

EDA技术实用丛书

CPLD/FPGA 设计及应用

罗朝霞 高书莉 编著

附光盘
CD-ROM



 人民邮电出版社
POSTS & TELECOM PRESS

EDA 技术实用丛书

CPLD/FPGA 设计及应用

罗朝霞 高书莉 编著

人民邮电出版社

北 京

图书在版编目 (CIP) 数据

CPLD/FPGA 设计及应用/罗朝霞, 高书莉编著. —北京: 人民邮电出版社, 2007.7
(EDA 技术实用丛书)

ISBN 978-7-115-15958-8

I. C... II. ①罗...②高... III. 可编程序逻辑器件 IV. TP332.1

中国版本图书馆 CIP 数据核字 (2007) 第 034058 号

内 容 提 要

本书从实际应用角度出发, 详细介绍了基于 EDA 技术的 CPLD/FPGA 设计流程, CPLD/FPGA 典型产品的结构原理, VHDL 硬件描述语言以及 Quartus II 开发软件; 列举了 VHDL 语言的各种语法结构以及相应例程; 还介绍了数字系统设计的一般描述方法和设计过程; 通过一些常用电路及具有使用价值的实例设计, 使读者掌握应用可编程器件进行设计的方法和技巧; 最后讨论了几个 CPLD/FPGA 设计中的基本问题。

本书适合 CPLD/FPGA 硬件工程师和集成电路 (IC) 工程师阅读和参考, 也可作为高等院校通信工程、电子工程、计算机应用技术、数字信号处理、图像处理等专业本科生或研究生的教材。

为了便于读者更好地阅读, 本书带有配套的光盘。

EDA 技术实用丛书

CPLD/FPGA 设计及应用

◆ 编 著 罗朝霞 高书莉

责任编辑 陈万寿

◆ 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>

北京艺辉印刷有限公司印刷

新华书店总店北京发行所经销

◆ 开本: 787×1092 1/16

印张: 17.25

字数: 429 千字

印数: 1—4 000 册

2007 年 7 月第 1 版

2007 年 7 月北京第 1 次印刷

ISBN 978-7-115-15958-8/TN

定价: 37.00 元 (附光盘)

读者服务热线: (010) 67129258 印装质量热线: (010) 67129223

前 言

随着电子技术的不断发展与进步, CPLD/FPGA 有着越来越广阔的应用前景, 它和 DSP 及 CPU 并称为电子系统的三大基石。CPLD/FPGA 具有体系结构先进、支持在线编程、应用灵活、集成度高、功能强大、设计周期短、开发成本低、开发工具先进、产品标准化程度高、性能稳定以及可在线检验等优点, 被广泛应用于通信、仪器仪表、工业控制、信息处理等系统的设计与生产中。而掌握 CPLD/FPGA 设计技术已成为当今硬件工程师和 IC 设计工程师的必备技能。

本书从实际应用角度出发, 以提高工程设计能力为目的, 系统地介绍了 CPLD/FPGA 的基本设计方法。

全书内容共分 8 章。第一章介绍 EDA 技术的基本知识; 第二章介绍可编程逻辑器件的基本原理, 并对 CPLD 与 FPGA 的异同及选择作了说明; 第三章具体介绍了 Altera 的 CPLD/FPGA 器件的结构、特点及编程/配置方法; 第四章介绍目前得到广泛应用的标准化硬件描述语言——VHDL, 并举例说明了该语言在使用时的难点、要点; 第五章通过例子说明了几种使用 Altera 的 Quartus II 开发软件进行设计的方法——原理图输入设计方法、文本输入设计方法、混合输入设计方法及使用宏功能模块设计方法; 第六章介绍了几个 CPLD/FPGA 的常用设计, 包括基本数学运算模块、分频器、存储器及状态机; 第七章介绍数字系统设计的基本概念、设计过程及其方法, 并给出了几个完整的数字系统设计实例。第八章对 CPLD/FPGA 设计中的几个基本问题进行了讨论。

本书第一、二、八章由高书莉编写, 第三、四、五、六、七章由罗朝霞编写, 全书由罗朝霞统稿。非常感谢 Altera 公司徐平波先生为本书编写提供的支持与帮助; 感谢西安邮电学院李轶、张盛和王晓宇为本书编写所做的工作; 另外, 本书在编写过程中参考了不少专家和学者的著作, 在此对参考文献的作者致以深深的谢意。

CPLD/FPGA 技术的发展非常迅速, 我们也一直处于不断的学习之中。虽然经过很大努力, 但由于水平有限, 加之时间仓促, 书中难免有疏漏或错误之处, 敬请读者批评指正 (责任编辑电子邮件: chenwanshou@ptpress.com.cn)。

