



21世纪高等院校**自动化类**实用规划教材

EDA技术与VHDL基础

杨 健 主 编
岳 绚 王永喜 副主编



免费赠送
电子课件

- 以实际技能应用为基础，突出重点知识讲解，注重发挥实例教学的优势。
- 基础知识的学习中穿插大量实例讲解，应用实例教学中穿插基础知识要点和设计经验讲解，使学生能在建立VHDL语言知识体系完整性的同时灵活应用所学知识。



清华大学出版社

013024463

TN702-43

100

21 世纪高等院校自动化类实用规划教材

EDA 技术与 VHDL 基础

杨 健 主 编

岳 绚 王永喜 副主编



TN702-43

100

清华大学出版社

北 京



北航

C1631908

内 容 简 介

本书以 Quartus II 9.1 集成开发环境的使用为例,通过理论知识和实例讲解,让读者熟悉可编程逻辑器件的设计流程,同时介绍了 VHDL、状态机设计、组合和时序逻辑电路设计和常用接口电路设计。

全书重点讲解基础知识,强调基础数字模块的设计与熟练应用。在内容编写上采用纵向和横向相结合的写法,纵向基础知识的学习穿插大量实例讲解,使学生建立 VHDL 知识体系的完整性;横向应用实例的学习穿插基础知识要点和设计经验讲解,使学生掌握理论知识的具体应用,从而帮助读者从不同角度认识 VHDL,提高灵活运用能力,建立自己的设计思路。

本书可作为高职高专电子信息类、计算机应用类等相关专业的教材或参考书,也可作为工程技术人员的参考书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

EDA 技术与 VHDL 基础/杨健主编;岳绚,王永喜副主编. —北京:清华大学出版社,2013
(21 世纪高等院校自动化类实用规划教材)

ISBN 978-7-302-31360-1

I. ①E… II. ①杨… ②岳… ③王… III. ①电子电路—计算机辅助设计—应用软件—高等学校—教材 ②硬件描述语言—程序设计—高等学校—教材 IV. ①TN702 ②TP312

中国版本图书馆 CIP 数据核字(2013)第 013982 号

责任编辑:李春明

装帧设计:杨玉兰

责任校对:周剑云

责任印制:李红英

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社总机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62791865

印 刷 者:北京富博印刷有限公司

装 订 者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:18.5

字 数:445 千字

版 次:2013 年 3 月第 1 版

印 次:2013 年 3 月第 1 次印刷

印 数:1~4000

定 价:33.00 元

产品编号:051171-01

前 言

EDA(electronic design automation)技术经过数十年的不断发展,已经进入快速发展阶段,成为推动现代电子工业发展的重要因素,甚至有人提出 EDA 技术是新世纪电子技术创新的源泉。特别是进入 21 世纪,随着可编程器件集成度的不断提高,开发工具功能的逐步增强,模拟电路的可编程芯片技术得到发展,使 EDA 技术与 ASIC 技术成为现代电子工业的重要支柱。

本书的内容共 10 章。第 1 章对 EDA 技术的相关基础知识进行简要介绍,使读者对 EDA 技术有一个整体的认识。第 2 章对可编程逻辑器件(PLD)的基本原理和编程配置方式进行简单的介绍。第 3 章对硬件描述语言(VHDL)的程序结构做初步讲解,引导读者开始进行深入学习。第 4 章介绍 VHDL 的语言要素和相关语句。第 5 章针对 Altera 公司的可编程逻辑器件,通过使用 Quartus II 9.1 集成开发环境,详细介绍原理图输入方式和 VHDL 文本输入方式的可编程逻辑器件开发过程。第 6 章介绍使用 Quartus II 9.1 集成开发环境实现“自底向上”和“自顶向下”的设计方法。第 7 章介绍使用 VHDL 实现有限状态机设计和在 Quartus II 软件中进行状态图输入法设计的方法。第 8 章介绍 Quartus II 9.1 集成开发环境中宏功能模块的调用方法。第 9 章介绍基础逻辑器件的 VHDL 描述方法。第 10 章通过对典型接口电路的设计与分析,介绍数字电子系统的设计方法,帮助读者进行简单数字系统的设计。

