Speed Integrated Circuit) Hardware Description Language, 于 1983 年由美国国防部 (DOD) 发起创建, 由 IEEE (the Institute of Electrical and Electronics Engineers) 进一步发展并在 1987 年作为“IEEE 标准 1076”发布。从此, VHDL 成为硬件描述语言的业界标准之一。自 IEEE 公布了 VHDL 的标准版本 (IEEE Std 1076) 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支持 VHDL。此后, VHDL 在电子设计领域得到了广泛应用, 并逐步取代了原有的非标准硬件描述语言。

VHDL 作为一种规范语言和建模语言, 随着 VHDL 的标准化, 出现了一些支持该语言的行为仿真器。由于创建 VHDL 的最初目标是用于标准文档的建立和电路功能的模拟, 其基本想法是在高层次上描述系统和元件的行为。但到了 20 世纪 90 年代初, 人们发现, VHDL 不仅可以作为系统模拟的建模工具, 而且可以作为电路系统的设计工具。可以利用软件工具将 VHDL 源码自动地转化为文本方式表达的基本逻辑元件连接图, 即网表文件。这种方法显然对于电路自动设计是一个极大的推进。很快, 电子设计领域出现了第一个软件设计工具, 即 VHDL 逻辑综合器, 它把标准 VHDL 的部分语句描述转化为具体电路实现的网表文件。

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了 VHDL 的内容, 公布了新版本的 VHDL, 即 IEEE 标准的 1076-1993 版本。现在, VHDL 和 Verilog 作为 IEEE 的工业标准硬件描述语言, 得到了众多 EDA 公司的支持, 在电子工程领域, 已成为事实上的通用硬件描述语言。

VHDL 语言具有很强的电路描述和建模能力, 能从多个层次对数字系统进行建模和描述, 从而大大简化了硬件设计任务, 提高了设计效率和可靠性。

VHDL 具有与具体硬件电路无关和与设计平台无关的特性, 并且具有良好的电路行为描述和系统描述的能力, 并在语言易读性和层次化结构化设计方面, 表现出强大的生命力和应用潜力。因此, VHDL 在支持各种模式的设计方法、自顶向下与自底向上或混合方法方面; 在面对当今许多电子产品生命周期的缩短, 需要多次重新设计以融入最新技术、改变工艺等方面都表现了良好的适应性。用 VHDL 进行电子系统设计的一个很大的优点是设

计者可以专心致力于其功能的实现，而不需要对不影响功能的与工艺有关的因素花费过多的时间和精力。

1.3.2 VHDL 综合

综合(Synthesis)，就其字面含义应为把抽象的实体结合成单个或统一的实体。因此，综合就是把某些东西结合到一起，把设计抽象层次中的一种表示转化成另一种表示的过程。对于电子设计领域的综合概念可以表示为将用行为和功能层次表达的电子系统转换为低层次的便于具体实现的模块组合装配而成的过程。

事实上，设计过程中的每一步都可称为一个综合环节。设计过程通常从高层次的行为描述开始，以最低层的结构描述结束，每个综合步骤都是上一层次的转换。

(1)从自然语言转换到 VHDL 语言算法表示，即自然语言综合。

(2)从算法表示转换到寄存器传输级(RTL, Register Transport Level)，从行为域到结构域的综合，即行为综合。

(3)RTL 级表示转换到逻辑门(包括触发器)表示，即逻辑综合。

(4)从逻辑门表示转换到版图表示(ASIC 设计)，或转换到 FPGA 的配置网表文件，即版图综合或结构综合。有了版图信息就可以把芯片生产出来，有了对应的配置文件就可以使对应的 FPGA 变成具有专门功能的电路器件。

显然，综合器就是能够自动将一种设计表示形式向另一种设计表示形式转换的计算机程序，或协助进行手工转换的程序。它可以将高层次的表示转化为低层次的表示，可以从行为域转化为结构域，可以将高一级抽象的电路表示(如算法级)转化为低一级的表示(如门级)，并可以用某种特定的技术(如 CMOS)来实现。

对设计者而言，在高抽象层次进行系统设计，再利用综合工具将设计转化为低层次的表示，与直接在低抽象层次来设计系统的情况相比较，就类似于一个程序员用高级语言编程并用编译器将程序编译成机器代码和直接用机器代码进行编程时的情况。前一种情况在于设计者可以将精力主要集中于系统级问题上，而不必关心低级结构设计的细节问题。由于不必关心低级层次的设计所面临的细节问题，在高级抽象层次上进行设计和编程将花费较少的时间和精力，并且减少错误。

另一方面，尽管从表面上看，VHDL 等硬件描述语言综合器和软件程序编译器都不过是一种“翻译器”，它们都能将高层次的设计表达转化为低层次的表达，但它们却具有许多本质的区别，如图 1.1 所示。

编译器将软件程序翻译成基于某种特定 CPU 的机器代码，这种代码仅限于这种 CPU 而不能移植，并且机器代码不代表硬件结构，更不能改变 CPU 的硬件结构，只能被动地为其特定的硬件电路结构所利用。如果脱离了已有的硬件环境(CPU)，机器代码将失去意义。此外，编译器作为一种软件的运行，除了某种单一目标器件，即 CPU 的硬件结构外，不需要任何与硬件相关的器件库和工艺库参与编译。因而，编译器的工作单纯得多，编译过程基本属于一种一一对应、机械转换式的翻译行为。