作 者

目 录

第一章 EDA 概述	1
1.1 EDA 技术的含义	1
1.2 EDA 技术的发展历史	1
1.3 EDA 技术的基本特征	2
1.4 EDA 技术的主要内容	4
1.5 EDA 技术的设计流程	5
1.6 EDA 技术的发展趋势	8
第二章 可编程逻辑器件的基本原理	10
2.1 专用集成电路设计与可编程逻辑器件	10
2.1.1 模拟 ASIC	10
2.1.2 数字 ASIC	11
2.2 可编程逻辑器件分类	12
2.3 可编程逻辑器件的发展历程	14
2.4 可编程逻辑器件的基本结构	15
2.4.1 简单低密度 PLD 的基本结构	15
2.4.2 复杂高密度 PLD 的基本结构	18
2.4.3 CPLD 和 FPGA 的异同	24
2.4.4 CPLD 和 FPGA 的选择	25
第三章 Altera 的 CPLD/FPGA	27
3.1 Altera 的 CPLD/FPGA 器件系列	27
3.2 Altera 的 CPLD 器件	29
3.2.1 MAX7000 器件	30
3.2.2 MAX II 器件	35
3.2.3 Cyclone 器件	36
3.3 Altera 的 CPLD/FPGA 的配置	42
3.3.1 配置方式	42
3.3.2 ByterBlaster II 下载电缆	45
3.3.3 配置芯片	46
3.3.4 CPLD/FPGA 器件的配置	46
第四章 VHDL 硬件描述语言	54
4.1 VHDL 概述	54
4.1.1 VHDL 的特点	54

4.1.2 VHDL 和 VerilogHDL 的比较	54
4.2 VHDL 语言基础知识	55
4.2.1 VHDL 的程序结构	55
4.2.2 VHDL 的语言元素	63
4.3 VHDL 的基本语句	68
4.3.1 VHDL 的顺序语句	68
4.3.2 VHDL 的并行语句	75
4.4 VHDL 的子程序	89
4.4.1 过程的定义和调用	90
4.4.2 函数的定义和调用	90
第五章 Quartus II 开发软件	92
5.1 Quartus II 简介	92
5.2 Quartus II 原理图输入设计方法	94
5.2.1 创建设计工程	94
5.2.2 设计输入	98
5.2.3 设计项目的编译	100
5.2.4 设计项目的仿真验证	103
5.2.5 引脚分配	106
5.2.6 器件编程	108
5.3 Quartus II 文本输入设计方法	110
5.3.1 设计输入	110
5.3.2 项目编译	113
5.3.3 仿真验证	117
5.4 Quartus II 混合输入设计方法	119
5.4.1 创建工程	119
5.4.2 译码电路模块的设计	119
5.4.3 顶层设计	120
5.4.4 仿真验证	123
5.5 Quartus II 调用宏功能模块的设计方法	124
5.5.1 宏功能模块概述	124
5.5.2 以图形方式调用宏功能模块进行设计	125
5.5.3 以文本方式调用宏功能模块进行设计	130
第六章 CPLD/FPGA 的常用设计	132
6.1 基本数学运算模块	132
6.1.1 加法器	132
6.1.2 乘法器	139
6.1.3 减法器	142
6.1.4 除法器	143

6.2 分频器设计	145
6.2.1 偶数分频器设计	145
6.2.2 奇数分频器设计	148
6.2.3 半整数分频器的设计	151
6.3 存储模块设计	153
6.3.1 只读存储器 (ROM)	153
6.3.2 随机存储器 (RAM)	159
6.3.3 顺序存取存储器 (SAM)	163
6.4 状态机设计	166
6.4.1 状态机的特点及 VHDL 描述	166
6.4.2 状态机设计举例	172
6.4.3 状态机编码	183
6.4.4 状态机设计中需要注意的问题	187
第七章 数字系统设计实例	188
7.1 数字系统设计概述	188
7.1.1 数字系统的组成	188
7.1.2 数字系统设计方法	188
7.1.3 数字系统设计的一般过程	189
7.1.4 两种数字系统设计方法的比较	190
7.2 同步信号提取电路的设计	191
7.2.1 位同步信号提取电路的设计	191
7.2.2 帧同步信号提取电路的设计	198
7.3 IIR 滤波器的设计	206
7.3.1 IIR 滤波器设计要求	208
7.3.2 IIR 滤波器设计方案	208
7.3.3 IIR 滤波器的实现	210
7.3.4 IIR 滤波器的其他改进型 CPLD/FPGA 实现方案	213
7.4 DDS 及相关设计	219
7.4.1 DDS 的基本原理	220
7.4.2 DDS 设计	221
7.4.3 三相多波形信号发生器设计	228
7.4.4 调制信号发生器的设计	234
7.5 曼彻斯特编解码器	235
7.5.1 曼彻斯特码编码规则	235
7.5.2 曼彻斯特编码器设计	236
7.5.3 曼彻斯特解码器设计	242
7.6 交织/解交织器的设计	246
7.6.1 交织/解交织器的原理	246
7.6.2 分组交织/解交织器的设计	247