本书以实际技能应用为基础,内容简明扼要,删除不常用或少用的知识点,突出重点知识讲解,强化应用,注重发挥实例教学的优势,叙述上力求深入浅出,将知识点讲解与能力培养相结合,注重培养学生的工程应用能力和解决实际问题的能力。

为解决理论和实践的“冲突点”,本书在内容上采用横向和纵向相结合的写法,纵向基础知识的学习穿插大量实例讲解,使学生建立 VHDL 知识体系的完整性;横向应用实例的学习穿插基础知识要点和设计经验讲解,使学生掌握理论知识的具体应用,从而帮助读者从不同角度认识 VHDL,提高灵活运用能力。

本书由杨健主编,岳绚、王永喜任副主编,胡玫参编。编定分工如下:兰州职业技术学院信息工程系杨健负责制定编写大纲,并编写第 8~10 章;岳绚编写第 4~6 章。兰州工业学院电子信息工程系王永喜编写第 2、3、7 章;胡玫编写第 1 章。另外,本书在编写过程中参考了大量的有关文献资料,特别是一些优秀学者和专家的著作和研究成果,在此谨向他们表示诚挚的谢意。

由于作者水平有限,书中难免出现错误与有待商榷之处,敬请读者批评指正。

编 者

目 录

第 1 章 EDA 技术概述	1	2.2 简单 PLD 基本结构原理	22
1.1 EDA 技术	2	2.3 CPLD 和 FPGA 的基本结构	24
1.1.1 EDA 技术的概念	2	2.3.1 CPLD 的基本结构	25
1.1.2 EDA 技术的特点	2	2.3.2 FPGA 的基本结构	26
1.1.3 EDA 技术的发展	2	2.4 CPLD 和 FPGA 的比较	29
1.2 EDA 技术的知识体系	3	2.5 CPLD 和 FPGA 的编程与配置技术	30
1.2.1 EDA 技术的主要内容	3	2.5.1 Altera 公司的下载电缆	30
1.2.2 可编程逻辑器件	5	2.5.2 Altera 公司 FPGA 器件的 编程/配置模式	31
1.2.3 可编程逻辑语言	5	2.6 可编程逻辑器件主要生产厂商及 典型器件	31
1.2.4 EDA 开发工具	7	本章小结	32
1.3 EDA 设计流程	8	习题	32
1.3.1 设计输入	8	第 3 章 VHDL 程序初步 ——程序结构	35
1.3.2 综合	9	3.1 初识 VHDL 程序	36
1.3.3 适配	9	3.2 VHDL 体系结构概述	38
1.3.4 仿真	10	3.2.1 VHDL 体系结构	38
1.3.5 编程下载	10	3.2.2 库、程序包	39
1.3.6 硬件验证	11	3.2.3 实体	40
1.4 EDA 技术的设计方法	11	3.2.4 结构体	43
1.4.1 基于 VHDL 的自顶向下的 设计方法	11	3.2.5 配置	44
1.4.2 EDA 设计方法与传统数字 系统设计方法的比较	11	3.3 结构体描述方式	45
1.4.3 基于 IP 的设计	12	3.3.1 行为描述方式	45
1.5 给初学者的学习建议	13	3.3.2 数据流描述方式	46
本章小结	14	3.3.3 结构描述方式	48
习题	14	3.4 D 触发器的 VHDL 描述概述	50
第 2 章 可编程逻辑器件基础	17	3.4.1 D 触发器的 VHDL 描述	50
2.1 可编程逻辑器件概述	18	3.4.2 不完整条件语句	54
2.1.1 数字集成电路的分类	18	本章小结	56
2.1.2 可编程逻辑器件的理论基础	19	习题	56
2.1.3 可编程逻辑器件的发展历程	19	第 4 章 VHDL 基础	59
2.1.4 可编程逻辑器件的分类	19	4.1 VHDL 的语言要素	60
2.1.5 可编程逻辑器件的发展趋势	22	4.1.1 VHDL 文字规则	60

