

基于VHDL的 CPLD/FPGA开发与应用

张丕状 李兆光 编著



國防工業出版社

National Defense Industry Press

基于 VHDL 的 CPLD/FPGA 开发与应用

张丕状 李兆光 编著

國防工業出版社

· 北京 ·

内 容 简 介

本书以 Xilinx 公司和 Alter 公司的 CPLD/FPGA 开发环境为基础,对 VHDL 硬件描述语言、CPLD/FPGA 开发应用及相关知识做了系统和完整的介绍,使读者掌握 CPLD/FPGA 在电路设计中的基本方法及实用技术。

全书共分 8 章。第 1 章介绍 EDA 基本概念和 CPLD/FPGA 开发的基本设计方法;第 2 章介绍几种 PLD 器件的典型结构和原理;第 3 章、第 4 章介绍 VHDL 语言的基本概念、框架和语法知识;第 5 章介绍典型的数字电路的 VHDL 参考设计;第 6 章介绍可编程器件的开发环境;第 7 章介绍片上可编程系统 SOPC 的设计入门;第 8 章以数据采集系统中控制器的设计为例,详述了 VHDL 语言在实际工程设计中的应用。

本书可作为高等学校电气信息类、电子信息类及其他相近专业本科生和研究生教材,也可作为有关工程技术人员的参考书。同时,也希望对于从事数据采集技术研究的技术人员能有一定的帮助。

图书在版编目(CIP)数据

基于 VHDL 的 CPLD/FPGA 开发与应用 / 张丕状,李兆光编
著. —北京:国防工业出版社,2009.6
ISBN 978-7-118-06402-5

I. 基... II. ①张...②李... III. ①可编程序逻辑器件 - 高等学校 - 教材②硬件描述语言,VHDL - 程序设计 - 高等学校 - 教材 IV. TP332.1 TP312

中国版本图书馆 CIP 数据核字(2009)第 116765 号

※

国防工业出版社出版发行

(北京市海淀区紫竹院南路 23 号 邮政编码 100048)

北京奥鑫印刷厂印刷

新华书店经售

*

开本 787×1092 1/16 印张 16 字数 366 千字

2009 年 6 月第 1 版第 1 次印刷 印数 1—4000 册 定价 30.00 元

(本书如有印装错误,我社负责调换)

国防书店:(010)68428422
发行传真:(010)68411535

发行邮购:(010)68414474
发行业务:(010)68472764

前 言

随着可编程逻辑电路和 EDA 技术的发展,现代电子设计技术进入一个全新的阶段。可编程逻辑器件已经是电子设计的一个重要技术基础,在逻辑电路设计及嵌入式系统设计方面,以 CPLD/FPGA 为代表的可编程逻辑器件已经逐步取代了传统的标准器件。传统的基于原理图设计方法已经不再适用于大型逻辑电路的设计,而硬件描述语言在这方面显示出了强大的优势。

VHDL 硬件描述语言,以其强大的行为描述能力、灵活的语句表达风格和多层次的仿真测试手段,受到业界的普遍认同和广泛的接受。本书以实用性和可操作性为基点,以掌握基于 VHDL 的 CPLD/FPGA 开发能力为目标,通过细致的概念讲解、大量的设计实例,系统地讲述了 VHDL 硬件描述语言、CPLD/FPGA 开发技术和逻辑电路的模块化设计方法。

目前,电子系统的设计已进入可编程片上系统 SOPC 时代。SOPC 集成了硬核或软核 CPU、锁相环、存储器、输入/输出接口及可编程逻辑,可以灵活、高效地解决 SOC 方案,大大缩短设计周期。本书第 7 章选用 Altera 公司的 Quartus II 7.2 作为 SOPC 的设计工具,简要介绍 SOPC 设计的流程,通过实例让读者了解 SOPC 的设计方法。