第八章 CPLD/FPGA 设计中的基本问题	259
8.1 有限字长的影响	259
8.2 时钟问题	259
8.2.1 建立时间和保持时间	260
8.2.2 全局时钟	260
8.2.3 门控时钟	260
8.2.4 多级逻辑时钟	261
8.2.5 行波时钟	262
8.2.6 多时钟系统	262
8.3 清零和置位信号	263
8.4 毛刺的产生及消除	264
8.4.1 信号毛刺的产生及判断	264
8.4.2 消除信号毛刺的方法	264
参考文献	267

第一章 EDA 概述

随着电子技术的飞速发展，现代电子产品渗透到了社会的各个领域，并有力地推动着社会生产力的发展和社会信息化程度的提高。现代电子产品在性能提高、复杂度增大的同时，价格却一直呈下降趋势，而且产品更新换代的步伐也越来越快，实现这种进步的主要原因就是生产制造技术和电子设计技术的发展。前者以微细加工技术为代表，目前已进展到深亚微米阶段，可以在几平方厘米的芯片上集成数千万个晶体管；后者的核心就是 EDA 技术。

1.1 EDA 技术的含义

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 技术有广义 EDA 技术和狭义 EDA 技术之分。广义 EDA 技术是指以计算机和微电子技术为先导，汇集了计算机图形学、数据库管理、图论和拓扑逻辑、编译原理、微电子工艺与结构学以及计算数学等多种计算机应用学科最新成果的先进技术。狭义 EDA 技术是指以大规模可编程逻辑器件为载体，以硬件描述语言 (HDL, Hardware Description Language) 为系统逻辑的主要表达方式，借助功能强大的计算机，在 EDA 工具软件平台上，对用 HDL 描述完成的设计文件，自动完成用软件方式设计的电子系统到硬件系统的逻辑编译、逻辑简化、逻辑分割、逻辑综合和优化、逻辑布局布线、逻辑仿真，直至对特定目标芯片的适配编译、逻辑映射、编程下载等工作，最终形成集成电子系统或专用集成电路 (ASIC, Application Specific Integrated Circuits) 芯片的一门新技术。本书中提到的 EDA 技术指的是狭义的 EDA 技术。

1.2 EDA 技术的发展历史

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

1. 计算机辅助设计 (CAD) 阶段

计算机辅助设计主要是指将原先由手工操作完成的 IC 版图编辑和印制电路板 (PCB, Printed Circuit Board) 布局布线等方面的工作借助相应的软件工具由计算机辅助完成。

早期的电子系统硬件设计采用的是分立元件。随着中小规模集成电路的开发和应用，电子系统设计进入到大量选用中小规模标准集成电路进行设计的初级阶段。这时，传统的手工制图设计 PCB 的方法已无法满足产品复杂性的要求，更不能满足工作效率的要求。因此，人

们开始借助计算机来完成电子设计过程中高度重复性的繁杂劳动。

20 世纪 70 年代主要研制了用于 PCB 布线设计、电路模拟、逻辑模拟及版图绘制等方面的软件工具。随着集成电路规模增大,制作复杂度的提高,20 世纪 80 年代初,许多软件公司主要针对电子产品开发研制出了用于设计、分析、生产、测试等方面的多个独立的软件包。

CAD 阶段虽然研制了一些单独的软件工具,但每个软件工具只能完成其中的一项工作,如果通过顺序循环使用这些软件来完成设计的全过程,则还存在一些问题:一是设计速度低。由于各个软件工具只解决一个领域的问题,且是由不同的公司开发的,若将一个软件工具的输出作为另一个软件工具的输入,必须进行繁琐的人工处理,降低了设计速度。二是修改困难。由于 CAD 阶段的软件工具缺乏系统级的设计考虑,无法提供系统级的仿真和综合,错误往往都是在设计开发的后期才被发现,这使得修改起来十分困难。

2. 计算机辅助工程 (CAE) 阶段

与 CAD 相比,CAE 除了具有图形绘制功能外,增添了通过电气连接网络表将电路功能设计和结构设计相结合以实现工程设计的功能。

CAE 阶段解决了 CAD 阶段存在的问题。由于在 CAE 阶段采用了统一数据管理技术,因而能够将各个设计工具,如原理图输入、编译与连接、逻辑仿真、测试码生成、电路分析、版图自动布局布线等集成为一个完整的系统。该系统按照设计方法学制定的设计流程,可以自动完成从设计输入到版图输出的电子系统设计的全程自动化。也就是说,CAE 阶段的软件工具已经可以进行设计描述、综合与优化和设计结果验证,这不仅为成功设计电子系统创造了有利条件,而且还为设计者的创造性劳动提供了方便。

但是,CAD 和 CAE 阶段取得的成果,并没有真正把设计者从繁重的设计工作中彻底解放出来。在整个设计过程中,自动化和智能化程度还不高,再加上各种设计软件工具界面千差万别,学习使用困难,并且互不兼容,直接影响到设计环节的衔接。由于上述原因,设计者开始追求整个设计的自动化。