4.1.2 数据对象	62	6.2 自顶向下混合设计	148
4.1.3 VHDL 数据类型	67	6.2.1 十六进制计数译码	
4.1.4 VHDL 数据类型转换	69	显示电路设计	148
4.1.5 VHDL 运算符	70	6.2.2 一位全加器的自顶向下	
4.2 VHDL 语句	72	混合设计	155
4.2.1 VHDL 的顺序语句	72	本章小结	158
4.2.2 VHDL 的并行语句	86	习题	158
4.2.3 VHDL 的属性语句	99		
本章小结	101	第 7 章 有限状态机设计	159
习题	101	7.1 有限状态机概述	160
第 5 章 Quartus II 集成开发软件		7.1.1 有限状态机的概念和分类	160
初步	107	7.1.2 有限状态机的状态转换图	161
5.1 Quartus II 软件概述	108	7.1.3 有限状态机的设计流程	161
5.1.1 Quartus II 软件开发流程	108	7.1.4 有限状态机的 VHDL 描述	162
5.1.2 Quartus II 软件的特点	108	7.2 Moore 型状态机	166
5.1.3 Quartus II 软件的图形用户		7.2.1 三进程描述	166
界面	109	7.2.2 双进程描述	168
5.2 原理图编辑方法	109	7.2.3 单进程描述	169
5.2.1 半加器电路输入与编辑	110	7.3 Mealy 型状态机	171
5.2.2 半加器的综合	116	7.4 设计实例——十字路口交通灯	
5.2.3 半加器的仿真	117	控制器	172
5.2.4 半加器的编程下载	124	7.5 Quartus 软件状态图输入法	175
5.3 用文本编辑方法设计编码器	130	本章小结	178
5.3.1 8 线-3 线编码器的文本输入		习题	179
与编辑	131	第 8 章 宏功能模块及应用	181
5.3.2 综合与仿真	132	8.1 Quartus II 宏功能模块概述	182
5.3.3 生成符号文件和 RTL		8.2 宏功能模块定制管理器	183
阅读器	132	8.2.1 宏功能模块定制管理器	
5.4 ModelSim 软件应用	133	的使用	183
5.4.1 ModelSim 软件的使用方法	133	8.2.2 8 位加法计数器的定制	183
5.4.2 使用 ModelSim 仿真		8.2.3 宏功能模块定制管理器	
Quartus II 已有文件	138	文件	187
5.4.3 ModelSim 的时序仿真	138	8.3 宏功能模块的应用	187
本章小结	139	8.3.1 Arithmetic 宏功能模块	187
习题	139	8.3.2 Gates 宏功能模块	191
第 6 章 VHDL 设计方法	143	8.3.3 I/O 宏功能模块	193
6.1 自底向上混合设计——六十进制		8.3.4 Memory Compiler 宏功能	
计数器设计	144	模块	197

本章小结	203	10.2.1 基础知识	256
习题	204	10.2.2 蜂鸣器控制电路的设计	257
第 9 章 VHDL 基本逻辑电路设计	205	10.3 拨码开关控制电路设计	260
9.1 基本组合逻辑电路设计	206	10.3.1 基础知识	260
9.1.1 门电路设计	206	10.3.2 拨码开关控制电路的设计	261
9.1.2 编码器设计	209	10.4 按键消抖电路设计	263
9.1.3 译码器设计	213	10.4.1 同步整形消抖电路的设计	263
9.1.4 加、减法器设计	220	10.4.2 计数法消抖电路的设计	264
9.1.5 双向电路和三态控制 电路设计	223	10.5 数码管控制电路设计	266
9.1.6 ROM 设计	225	10.5.1 基础知识	266
9.1.7 乘法器设计	227	10.5.2 静态数码管显示电路的 设计	267
9.2 基本时序逻辑电路设计	229	10.5.3 动态数码管扫描显示电路 的设计	270
9.2.1 触发器设计	229	10.6 矩阵键盘控制电路设计	274
9.2.2 移位寄存器设计	234	10.6.1 基础知识	274
9.2.3 计数器设计	237	10.6.2 矩阵键盘扫描电路的设计	275
9.2.4 分频器设计	242	10.7 8×8 点阵显示控制电路设计	279
本章小结	248	10.7.1 基础知识	279
习题	248	10.7.2 行扫描 8×8 点阵数码管 显示电路的设计	280
第 10 章 接口电路设计	251	10.7.3 列扫描 16×16 点阵数码管 显示电路的设计	282
10.1 LED 控制电路设计	252	本章小结	286
10.1.1 基础知识	252	习题	286
10.1.2 12 路彩灯控制器的设计	252	参考文献	287
10.1.3 彩灯控制电路的设计	254		
10.2 蜂鸣器控制电路设计	256		