在内容的组织和编写风格上,力求做到结合新颖而详尽的设计实例、深入浅出、信息量大、注重实践和设计技巧。

为了读者对本书有更好的认识,现对本书的组织思想介绍如下。

主要介绍 VHDL 及其在基于 CPLD/FPGA 的数字电路系统开发中的应用,也简单介绍一些 EDA 基本知识和开发工具。

第 1 章,简单介绍 EDA 的基本概念和 CPLD/FPGA 开发流程。

第 2 章,介绍 CPLD 和 FPGA 的几种典型结构。该内容对于 VHDL 学习本身不是必要的,但对于电路的设计,尤其在实际工作中的芯片选型,这些知识又是必须的。该内容对电路设计中性能、速度和资源利用方面也是非常有意义的。

第 3 章,VHDL 的主体部分,也是理解 VHDL 的关键知识,重点是概念的理解。书中不少内容是作者自己个人的理解和总结,这些内容是学习 VHDL 的关键知识。

第 4 章,VHDL 的另一个主体内容,本章重点介绍一些典型语句,概念性的东西相对第 3 章要少一些。在内容组织时,重点放在了语句的理解和用法上,在此基础上兼顾语法知识。

第 5 章,常见电路功能模块的 VHDL 描述。但对于一名从事电路设计的人员来说,本章的所有例子都应该像围棋学习中的“定势”一样熟记在心。虽然数字逻辑电路课程中有部分相关内容(状态转换图等),但有限状态机的概念、描述对大多数读者而言大都

是新的内容,考虑到该内容在实际电路设计中的重要性,建议读者重点关注并掌握。

第6章,介绍两个现在市场上常用的开发平台软件。由于任何一个开发平台软件的功能都非常强大,内容也多,所以在内容组织时,只以一个实例开发的形式介绍了从工程创建到仿真的整个流程。在本章中,读者在掌握实验流程的基础上,重点学习并熟练掌握一种仿真方法。

第7章,介绍嵌入式系统开发流程(SOC)。FPGA技术是嵌入式系统的一个发展方向,考虑到知识的完整性和系统性,在本书中增加了这部分了解性质的内容。进一步的学习需要参考更详细的相关文献。

第8章,VHDL的综合应用示例。作为VHDL的应用示例,过于零碎的或小规模的电路设计不足以体现VHDL的特点,甚至不能体现语句的恰当用法;而过于庞大的设计又需要读者具备太多的专业知识(这样的要求对于本科生和低年级研究生来说基本是不切实际的),造成对整体设计思想上的理解困难。本书结合作者多年电路设计经验,选用大多数读者都具有一定基础知识的数据采集领域,经过归纳、整理和系统化,作为VHDL的综合应用示例。既可反映出VHDL中大部分概念、语句的用法,也能体现VHDL灵活的电路描述风格,同时也增加了数据采集的专业知识。

参加本书编写工作的有中北大学张丕状(第3章、第8章)、李兆光(第4章~第7章),太原工业学院杨翠娥、索静、郭彩萍(三位老师合编第1章、第2章)。

由于作者水平有限,书中错误之处在所难免,恳请广大读者提出批评和改进意见。读者的反馈信息可通过电子邮件发送至:WYF20040702@163.COM。

编者

2009年5月于太原

目 录

第 1 章 概述	1
1.1 EDA 技术的发展概况	1
1.2 EDA 技术的基本内容	2
1.3 可编程逻辑器件	3
1.4 软件开发工具	4
1.5 硬件描述语言概述	5
1.6 基于 EDA 软件的 CPLD/FPGA 设计流程	7
1.7 IP 核	8
第 2 章 可编程逻辑器件	10
2.1 概述	10
2.2 简单可编程器件	10
2.2.1 PLD 的基本结构	10
2.2.2 PLD 的表示方法	11
2.2.3 PROM	13
2.2.4 PLA	14
2.2.5 PAL	14
2.2.6 GAL	15
2.3 CPLD	19
2.3.1 CPLD 的基本结构与工作原理	19
2.3.2 常用的 CPLD 器件	22
2.4 FPGA	24
2.4.1 查找表	24
2.4.2 FPGA 的基本结构和工作原理	25
2.4.3 常用的 FPGA 器件	31
2.5 CPLD 与 FPGA 的比较	34
第 3 章 硬件描述语言 VHDL 的基本框架介绍	36
3.1 概述	36
3.1.1 硬件描述语言的简介	36
3.1.2 VHDL 与高级语言的联系与区别	36
3.2 VHDL 的基本结构	39
3.2.1 实体说明	41
3.2.2 结构体(构造体)	43