3. 电子设计自动化 (EDA) 阶段

20 世纪 90 年代后期,进入到 EDA 阶段。EDA 阶段的软件工具以高级语言描述、系统仿真和综合技术为核心技术。

EDA 阶段软件工具的主要特征是自动化程度大大提高。设计者只需要在前期进行较少的设置便可以完全由计算机完成整个设计,这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。EDA 阶段系统设计主要以并行设计工程的方式和系统级目标设计方法为支持,其核心是可编程逻辑器件的设计。可编程逻辑器件本身的可重复编写特性使电子设计的灵活性和效率大大提高。

1.3 EDA 技术的基本特征

EDA 技术具有下列基本特征。

1. EDA 技术采用自顶向下的设计方法

自顶向下的设计方法是 EDA 技术的首选设计方法,它利用功能分割手段将设计由上至下进行层次化和模块化,即分层次、分模块地对电子系统进行设计和仿真。功能分割采用逐级分割的方式,首先将系统分割成各个功能子模块,然后再将各个功能子模块分解为逻辑块,而逻辑块又可以分割为更小的逻辑块和电路。按照这样的分割方式,设计者可以将一个复杂的电子系统逐步细化,并将功能模块化。

从以上描述可以看出,自顶向下的设计方法实际上是基于芯片的系统设计方法,它在功能划分、任务分配和设计管理上具有一定的优越性。该方法有助于在设计早期发现结构设计中的错误,提高设计的一次成功率。

2. EDA 技术的设计语言是硬件描述语言

硬件描述语言(HDL, Hardware Description Language)是对硬件电路进行行为描述、寄存器传输描述或结构化描述的一种新兴语言,它比传统的电路原理图更能有效地表示硬件电路的特性,并可借鉴高级程序语言设计的具体方法对硬件电路的行为和功能结构进行高度抽象化的描述。此外,硬件描述语言还可以对硬件电路的设计进行不同层次、不同领域的模拟验证和综合优化等处理,从而实现硬件电路设计的高度自动化。采用硬件描述语言进行设计是当前发展的趋势,它可以大大降低成本,缩短研制周期,既适用于小批量产品开发,也适用于大批量产品的样品研制。

目前,最有代表性的硬件描述语言是美国国防部开发的 VHDL(VHSIC Hardware Description Language)和 Verilog 公司开发的 Verilog HDL。一般而言,两种语言的侧重点稍有不同,前者非常适合于大型电子系统的描述;后者更适合于硬件细节的描述。

3. EDA 技术具有逻辑综合和优化的功能

逻辑综合功能是指将较高层次的抽象描述转化到较低级别抽象的一种方法,即将设计的描述转化成底层电路表示。通常,逻辑综合的结果是电路网表或者是一组逻辑方程。

优化功能是指根据布尔方程功能等效的原则,利用逻辑综合生成的结果,采用更小更快的综合结果代替一些复杂的单元,然后根据制定的目标可以映射成新的网表或者一组逻辑方程。优化的作用是将电路设计的时延缩短到最小和有效利用资源。几乎所有的 EDA 综合工具都可以利用约束条件对电路设计进行优化。一般情况下,常用的约束条件主要包括时间约束和面积约束。

4. EDA 技术采用开放性和标准化的软件框架

软件框架是指一种软件平台结构,它的主要功能是为 EDA 工具提供相应的操作环境。框架的关键在于提供与硬件平台无关的图形用户界面、工具之间的通信、设计数据 and 设计流程的管理,此外,框架还包括各种与数据库相关的服务。

任何一个 EDA 工具系统只有建立了一个符合标准的开放式框架结构,才可以接纳其他厂商的 EDA 工具,从而形成软件工具之间的无缝连接。另外,硬件描述语言的标准化使得不同厂商的 EDA 工具可以集成在一个工作平台上进行工作,这将导致 EDA 工具的软件框架标准化和具有开放性。正是由于这两方面的原因,EDA 技术只有采用开放性和标准化的软件

框架, 才能保证不同 EDA 工具的兼容性, 从而实现资源共享。

1.4 EDA 技术的主要内容

EDA 技术作为一门迅速发展的新技术, 它涉及面广, 内容丰富, 从实用的角度讲, 应主要掌握这几方面的内容: 可编程逻辑器件、硬件描述语言和软件开发工具。大规模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. 可编程逻辑器件

可编程逻辑器件是一种由用户编程以实现某种逻辑功能的新型器件。可编程逻辑器件也称为可编程 ASIC, 它是 EDA 技术的物质基础。这里的可编程逻辑器件主要是指高密度可编程逻辑器件, 即复杂可编程逻辑器件 (CPLD) 和现场可编程门阵列 (FPGA)。有关可编程逻辑器件方面的知识, 本书第二章有详细介绍。