第 1 章

EDA 技术概述

教学目标

通过本章知识的学习，了解电子设计自动化技术的特点与发展，掌握 EDA 技术的定义、硬件描述语言和集成开发工具，掌握 EDA 技术的硬件设计对象，掌握 EDA 技术的设计流程和设计方法。

1.1 EDA 技术

1.1.1 EDA 技术的概念

EDA 技术是微电子技术和计算机技术结合发展的产物,它融多学科于一体,是一门综合性学科。它是以计算机硬件和系统软件为基本的工作平台,集电路和系统、数据库、图形学、图论和拓扑逻辑、计算数学、优化理论等多学科的最新科技成果于一体而研制成的商品化通用支撑软件和应用软件包,其目的在于帮助电子设计工程师在计算机上完成电路的功能设计、逻辑设计、性能分析、时序测试直至 PCB(printed circuit board, 印制电路板)的自动设计等。

狭义 EDA 技术的定义为:EDA 技术就是以计算机为工具,在 EDA 软件平台上,对以硬件描述语言(HDL)为系统逻辑描述手段完成的设计文件自动地完成逻辑编译、逻辑化简、逻辑分割、逻辑综合及优化、逻辑布局布线、逻辑仿真,直至对于特定目标芯片的适配编译、逻辑映射和编程下载等工作的一门技术。本书是一本关于狭义 EDA 技术的教材。

1.1.2 EDA 技术的特点

利用 EDA 技术进行电子系统的设计,具有以下几个特点。

- (1) 用软件的方式设计硬件,加速硬件设计周期。
- (2) “自顶向下”的设计方法,简化设计流程。
- (3) 用软件方式设计的系统到硬件系统的转换是由相关的开发软件自动完成的。
- (4) 设计过程中可用有关软件进行各种仿真,包括时序和功能仿真。
- (5) 系统可现场编程、在线升级,简化系统的设计维护。
- (6) 整个系统可集成在一个芯片上,体积小、功耗低、可靠性高。

1.1.3 EDA 技术的发展

1. 20 世纪 70 年代的计算机辅助设计阶段

20 世纪 70 年代,随着计算机技术的快速发展,人们开始研究各种计算机辅助设计(computer aided design, CAD)技术。CAD 主要表现为一些 PCB 软件,用于布线设计、电路模拟、版图绘制等。CAD 利用计算机的计算功能,将设计人员从大量烦琐的计算和绘图中解脱出来。

2. 20 世纪 80 年代的计算机辅助工程设计阶段

20 世纪 80 年代初,随着集成电路规模的快速发展,出现了计算机辅助工程(computer

aided engineer, CAE)技术,主要表现为设计工具和单元库的完备,并具备原理图输入、编译和连接、逻辑模拟、测试代码生成以及版图自动布局等功能。CAE 软件需要针对产品开发,按照设计、分析、生产、测试等划分阶段,不同阶段使用不同软件,通过顺序循环使用这些软件完成整个设计过程,设计人员开始使用计算机完成大部分设计任务。

3. 20 世纪 90 年代的电子设计自动化阶段

基于 CAD 和 CAE 的设计过程,其自动化和智能化的程度不高,需要使用多个软件完成一个完整的工程,各种软件性能千差万别,互不兼容,学习使用困难,直接影响到设计环节的衔接。同时,20 世纪 90 年代中期以后,微电子技术以惊人的速度发展,在单芯片上可以集成数百万甚至数千万只晶体管,工作速度达到 GHz 以上,在这种芯片上设计系统就需要更加先进的工具来支持。因此,人们开始追求将整个设计过程自动化,即电子系统设计自动化(electronics design automation, EDA)。