3.2.3 库、程序包及配置	45
3.3 VHDL 语言要素	48
3.3.1 数据对象	48
3.3.2 数据类型	52
3.3.3 基本运算符	61
3.3.4 属性	65
3.4 VHDL 语言的结构体描述方式	67
3.4.1 行为描述方式	67
3.4.2 数据流描述方式	70
3.4.3 结构化描述方式	71
第 4 章 VHDL 典型语句	73
4.1 VHDL 顺序语句	73
4.1.1 赋值语句	73
4.1.2 条件控制语句	76
4.1.3 循环语句	81
4.1.4 其他顺序语句	85
4.2 VHDL 并行语句	87
4.2.1 进程语句	87
4.2.2 并行信号赋值语句	91
4.2.3 元件说明与元件例化语句	94
4.2.4 生成语句	96
4.2.5 块语句	98
4.3 子程序	101
4.3.1 函数	101
4.3.2 过程	106
4.4 程序包、配置	110
4.4.1 程序包	110
4.4.2 配置	111
第 5 章 常用数字逻辑电路与 VHDL 描述方法	117
5.1 组合逻辑电路的设计	117
5.1.1 加法器	117
5.1.2 编码器、译码器	118
5.1.3 多路选择器、多路分配器	121
5.1.4 三态门及数据缓冲器	123
5.2 时序逻辑电路	125
5.2.1 时钟信号与进程	125
5.2.2 触发器的描述与置位、复位方式	125
5.2.3 寄存器	127
5.2.4 计数器	130

5.3 有限状态机的设计	133
5.3.1 状态转换图与有限状态机	133
5.3.2 Moore 型的有限状态机设计	136
5.3.3 Mealy 型的有限状态机设计	138
5.4 存储器的设计	140
5.4.1 只读存储器	140
5.4.2 随机存储器	141
5.4.3 FIFO(先进先出堆栈)	141
5.5 仿真方法	145
5.5.1 测试平台的概念	145
5.5.2 测试平台的编写	145
5.6 毛刺及其消除	148
5.6.1 毛刺及其产生的原因	148
5.6.2 毛刺的消除	149
5.7 CAN 总线控制器	150
5.7.1 CAN 总线协议	151
5.7.2 CAN 通信控制器的基本框架	152
5.7.3 CAN 通信控制器的具体实现	154
第 6 章 可编程器件的开发环境	158
6.1 Quartus II 使用入门	158
6.1.1 Quartus II 软件的用户界面	158
6.1.2 Quartus II 软件的开发流程	159
6.2 Xilinx ISE 使用入门	171
6.2.1 Xilinx ISE 软件的用户界面	171
6.2.2 Xilinx ISE 软件的开发流程	172
第 7 章 SOPC 设计入门	178
7.1 SOPC 的基本概念	178
7.1.1 SOPC 及其技术	178
7.1.2 Nios II 软核 SOPC 系统及组件	179
7.2 SOPC 的硬件开发环境及硬件开发	184
7.2.1 SOPC Builder 简介	184
7.2.2 SOPC 的硬件开发	185
7.3 SOPC 的软件开发环境及软件开发	192
7.3.1 Nios II IDE 软件开发环境	192
7.3.2 Nios II 外设及其编程实例	196
7.4 自定义外设的开发	204
7.4.1 自定义外设的 VHDL 描述	204
7.4.2 封装为 SOPC Builder 组件	208
7.4.3 在 Nios 系统中添加组件	210