2. 硬件描述语言

硬件描述语言是 EDA 技术的重要组成部分。它是一种用于设计硬件电子系统的计算机语言, 用软件编程的方式来描述电子系统的逻辑功能、电路结构和连接形式。与传统的门级描述方式相比, 硬件描述语言的特点是可以借鉴高级程序语言的设计方法, 从而可以对硬件电路的行为和功能结构进行高度抽象化的描述; 因而它更适合大规模系统的设计。

采用硬件描述语言进行电子系统设计可以提高效率, 增加开发成果的可继承性以及缩短开发时间。此外, 硬件描述语言还可以通过对硬件电路的设计进行不同层次、不同领域的模拟验证和综合优化等处理, 从而实现硬件电路设计的高度自动化。

采用硬件描述语言来设计硬件电路既适用于小批量产品开发, 也适用于大批量产品的样品研制, 因而得到了广泛地应用。在目前的电子设计中, 设计人员采用的最有代表性的硬件描述语言是美国国防部开发的 VHDL 语言、Verilog 公司开发的 Verilog HDL 语言。本书主要介绍 VHDL 语言。

3. 软件开发工具

EDA 软件开发工具是指以工作站或者高档计算机为基本工作平台, 利用计算机图形学、拓扑逻辑学、计算数学和人工智能等多种应用学科的最新成果而开发出来的一套软件工具, 它是一种帮助设计工程师进行电子系统设计的辅助工具。

EDA 软件开发工具是 EDA 技术的强有力支持。一般的 EDA 软件开发工具包括: 编辑器、仿真工具、检查/分析工具、优化/综合工具和布局布线工具等。

编辑器的主要功能是用来对设计输入进行图形或文本等方面的编辑操作, 它通常包括图形编辑器和文本编辑器。图形编辑器用来编辑表示器件的几何图形、电子系统的框图以及原理图等。文本编辑器在系统级上用来编辑电子系统的自然描述语言, 在其他层次上主要用来编辑电路的硬件描述语言文本。

仿真器是用来设计仿真操作的一种工具，它是用来评价 EDA 工具的一项重要指标，其功能是用来验证设计的正确性和准确度。仿真是 EDA 技术中的一项非常重要的技术，采用仿真可以在开发设计的早期阶段发现设计中的错误，这样便可以大大减少设计重复和修改的次数及时间，从而提高了设计者的工作效率。

检查/分析工具是用来对设计者的具体设计进行编译、检查和分析，目的是发现设计中的错误和对可能的结果进行分析。通过分析检查/分析工具产生的各种报表文件，设计者可以对设计系统中的错误、系统性能以及相应的各种时序关系等有一个清楚地认识，从而能及时发现设计中的错误和检查系统性能能否满足设计的需求。

优化/综合工具是用来完成优化功能和逻辑综合功能的一种开发工具。其中，优化功能是指根据布尔方程功能等效的原则，采用不同的优化方法来对设计进行优化操作，从而提高设计系统的性能和占用较少资源；逻辑综合功能是将抽象描述转化成电路网表或者是一组逻辑方程的形式，目的是方便设计系统的具体实现。

布局布线工具实现由逻辑设计到物理实现的映射。最终物理实现对应的器件不同，则各自的布局布线工具也会有较大差异。

EDA 软件工具开发厂商大体可分两类：一类是 EDA 专业软件公司，较著名的有 Mentor Graphics、Cadence Design Systems、Synopsys 和 Viewlogic Systems 等；另一类是半导体器件厂商，为了销售他们的产品而开发 EDA 工具，较著名的公司有 Altera、Xilinx 和 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作出优化设计，提高资源利用率，降低功耗，改善性能，比较适合于产品开发单位使用。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，注重器件生产、设计服务和 IP 模块的提供。

1.5 EDA 技术的设计流程

EDA 技术的设计流程属于设计方法学的范畴，它对于整个设计活动的作用表现在这几个方面：设计流程是规范设计活动的准则，它使得设计活动在各个阶段有了交流、比较的可能；设计流程规定了工具的选择和使用，为各种工具之间的接口提供了可能；设计流程规定了设计者的工作次序与内容，这使得在同一个设计项目中可以进行多人分工与协作，从而缩短设计周期；设计流程自身的科学性也保障了所进行的设计的正确性和可靠性。EDA 技术的设计流程对设计活动具有重要的指导意义，可以提高设计活动的效率和可靠性，有利于设计活动的管理和交流。

在电子设计领域，自顶向下的设计方法是 EDA 技术的首选方法。本节首先介绍一下自顶向下的 EDA 技术设计流程，然后详细地介绍一个基于 CPLD/FPGA 的 EDA 技术的工程设计流程。

1. EDA 技术设计流程

自顶向下的 EDA 技术设计流程包括以下设计阶段：

(1) 提出设计说明书,即用自然语言表达系统的功能特点和技术参数等。

(2) 建立 VHDL 行为模型,即将设计说明书转化为 VHDL 行为模型。在 VHDL 建模过程中,可以使用满足 IEEE 标准的 VHDL 的所有语句而不考虑可综合性。建模行为的目的是通过 VHDL 仿真器对整个系统进行系统行为仿真和性能评估。