目前,EDA 技术主要以硬件描述语言输入、系统级仿真和综合技术为核心,能够自动将用户以硬件语言描述的功能需求转化为基础门电路,将设计封装到 FPGA/CPLD 中或制成 ASIC 芯片,极大地提高系统的设计效率,使设计人员摆脱大量的辅助性和基础性工作,将其精力集中于创造性的方案与算法设计和系统结构优化上,同时大大缩短了设计研发时间。

1.2 EDA 技术的知识体系

1.2.1 EDA 技术的主要内容

EDA 技术主要包含以下 4 方面的内容:可编程逻辑器件、硬件描述语言、软件开发工具和实验开发系统。其中,可编程逻辑器件是利用 EDA 技术进行电子设计系统的载体,即硬件设计对象;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(集成开发环境)是利用 EDA 技术进行电子系统设计的智能化、自动化设计工具;实验开发系统是利用 EDA 技术进行电子系统设计的下载工具及硬件验证工具。

1. 可编程逻辑器件

传统数字系统通常采用具有固定逻辑功能的 74 系列数字电路器件等专用集成电路进行设计,设计人员大量的时间花费在数字器件的选择上。可编程逻辑器件(programmable logic device, PLD)是一种由用户根据自己的要求构造逻辑功能的数字集成电路。PLD 本身在未编程前没有确定的逻辑功能,就如同一张白纸,要由用户利用计算机辅助设计(原理图或硬件描述语言)方法表示设计思想,经过编译和仿真,生成目标文件,再由编程器或下载电缆将设计文件配置到目标器件中,形成该芯片独有的逻辑功能,PLD 就变成能满足用户需求的专用集成电路。其主要特点如下。

(1) 缩短研制周期。对于用户而言,PLD 可像通用器件一样按一定的规格型号在市场买到,其功能的实现完全独立于 IC(integrated circuit)集成芯片厂家,由用户自己设计完成,

不必像传统 IC 那样花费样片制作等待时间。由于采用先进的 EDA 技术, PLD 的设计和编程均十分方便和有效, 整个设计通常只需几天便可完成, 缩短产品的研制周期, 有利于产品的快速上市。

(2) 降低设计成本。采用 PLD 设计不需样片制作费用, 在设计初期或小批量的试制阶段, 其平均单片成本很低。如果要转入大批量生产, 由于已用 PLD 进行了原型验证, 也比直接设计 IC 费用小、成功率高。

(3) 提高设计灵活性。第一, PLD 在设计完成后可立即进行验证, 有利于及早发现设计中的问题, 完善设计; 第二, 大多数 PLD 器件可反复编程, 为设计修改和产品升级带来了方便; 第三, 基于 SRAM(静态随机读写存储器)开关的 FPGA(现场可编程门阵列)具有动态重构特性, 在系统设计中引入了“软硬件”(固件)的全新概念, 使得电子系统具有更好的灵活性和自适应性。

2. 硬件描述语言

硬件描述语言(hardware describe language, HDL)以文本形式来描述数字系统硬件结构和行为, 是一种用形式化方法来描述数字电路和系统的语言, 可以从上层到下层(从抽象的系统级到具体的寄存器级)逐层描述设计者的设计思想。

用硬件描述语言进行电路系统设计是当前 EDA 技术的一个重要特征。与传统的原理图设计方法相比, 硬件描述语言更适合规模日益增大的电子系统, 它还是进行逻辑综合优化的重要工具。硬件描述语言能使设计者在比较抽象的层次上描述设计的结构和内部特征, 其优点是语言的公开性、设计与硬件工艺的无关性、宽范围的描述能力、便于组织大规模系统设计、便于设计的复用和继承。

3. 软件开发工具

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, 在 EDA 技术应用中占据极其重要的地位。EDA 工具大致可以分为 5 个模块, 即设计输入编辑器、HDL 综合器、仿真器、适配器(或布局布线器)和下载器。