7.4.4 自定义外设的应用	211
第 8 章 数据采集中的控制器设计	212
8.1 数据采集的基础知识	212
8.1.1 采集速度与电路结构的关系	212
8.1.2 A/D 分辨率和数据输出格式	213
8.1.3 信号采样周期	214
8.1.4 信号处理方式	214
8.1.5 多通道数据采集	215
8.1.6 触发方式	216
8.1.7 负延迟触发	217
8.1.8 常见的数据采集系统的组织结构	218
8.2 典型数据采集系统中的控制电路设计	219
8.2.1 低速多路数据采集系统中控制电路设计	219
8.2.2 带 FIFO 缓冲存储的多通道数据采集控制电路设计	225
8.2.3 流式信号实时处理算法的寄存器组与 A/D 转换控制电路设计	240
8.2.4 存储式数据采集系统中负延迟触发的控制电路设计	244
参考文献	247

第1章 概述

1.1 EDA 技术的发展概况

EDA 是电子设计自动化 (Electronic Design Automation) 的缩写。EDA 工具是以计算机的硬件和软件为基本工作平台, 集数据库、图形学、图论与拓扑逻辑、计算数学、优化理论等多学科最新成果研制的计算机辅助设计通用软件包。利用 EDA 工具可以代替设计者完成电子系统设计中的大部分工作, 主要包括三方面的辅助设计工作, 即集成电路 (IC) 设计、电子电路设计以及 PCB 设计等。利用 EDA 技术进行电子系统的设计, 具有以下几个特点: 用软件的方式设计硬件; 用软件方式设计的系统到硬件系统的转换是由有关的开发软件自动完成的; 设计过程中可用软件进行各种仿真; 系统可现场编程, 在线升级; 整个系统可集成在一个芯片上, 体积小、功耗低、可靠性高。

EDA 技术伴随着计算机、集成电路和电子系统设计的发展, 经历了计算机辅助设计 (Computer Assist Design, CAD)、计算机辅助工程设计 (Computer Assist Engineering Design, CAED) 和电子设计自动化 (Electronic Design Automation, EDA) 三个发展阶段。

1. 计算机辅助设计阶段

20 世纪 60 年代中期至 20 世纪 80 年代初期为 CAD 发展的阶段。这个阶段人们研制了一些单独的软件工具, 主要有印制电路板 (Printed Circuit Board, PCB) 布线设计、电路模拟、逻辑模拟及版图的绘制等, 从而可以利用计算机将设计人员从大量烦琐、重复的计算和绘图工作中解脱出来。

2. 计算机辅助工程设计阶段

20 世纪 80 年代初期至 20 世纪 90 年代初期为 CAED 阶段, 这个阶段在集成电路与电子系统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均已齐全。

3. 电子系统设计自动化阶段

20 世纪 90 年代以后, 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, 它不仅极大地提高了系统的设计效率, 而且使设计者摆脱了大量的辅助性工作, 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) 高层综合的理论与方法取得了较大的进展, 将 EDA 设计层次由 RTL 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,使设计者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。

(2) 采用硬件描述语言来描述 10 万门以上的设计,并形成了 VHDL 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,多应用于 FPGA/CPLD 的设计中。

(3) 可测试综合设计。随着 ASIC 的规模与复杂性的增加,测试难度与费用急剧上升,由此产生了将可测试电路结构制作在 ASIC 芯片上的想法,于是开发了扫描插入、BLST(内建自测试)、边界扫描等可测试设计(DFT)工具,并已集成到 EDA 系统中。

(4) 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、形式验证和可测试设计这几个方面。

(5) 建立并行设计工程(Concurrent Engineering, CE)框架结构的集成化设计环境,以适应当今大规模专用集成电路(Application Specific Integrated Circuit, ASIC)的一些特点: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信管理系统,由若干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。

1.2 EDA 技术的基本内容

从 EDA 技术的应用范围而言,EDA 技术包括电子电路设计的各个领域。从低频电路到高频电路,从线性电路到非线性电路,从模拟电路到数字电路,从分立电路到集成电路的全部设计过程,涉及电子工程师进行产品开发的全过程,以及电子产品的全过程中期望由计算机提供的各种辅助设计工作。