(3) VHDL 行为仿真。本阶段可以利用 VHDL 仿真器对顶层系统的行为模型进行仿真测试,检查模拟结果,以便进行修改和完善。VHDL 行为仿真过程与最终实现的硬件没有任何关系,无需考虑硬件实现中的技术细节,测试结果主要是对系统纯功能行为的考察,有些 VHDL 语句表达是不能用真实的硬件来实现的,它们仅起帮助了解系统在各种条件下功能特性的作用。

(4) VHDL-RTL 级建模。本阶段必须将 VHDL 的行为模型表达为 VHDL-RTL 级模型。VHDL-RTL 级模型是用 VHDL 中可综合子集中的语句完成的,即可以最终实现目标器件的描述。

(5) 前端功能仿真。前端功能仿真也称功能仿真,就是对 VHDL-RTL 级模型进行的仿真。尽管 VHDL-RTL 级模型是可综合的,但对它的功能仿真仍然与硬件无关,仿真结果表达的是可综合模型的逻辑功能。

(6) 逻辑综合。使用逻辑综合工具将 VHDL 行为描述转化为结构化的门级电路。在 ASIC 设计中,门级电路可以由 ASIC 库中的基本单元组成。

(7) 测试向量生成。本阶段主要是针对 ASIC 设计的。CPLD/FPGA 设计的时序测试文件主要产生于适配器。对 ASIC 的测试向量文件是综合器结合含有版图硬件特性的工艺库后产生的,用于对 ASIC 的功能测试。

(8) 功能仿真。利用获得的测试向量对 ASIC 的设计系统和子系统的功能进行仿真。

(9) 结构综合。主要综合产生的表达逻辑连接关系的网表文件,结合具体的目标硬件环境进行标准单元调用、布局/布线和满足约束条件的结构优化配置。

(10) 门级时序仿真。本阶段将使用门级仿真器或仍然使用 VHDL 仿真器,进行门级时序仿真,以便了解更接近目标器件工作的功能时序。这里仍可使用 VHDL 仿真器是因为结构综合后能同步生成 VHDL 格式的时序仿真文件。ASIC 设计将本阶段进行的仿真称为布局后仿真,它将带有从布局布线得到的精确时序信息映射到门级电路重新进行仿真,以检查电路时序,并对电路功能进行最后检查,仿真成功后即可将设计提供给硅铸造生产工序。

(11) 硬件测试。本阶段对最后完成的 ASIC 或 CPLD/FPGA 硬件系统进行检查和测试。

2. 基于 CPLD/FPGA 的 EDA 技术的工程设计流程

基于 CPLD/FPGA 器件的 EDA 技术的工程设计流程如图 1-1 所示。

(1) 源程序的编辑和编译

利用 EDA 技术进行一项工程设计,首先需要将其用一定的逻辑表达手段表达出来,然后进行排错编译,变成 VHDL 文件格式,为下一步的逻辑综合做准备。EDA 工具的文本编辑器或图形编辑器允许设计者将设计用文本或者图形的方式表达出来。

(2) 逻辑综合和优化

逻辑综合,就是将电路的高级语言或原理图描述转换成低级的,可与 CPLD/FPGA 或构成 ASIC 的门阵列基本结构相映射的网表文件。而逻辑映射就是针对给定硬件结构组件,将电路的高级描述进行编译、优化、转换和综合得到门级电路甚至更底层的电路描述文件。网表文件就是按照某种规定描述电路的基本组成和如何相互连接的文件。