每个 FPGA/CPLD 生产厂家为了方便用户, 往往都提供集成开发环境, 基本都可以完成所有的设计输入、仿真、综合、布线和下载等工作。主流厂家的 EDA 集成开发工具有 Altera 的 MAX+PLUS II 和 Quartus II、Lattice 的 ispLEVER 和 Xilinx 的 ISE 设计套件。

本书所使用的 EDA 集成开发环境为 Quartus II, 它支持原理图、VHDL 和 Verilog HDL 文本文件, 以及以波形与 EDIF 等格式的文件作为设计输入, 并支持这些文件的任意混合设计。该软件具有门级仿真器, 可以进行功能仿真和时序仿真, 能够产生精确的设计结果。在适配之后, 可生成供时序仿真用的 EDIF、VHDL 和 Verilog HDL 这 3 种不同格式的网表文件。

4. 实验开发系统

实验开发系统是利用 EDA 技术进行电子系统设计的下载及硬件验证工具。该系统提供芯片下载电路及 EDA 实验/开发的外围资源(类似于用于单片机开发的仿真器), 供硬件验证

用。实验开发系统一般包括：实验或开发所需的各类基本信号发生模块，包括时钟、脉冲、高低电平；FPGA/CPLD 输出信息显示模块，包括数码显示、发光管显示、声响指示等；监控程序模块，提供“电路重构软配置”；目标芯片适配座以及上面的 FPGA/CPLD 目标芯片和编程下载电路。

1.2.2 可编程逻辑器件

一般的集成电路芯片的功能已经设置好，是固定不变的，而可编程逻辑器件(PLD)的优点在于允许用户编程(使用硬件描述语言，如 VHDL)来实现所需要的逻辑功能。用户首先用硬件描述语言来表示所需要实现的逻辑功能，然后经过编译和仿真生成目标文件，再由编程器或下载电缆将设计文件配置到目标器件中，PLD 就变成了能满足用户需求的专用集成电路(ASIC)。PLD 可以被重复编程，用户可以随时通过修改程序来修改器件的逻辑功能，而无须改变硬件电路。

1. FPGA

FPGA 是 field-programmable gate array 的缩写，即现场可编程门阵列，是由美国的 Xilinx 公司率先推出的。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 的编程无须专用的编程器，只需使用通用的 EPROM、PROM 编程器即可。当需要修改 FPGA 功能时，只需要换一片 EPROM 即可。FPGA 能够反复使用。同一片 FPGA，采用不同的编程数据可以产生不同的电路功能。因此，FPGA 的使用非常灵活。

2. CPLD

CPLD 是 complex programmable logic device 的缩写，即复杂可编程逻辑器件。CPLD 也是一种用户根据需要而自行构造逻辑功能的数字集成电路。其基本设计方法是借助集成开发软件平台，用原理图、硬件描述语言等方法，生成相应的目标文件，通过下载电缆(在“系统”编程)将代码直接传送到目标芯片中，从而实现数字系统的设计。

FPGA 和 CPLD 都是 PLD 器件，两者的功能基本相同，只是实现的硬件原理有所区别，所以有时可以忽略两者的区别，统称为可编程逻辑器件或 CPLD/FPGA。

使用 CPLD/FPGA，工程师可以通过传统的原理图输入法，或是硬件描述语言自由地设计一个数字系统。设计完成后，可以通过软件仿真来验证设计的正确性，可以利用 CPLD/FPGA 的在线修改功能随时修改设计而不必改变硬件电路。

1.2.3 可编程逻辑语言

目前，国际上越来越多的 EDA 工具接受 HDL(硬件描述语言)作为设计输入，因此世界各大公司也都相继开发了自己的 HDL。进入 20 世纪 80 年代后期，硬件描述语言向着标准

化、集成化的方向发展,最终,VHDL 和 Verilog HDL 适应了这种趋势的要求,先后成为 IEEE 标准。因此,目前应用最广泛的硬件描述语言有 VHDL 和 Verilog HDL 两种。

1. VHDL

VHDL(very-high-speed integrated circuit hardware description language)涵盖面广,抽象描述能力强,支持硬件的设计、验证、综合与测试。VHDL 能在多个级别上对同一逻辑功能进行描述,如可以在寄存器级别上对电路的组成结构进行描述,也可以在行为描述级别上(这也是 VHDL 的优势)对电路的功能与性能进行描述。VHDL 具有广泛的使用范围,其在工程设计上的优点主要表现为以下几个方面。