利用 EDA 技术进行电子系统设计主要有四个方面:PCB、全定制或半定制 ASIC、FPGA/CPLD 开发与应用及混合电路设计。印制电路板设计是 EDA 技术最初的实现目标,在电子系统实现过程中,印制电路板设计、装配和测试需要的工作量是很大的,也是一个很具有工艺性、技巧性的工作。利用 EDA 工具来进行印制电路板的布局布线设计和验证分析,是早期 EDA 技术最基本的应用。集成电路设计是 EDA 技术推广发展的重要源泉和动力。随着超大规模集成电路的出现,传统的手工设计方法遇到的困难越来越多,为了保证设计的正确性和可靠性,必须采用先进的 EDA 技术来进行集成电路的逻辑设计、电路设计和版图设计。可编程逻辑器件的发展使用户可自行构造特定逻辑功能的集成电路,从而促进 EDA 技术的推广应用。FPGA/CPLD 的开发与应用是 EDA 技术将电子系统设计与硬件实现进行有机融合的一个重要体现。

本书重点介绍 FPGA/CPLD 的开发与应用,读者应从以下三个方面了解 FPGA/CPLD 的开发与应用:可编程逻辑器件;软件开发工具;硬件描述语言。可编程逻辑器件是利用 EDA 技术进行电子系统设计的载体;硬件描述语言是利用 EDA 技术进行电子系统设计的主要表达手段;软件开发工具是利用 EDA 技术进行电子系统设计的智能化的自动

化设计工具。

1.3 可编程逻辑器件

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“自底而上”的设计方法改变为“自顶而下”的设计方法。ASIC 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就可以设计出合适的 ASIC 芯片,并且立即投入实际应用中,这都得益于可编程逻辑器件(Programmable Logic Device, PLD)的出现。现在应用最广泛的 PLD 主要是现场可编程门阵列(Field Programmable Gate Array, FPGA)和复杂可编程逻辑器件(Complex Programmable Logic Device, CPLD)。

PLD 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。同时,PLD 也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 年研制成的可编程只读存储器 PROM,它是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 EPROM 和 E²PROM,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。

可编程逻辑阵列(Programmable Logic Array, PLA)器件于 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(Programmable Array Logic, PAL)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(Generic Array Logic, GAL)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 的基础上,采用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 类型。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,因而包括 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。

复杂可编程逻辑(Complex PLD, CPLD)器件是 20 世纪 90 年代提出的,它至少包

含三种结构,即可编程逻辑宏单元、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。

现场可编程门阵列(Field Programmable Gate Array, FPGA)器件是 Xinlinx 公司在 1985 年推出的,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,即可编程逻辑块、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。

20 世纪末出现了片上可编程系统(SOPC),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 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”。

1.4 软件开发工具

目前,比较流行的、基于可编程器件的 EDA 软件工具主要有 Altera 公司的 MAX+Plus 和 Quartus II、Lattice 公司的 ispDesignEXPERT 和 ispLEVER、Xinlinx 公司的 Foundation 和 ISE 等。

1. Altera 公司的 EDA 集成开发工具

由 Altera 公司开发的 EDA 集成开发工具,现已经历了四代产品,分别为 A+Plus、MAX+Plus、MAX+Plus II、Quartus II。

MAX+Plus II 提供了与结构无关的设计环境,是一个完全集成化、易学易用的 EDA 开发软件,它支持原理图、VHDL 和 Verilog HDL 语言文本文件,以及以波形与 EDIF 等格式的文件作为设计输入,并支持这些文件的混合设计,配有编辑、编译、综合及芯片编程下载等功能。

Quartus II 支持百万门级的设计,适合大规模 FPGA 的开发,可完成从设计输入、综合适配、仿真到编程下载整个设计过程的设计,并提供了更优化的综合和适配的功能,改善了对第三方仿真和时序分析工具的支持,可以直接调用 Synplify Pro (Synplicity 公司)、Leonardo Spectrum (Mentor 公司),以及 ModelSim (Mentor 公司的 Model Technology 子公司)等第三方 EDA 工具来完成设计任务的综合和仿真。另外,Quartus II 包含了 SOPC Bulider,可自动添加、参数化和链接 IP 核,包括嵌入式处理器、协处理器、外设和用户定义等。它不仅可以与 MATLAB 和 DSP Builder 结合进行基于 FPGA 的 DSP 系统的开发,而且还可以实现可编程片上系统 SOPC 的开发。