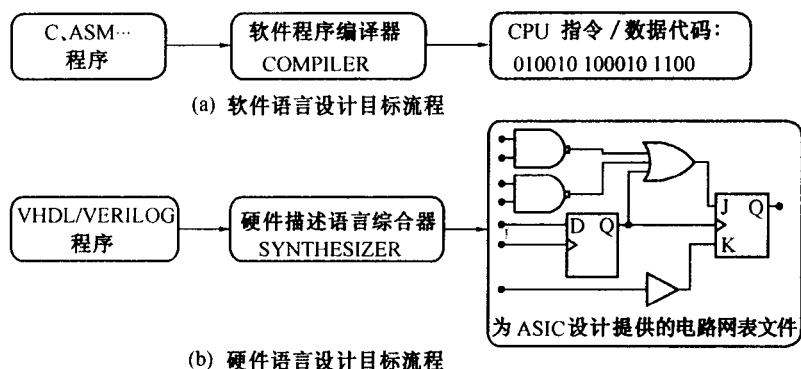


图 1.1 编译器和综合器功能比较

综合器则不同，同样是软件代码(如 VHDL 程序)，综合器转化的目标是底层的电路结构网表文件，这种满足原设计程序功能描述的电路结构不依赖于任何特定的硬件环境，因此可以独立地存在，并能轻易地被移植到任何通用的硬件环境中，如 ASIC、FPGA 等。换言之，电路网表代表了特定的硬件结构，因此具备了随时改变硬件结构的依据。综合的结果具有相对独立性。另一方面，综合器在将硬件描述语言表达的电路功能转化成具体的电路结构网表的过程中，具有明显的能动性和创造性，它不是机械的一一对应式的翻译，而是根据设计库、工艺库以及预先设置的各类约束条件，选择最优的方式完成电路结构的形成。这就是说，对于相同的 VHDL 表述，综合器可以用不同的电路结构实现相同的功能。

与编译器相比，综合器具有更复杂的工作环境。综合器在接受 VHDL 程序并准备对其综合前，必须获得最终实现设计电路硬件特征相关的工艺库的信息，以及获得优化综合的诸多约束条件。一般地，约束条件可以分为三种，即设计规则、时间约束和面积约束。通常，时间约束的优先级高于面积约束。设计优化要求，当综合器把 VHDL 源码翻译成通用原理图时，将识别状态机、加法器、乘法器、多路选择器、寄存器等。这些运算功能是根据 VHDL 源码中的符号来完成的，如加、减、乘、除。每种运算都可用多种方法实现，如加法可实现的方案有多种，有的面积小、速度慢；有的速度快、面积大。VHDL 行为描述强调的是电路的行为和功能，而不是电路如何实现。选择电路的实现方案正是综合器的任务。综合器将选择一种能充分满足各项约束条件且成本最低的实现方案。

对于相同的 VHDL 源代码，不同的 VHDL 综合器可能综合出在结构和功能上并不完全相同的电路系统。对此，设计者应给予充分的注意，对于不同的综合结果，不对综合器的特性贸然做出评价，同时在设计过程中，必须尽可能全面地了解所使用的综合工具的基本特性。

1.4 EDA 与传统电子设计方法

1.4.1 传统的电子设计方法

传统的电子设计技术通常是自底向上的，即首先确定构成系统的最底层的电路模块或

元件的结构和功能，然后根据主系统的功能要求，将它们组合成更大的功能块，使它们的结构和功能满足上层系统的要求。以此流程，逐步向上递推，直至完成整个目标系统的设计。

例如，对于一般电子系统的设计，使用自底向上的设计方法，必须首先决定使用的器件的类别和规格，如 74 系列的器件、某种 RAM 和 ROM、某类 CPU 或单片机以及某些专用功能芯片等；然后是构成多个功能模块，如数据采集控制模块、信号处理模块、数据交换和接口模块等，直至最后利用它们完成整个系统的设计。

以一个六进制计数器设计为例予以说明。

第一步，选择逻辑元、器件。由数字电路的基本知识可知，可以用与非门、或非门、D 触发器、JK 触发器等基本逻辑元、器件来构成一个计数器。设计者根据电路尽可能简单、价格合理、购买和使用方便及各自的习惯来选择六进制计数器的逻辑元、器件。本例中我们选择 JK 触发器和 D 触发器作为六进制计数器的主要逻辑元、器件。

第二步，设计电路。假设六进制计数器采用约翰逊计数器，3 个触发器应该产生 8 种状态，现在只使用 6 种状态，将其中的 010 和 101 两种状态禁止掉，如图 1.2、图 1.3、表 1.1、表 1.2 所示。

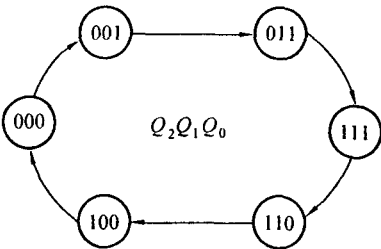


图 1.2 六进制计数器状态图

表 1.1 六进制计数器状态变化表

触发器 状态 计数脉冲	Q_2		Q_1		Q_0	
	Q_{n-1}	Q_n	Q_{n-1}	Q_n	Q_{n-1}	Q_n
1	0	0	0	0	0	1
2	0	0	0	1	1	1
3	0	1	1	1	1	1
4	1	1	1	1	1	0
5	1	1	1	0	0	0
6	1	0	0	0	0	0