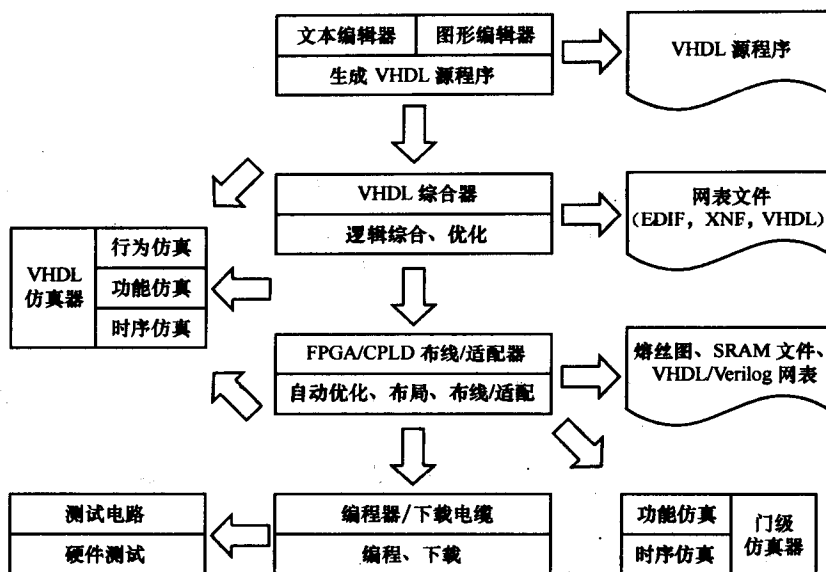


图 1-1 EDA 技术的工程设计流程

HDL 综合器对 VHDL 源文件的综合是针对某一 CPLD/FPGA 产品系列的，综合后的结果可以为硬件系统所接受，具有硬件可实现性。需要注意的是：由于 VHDL 仿真器的行为仿真功能是面向高层次的系统仿真，只能对 VHDL 的系统描述作可行性的评估测试，不针对任何硬件系统，因此基于这一仿真层次的许多 VHDL 语句不能被 HDL 综合器所接受。不能被 HDL 综合器接受的语句在现阶段是无法在硬件系统中实现的。

(3) 目标器件的布线/布局

目标器件的布线/布局又称为适配。所谓逻辑适配，就是将由综合器产生的网表文件针对某一具体的目标器件进行逻辑映射操作，其中包括底层器件配置、逻辑分割、逻辑优化、布线与布局操作等，配置于指定的目标器件中，产生最终的下载文件。适配由适配器来完成，适配器又称为结构综合器。适配完成后可以利用适配所产生的仿真文件作精确的时序仿真，同时产生可用于编程的文件。适配所选定的 CPLD/FPGA 目标器件必须属于原综合器指定的目标器件系列。通常，EDA 软件中的综合器可由专业的第三方 EDA 公司提供，而适配器则需由 CPLD/FPGA 供应商自己提供，因为适配器的适配对象直接与器件结构相对应。

(4) 仿真

设计的过程中有三种仿真：行为仿真、功能仿真和时序仿真。

行为仿真只是根据 VHDL 的语义进行的，与具体电路没有关系，它是将 VHDL 设计源程序直接送到 VHDL 仿真器中所进行的仿真。该仿真可以充分发挥 VHDL 中的适用于仿真控制的语句及有关的预定义函数和库文件。

功能仿真仅对 VHDL 描述的逻辑功能进行测试模拟，以了解其实现的功能是否满足原设计的要求，仿真过程不涉及具体器件的硬件特性，如延时特性，它是将综合后的 VHDL 网表文件再送到 VHDL 仿真器中所进行的仿真。

时序仿真是将布线器/适配器所产生的 VHDL 网表文件送到 VHDL 仿真器中所进行的仿真。该仿真将器件特性加以考虑，故可得到精确的时序仿真结果。经适配处理后生成的 VHDL 网表文件中包含了较为精确的延时信息，网表文件中描述的电路结构与适配后的结果

是一致的。

需要注意的是，图 1-1 中有两个仿真器，一是 VHDL 仿真器，另一个是门级仿真器，它们都能进行功能仿真和时序仿真。所不同的是仿真用的文件格式不同，即网表文件不同。

(5) 目标器件的编程/下载

把适配后生成的下载或编程文件，通过编程器或编程电缆下载到 CPLD/FPGA 中完成具体的系统设计。通常，把对 CPLD 的下载称为编程 (Program)，对 FPGA 中的 SRAM 进行直接下载的方式称为配置 (Configure)。但对于反熔丝结构和 Flash 结构的 FPGA 的下载以及对 FPGA 的专用配置 ROM 的下载仍称为编程。

(6) 硬件仿真/硬件测试

硬件仿真就是在 ASIC 设计中，常利用 CPLD/FPGA 对系统的设计进行功能检测，通过后再将其 VHDL 设计以 ASIC 形式实现的过程。

硬件测试就是将含有载入了设计的硬件系统进行统一测试，根据设计项目在目标系统上的实际工作情况验证所设计的系统是否符合设计要求。

硬件仿真和硬件测试的目的，是为了在更真实的环境中检验 VHDL 设计的运行情况，特别是对于 VHDL 程序设计上不是完全规范、语义上含有一定歧义的程序。由于目标器件功能的可行性约束，综合器对于设计的理解常在有限范围内选择；而 VHDL 仿真器的理解是纯软件行为，其理解的选择范围要宽得多。这种理解上的偏差将会导致仿真结果和综合后实现的硬件电路在功能上的不一致。除此之外，还有许多其他的因素也会产生这种不一致。因此，进行硬件仿真和硬件测试是必不可少的。

1.6 EDA 技术的发展趋势

随着科学技术的不断发展，EDA 技术主要呈现出以下发展趋势。

1. EDA 技术使开发工具得到进一步的发展

当今，EDA 开发工具正在朝着功能强大、简单易学、使用方便的方向发展，主要体现在：

(1) EDA 工具的 PC 平台化。在过去相当长的一段时间内，EDA 开发工具价格昂贵，操作平台是工作站和 UNIX 操作系统，这大大妨碍了其迅速普及。随着 PC 性能的不断提高，在 EDA 工具厂商和科研人员共同努力下，基于 PC 平台的 EDA 开发工具功能逐步完善和齐全，因而得到了迅速普及。