2. Lattice 公司的 EDA 集成开发工具

Lattice 公司推出的 EDA 集成开发工具主要有 ispSynario、ispExpert、ispDesign EXPERT 和 ispLEVER。

ispDesign EXPERT 是 Lattice 公司的第四代产品，它是一套完整的 EDA 开发软件，可以用原理图或硬件描述语言进行设计输入，也可采用混合方式输入；软件还配有编辑、编译、综合、适配、仿真和编程下载等功能；具有界面良好、操作方便、功能强大、与第三方 EDA 工具能有良好的兼容等优点。

ispLEVER 是 Lattice 公司最新推出的集成开发工具，除了支持原有的器件外，还支持 ispMACH、ispXPLD、ispXPGA、ispGDX2 等最新的器件。该软件同时集成了许多第三方专业工具，如综合工具 Synplify Pro、Leonardo Spectrum 以及 ModelSim 等。

3. Xilinx 公司的 EDA 集成开发工具

Xilinx 公司推出的 EDA 集成开发工具主要有 Foundation 和 ISE。

Foundation Series 集成开发工具采用自动化的、完整的集成设计环境。所有的设计输入、实现和仿真都是在项目管理器中完成的；能够提供多层次原理图设计流程、HDL 设计流程和状态图设计流程；并能进行真正的混合语言综合和优化。另外，该软件包括了强大的 FPGA Express 综合系统，并将它们完美地集成到统一的项目管理器中，从而为支持使用第三方 IP 核提供了有利的条件。因此，该软件成为 IC 制造业最强大的 EDA 设计工具之一。

ISE 是 Xilinx 公司推出的基于 CPLD/FPGA 的集成开发软件，它提供给用户一个从设计到综合、布线、仿真、下载的全套解决方案，可以方便地与其他 EDA 工具接口。其中，设计输入可以使用原理图编辑器、文本编辑器和状态机编辑器；HDL 综合可以使用本公司自己开发的 XST，也可以使用 Synplify Pro (Synplicity 公司)、Leonardo Spectrum (Mentor 公司) 等第三方综合系统；设计仿真可以使用图形化的测试激励生成器，以及 Model Tech 公司的仿真工具 ModelSim。

除了上述介绍的几种开发 PLD 的软件外，许多 EDA 公司还提供了各种专业软件，用以进行更复杂和更高效的设计。这些软件分别是：用于帮助用户完成原理图和 HDL 文本编辑输入的设计输入工具；用于对设计输入文件进行编译、优化和转换的逻辑综合器；对设计进行模拟仿真的仿真器；用于 IC 版图设计的版图设计工具。

1.5 硬件描述语言概述

硬件描述语言 (Hardware Description Language, HDL) 是以高级语言为基础，以形式化方式描述电路的结构和行为，并用于模拟和综合的高级描述方法，目的是用软件的方法实现硬件电路的设计、实现 RTL 级仿真、验证设计的正确性。

随着 ASIC 的开发和研制，为了提高开发效率，增加已有开发成果的可继承性和缩短开发时间，各 ASIC 研制和生产厂家相继开发了用于各自目的的硬件描述语言。硬件描述语言从诞生至今，已经有百余种之多，并成功地应用于系统开发的设计、综合、仿真和验证阶段，使设计过程达到高度自动化，对 EDA 技术的发展起到了促进和推动作用。目前，常用的硬件描述语言主要有 VHDL、Verilog HDL、ABEL-HDL 和 AHDL 等。

1. VHDL

VHDL 语言的全称为“超高速集成电路硬件描述语言 (Very High Speed Integration Circuits Hardware Description Language)”,它起源于美国国防部提出的超高速集成电路 (Very High Speed Integration Circuits, VHSIC) 计划。由 IEEE (the Institute of Electric 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 标准 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。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 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。

随着 VHDL 的标准化, 出现了一些支持该语言的行为仿真器。20 世纪 90 年代初, 出现了 VHDL 逻辑综合器, 它把标准 VHDL 的语句描述转化为具体电路实现的网表文件, 即基本逻辑元件连接图, 这样 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。