(2) 设计输入工具的发展。设计输入工具可将设计者采用自己熟悉的设计方式，如状态图、框图、逻辑方程等建立的设计文件转换为硬件描述语言文件。

(3) EDA 工具具有混合信号处理能力。具有该能力的 EDA 工具能处理含有数字信号处理、专用集成电路宏单元、数模/模数转换模块及各种压控振荡器在内的混合系统设计。

(4) 更为有效的仿真工具的发展。仿真是花费时间最多的工作也是占用 EDA 工具资源最多的一个环节。若能建立合理的仿真算法和仿真模型，则仿真工具将有一个较大的发展。

(5) 更为理想的综合工具的开发。电子系统和电路的集成规模越来越大，几乎不可能直接面向版图做设计，若要找出版图中的错误，更是难上加难。综合工具可将设计者的精力从繁琐的版图设计和分析中转移到设计前期的算法开发和功能验证上。好的综合工具将更能有

效地提高设计效率。此外,EDA 工具还应具有增量编译能力。对于 EDA 开发厂商而言,除了加快软件的编译速度外,EDA 开发工具还应该能够减少编译时间,而不需要考虑其编译处理的能力如何。在一个具体的设计过程中,最耗费时间的应该是布局、布线过程,如果能减少该过程的时间,将会大大提高系统设计的效率。随着技术的不断发展,设计者找到了这样一种增量编译方案,即 EDA 开发工具只对上次发生变化的那部分设计进行布局、布线。EDA 开发工具具有增量编译能力将是未来 EDA 技术的一个发展方向,它可大大提高设计效率,从而缩短产品开发周期,进而提高产品的市场竞争能力。

2. EDA 技术促使 ASIC 和 CPLD/FPGA 的融合

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;而可编程逻辑器件的开发费用低廉,能在现场进行编程,但体积大、功能有限,而且功耗大。随着系统开发对 EDA 技术的目标器件各种性能指标要求的提高,ASIC 和 CPLD/FPGA 将取长补短,互相融合,以满足成本和上市速度的要求。

尽管两者的融合并不意味着使 ASIC 更加便宜,或者使 CPLD/FPGA 更加省电。但设计者可将两者的优点结合起来,通过去掉 CPLD/FPGA 的一些功能,以减少开发成本并增加灵活性;也可通过在 ASIC 器件中留出 CPLD/FPGA 的空间来修改设计,以降低设计成本。ASIC 和 CPLD/FPGA 的融合为设计者带来了更多的选择。

3. EDA 技术朝着 ESDA 和 CE 方向发展

电子系统设计自动化(ESDA, Electronic Design Automation)强调建立从系统到电路的统一描述语言,同时考虑仿真、综合与测试,把定时、驱动能力、电磁兼容性、机械特性和热学特性等约束条件加入到设计综合中。然后进行统一的设计描述和优化操作,从而提高系统设计的一次成功率。ESDA 要求系统级设计者改变优先考虑具体实现的传统思路,而是集中精力进行系统的总体设计、综合方案比较和优化设计。这将会是一种全新的设计思路。

ESDA 中所提到的“建立从系统到电路的统一描述语言”实际上是寻求一种高级设计语言。高级语言是系统级设计方法和软硬件联合设计的利器。系统级设计方法是指在系统层次进行设计和仿真,运用系统级综合工具将代码综合为门级网表,然后进行布局、布线的设计流程。系统级设计的优点是有益于自顶向下和分工协作,抽象层次高,支持软硬件联合设计、仿真和结构描述代码一致,优化效果好,缩短了设计周期。现在常用的硬件描述语言是 HDL 语言,在此基础上又发展出了许多抽象程度更高的硬件描述语言,如 SystemVerilog、Superlog、SystemC 等,这些高级 HDL 语言的语法结构更加丰富,更适合做系统级、功能级等高层次的设计描述和仿真。ESDA 除了需要用高级 HDL 语言描述外,更重要的是要得到系统级仿真、综合工具的强有力支持。目前高级 HDL 语言发展迅猛,并逐步完善;而系统级综合和仿真工具还不十分成熟,系统级设计方法的演进还需要相当长的一段时间。

并行工程(CE, Concurrent Engineering)要求 EDA 工具从管理层次上把与系统设计有关的工具、任务、时间和工艺等进行合理安排,设计者使用统一的集成化设计环境,各个设计小组能够共享与设计相关的数据库和其他资源,从而可以同步地进行系统的设计工作。可见,CE 改变了在系统设计中过分依赖专业分工和设计者专业知识的传统设计方法,并且允许在设计过程中投入更多的人力,进行并行工作、协同设计。