VHDL 具有与具体的硬件电路和设计平台无关的特性, 并具有良好的电路行为描述和系统描述的能力, 在语言易读性和层次化结构化设计方面, 表现了强大的生命力和应用潜力。因此, VHDL 支持自顶向下与自底向上或混合方法的设计, 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。

VHDL 主要用于描述数字系统的结构、行为、功能和接口。VHDL 的语言形式是在一般的计算机高级语言的基础上, 加上一些具有硬件特征的语句。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 (或称可视部分及端口) 和内部 (或称不可视部分)。在对一个设计实体定义了外部界面后, 一旦其内部开发完成, 其他的设计就可以直接调用这个设计实体。

2. Verilog HDL

Verilog HDL 在许多领域的应用也很普遍。它是在 C 语言的基础上发展起来的一种描述语言, 于 1983 年由 GDA (Gateway Design Automation) 公司创造的, 最初只设计了一个仿真与验证工具, 之后又陆续开发了相关的故障模拟与时序分析工具。1985 年 GDA 公司推出它的第三个商用仿真器 Verilog-XL, 获得了巨大的成功, 从而使得该语言得到迅速的推广和应用。1989 年 GDA 公司被 Cadence 公司收购, 使得 Verilog HDL 成为该公司的独家专利。1995 年成为 IEEE 标准, 编号: IEEE Std 1364-1995。

Verilog HDL 的最大特点就是易学、易用, 如果有 C 语言的编程经验, 可以在一个较短的时间内很快地学习和掌握。支持 Verilog HDL 的 EDA 工具较多, 适合于寄存器传输级 (RTL) 和门电路级的描述, 其综合过程比 VHDL 简单, 但在高级行为描述方面不如 VHDL。

1998 年 Verilog HDL 通过新的标准, 将 Verilog HDL-A 并入 Verilog HDL 设计中,

使其 Verilog HDL 不仅支持数字逻辑电路的描述,还支持模拟电路的描述,因而在数模混合设计中得到了广泛的应用,特别是在亚微米和深亚微米专用集成电路及高密度 FPGA 中, Verilog HDL 的发展前景将更加广阔。

3. ABEL-HDL

ABEL-HDL 起源于可编程器件的设计,是由美国 Data I/O 公司开发的一种硬件描述语言,可支持各种行为的输入方式和层次结构的逻辑描述,包括逻辑方程、状态图和真值表等多种逻辑描述方式,因此,被广泛用于各种可编程器件的逻辑功能设计。但它不是标准化的硬件描述语言, Lattice 公司的 EDA 集成开发工具支持这种语言。

4. AHDL

AHDL 是由 Altera 公司根据其生产的 MAX 系列器件和 FLEX 系列器件的特点专门设计的一套完整的硬件描述语言。AHDL 是一种模块化的硬件描述语言,特别适合于描述复杂的组合逻辑、状态机、真值表和参数化的逻辑,但它不是标准化的硬件描述语言。

目前,在众多硬件描述语言中, VHDL 和 Verilog HDL 作为 IEEE 的工业标准语言,得到了许多 EDA 公司的支持,在电子工业领域中,已成为通用硬件描述语言。现在 VHDL 和 Verilog HDL 将承担几乎全部的数字系统设计任务。

1.6 基于 EDA 软件的 CPLD/FPGA 设计流程

可编程逻辑器件的开发系统由硬件和软件两个部分组成。硬件包括计算机和专用的编程器、编程电缆等,软件泛指各种编程软件。了解基于 EDA 技术进行 CPLD/FPGA 设计开发的流程,对于正确选择和使用 EDA 软件、优化设计项目、提高设计效率十分有益。一个完整的 EDA 设计流程,即是自顶向下设计方法的具体实施途径,也是 EDA 工具软件本身的组成结构。

EDA 设计流程图如图 1-1 所示,该设计流程包括设计输入、编译、仿真、编程四个部分。以下分别介绍各设计模块的功能特点。

1. 设计输入

设计输入是指利用 EDA 工具中的文本编辑器或图形编辑器等对系统的逻辑功能进行描述,以文本方式或图形方式表达出来,进行编辑和编译,变成 HDL 文件格式。常用的设计输入方式有以下四种方式。

(1) 原理图输入方式。原理图输入方式是指在 EDA 工具软件的图形编辑界面上,绘制出能完成预定功能的电路原理图。原理

图输入方式与传统的器件连接方式完全一样,很容易掌握,但同时也存在缺点:随着设计规模的增大,设计的易读性迅速下降;一旦完成,电路结构的改变将十分困难。

(2) 文本输入方式。文本输入方式是指在 EDA 工具软件的文本编辑界面上,用硬件描述语言表述电路的设计输入方式。这种方法与传统的计算机高级语言输入类似。该

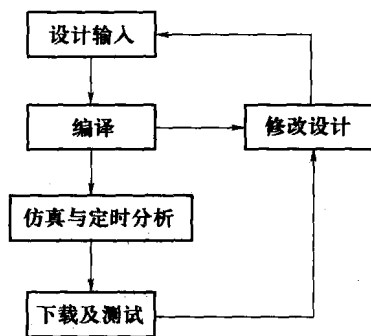


图 1-1 基于 EDA 的 CPLD/FPGA 设计流程

方法的优点是输入效率高,在不同的设计输入之间转换十分方便,成为目前最一般化、最具普遍性的输入方法。

(3) 波形输入方式。波形输入方式是指在 EDA 工具软件的波形图编辑界面上,绘制电路输入信号与输出信号的波形关系,然后由 EDA 编辑器根据上述波形完成电路的设计。

(4) 状态图输入方式。状态图输入方式是指依据电路的控制条件和状态转换的因果关系,在 EDA 工具软件的状态图编辑界面上,绘制时序电路的状态转换图。这种方式简化了状态机的设计,比较方便。

2. 编译

编译包括排错、数据网表文件提取、逻辑综合、装配文件(仿真文件与编程配置文件)生成以及基于目标器件的工程是时序分析等。

3. 仿真

仿真包括功能仿真、时序仿真两部分,开发者可以利用软件的仿真功能来验证设计项目的逻辑功能是否正确。功能仿真是在设计输入完成之后,选择具体器件进行编译之前进行的逻辑功能验证,因此又称为前仿真。此时的仿真没有延时信息或由系统添加的微小标准延时,这对于初步的功能检测非常方便。时序仿真是在选择了具体器件并完成布局、布线之后进行的时序仿真,因此又称为后仿真或延时仿真。在编译通过以后,对系统和各模块进行时序仿真,分析其时序关系,对估计设计的性能及检查和消除竞争冒险等是非常必要的。

4. 下载及测试

把经过仿真后的编程文件通过编程器(Programmer)将设计文件下载到实际芯片中,最后测试芯片在系统中的实际运行性能。

在设计过程中,如果出现错误则需重新回到设计输入阶段,改正错误或调整电路后重复上述过程。

1.7 IP 核

一个较复杂的数字系统往往由许多功能块构成。一个新的设计虽然包含着体现设计者新思想的核心单元,但其中也有通用的功能单元,如快速傅里叶变换(Fast Fourier Transform, FFT)转换器、无限冲击响应(Finite Impulse Response, FIR)滤波器、PCI 总线接口等。这些通用单元具有可重用性,适用于不同的系统。FPGA 厂家及其第三方合作者预先设计好这些通用单元并根据各种 FPGA 芯片的结构对于布局布线进行优化,从而构成具有知识产权的功能模块,称为 IP (Intellectual Property) 模块,也称为 IP 核 (Intellectual Property Core)。在大容量 FPGA 中使用 IP 核设计复杂的系统电路,可以降低设计风险,减少资金投入,缩短研发周期,从而提供设计系统的可靠性、适应性和实用性。

美国 Dataquest 咨询公司将半导体产业的 IP 定义为用于 ASIC、ASSP 和 PLD 等当中,并且是预先设计好的电路模块。IP 核模块有行为(Behavior)、结构(Structure)和物理(Physical)三个不同级别的设计,对应描述功能的不同分为三类,即软核(Soft IP Core)、