(1) 与其他的硬件描述语言相比,VHDL 具有更强的行为描述能力,从而决定了它成为系统设计领域最佳的硬件描述语言。强大的行为描述能力是避开具体的器件结构,从逻辑行为上描述和设计大规模电子系统的重要保证。

(2) VHDL 丰富的仿真语句和库函数,可帮助设计者在任何大系统的设计早期就能查验设计系统的功能可行性,随时可对设计进行多个级别的仿真模拟。

(3) VHDL 语句的行为描述能力和程序结构决定了它具有支持大规模设计的分解和已有设计的再利用功能。

(4) 对于用 VHDL 完成的一个确定的设计,可以利用 EDA 工具进行逻辑综合和优化,并自动地把 VHDL 描述设计转变成门级网表文件。

(5) VHDL 对设计的描述具有相对独立性,设计者可以不懂硬件的结构,也不必管最终设计实现的目标器件是什么,而进行独立的设计。

2. Verilog HDL

Verilog HDL 是专为专用集成电路(application specific intergrated circuits, ASIC)设计而开发的。Verilog HDL 较为适合算法级、寄存器传输级(RTL)、逻辑级和门级的设计,它可以很容易地把完成的设计移植到不同厂家的不同芯片中去,并且设计很容易修改,更适合电子专业技术人员进行数字系统的设计。采用 Verilog HDL 输入法的最大优点是其与工艺的无关性,这使得设计者在功能设计、逻辑验证阶段可不必过多考虑门级及其工艺的具体细节,只要利用系统设计时对芯片的要求,施加不同的约束条件,即可设计出实际电路。

3. Abel HDL

Abel HDL 是一种支持各种不同输入方式的 HDL,被广泛用于各种可编程逻辑器件的逻辑功能设计,由于其语言描述的独立性,因而适用于各种不同规模的可编程器件的设计。Abel HDL 具有 C 语言的风格,易学易用,但是可移植性较差,只能在 Altera 公司的开发系统上使用,这限制了它的使用范围。

4. Superlog 语言

Verilog HDL 的首创者 Phil Moorby 和 Peter Flake 等硬件描述语言专家,在一家叫 Co-Design Automation 的 EDA 公司进行合作,开始对 Verilog HDL 进行扩展研究。1999 年,

Co-Design 公司发布了 Superlog 系统设计语言,同时发布了两个开发工具: SYSTEMSIM 和 SYSTEMEX,一个用于系统级开发,一个用于高级验证。2001 年,Co-Design 公司向电子产业标准化组织 Accellera 提交了 Superlog 扩展综合子集,这样,它就可以在 Verilog HDL 的 RTL 级综合子集的基础上,提供更多级别的硬件综合,为各种系统级的 EDA 软件工具所利用。

5. System C 语言

System C 是由 Synopsys 公司和 CoWare 公司合作开发的。1999 年,40 多家世界著名的 EDA 公司、IP 公司、半导体公司和嵌入式软件公司宣布成立“开放式 System C 联盟”。Cadence 公司也于 2001 年加入了 System C 联盟,使 System C 跨业界标准的进程大大加快。

以上几种 HDL 各有其优点和缺点,其中 VHDL 和 Verilog HDL 的应用更加广泛。读者学好一种计算机语言后,很容易地就能掌握另一种计算机语言。

1.2.4 EDA 开发工具

目前比较流行的、主流厂家的 EDA 集成开发工具有 Altera 的 MAX+PLUS II 和 Quartus II、Xilinx 的 ISE 设计套件、Lattice 的 ispLEVEL。

1. MAX+PLUS II

MAX+PLUS II 是 Altera 公司推出的第三代 PLD 开发系统。对于一般几千门的电路设计,使用该软件,从设计输入到器件编程完毕,用户拿到设计好的逻辑电路,大约只需几小时。特别是在原理图输入方面,其友善的人机界面,特别适合初学者使用。至今一些资深的工程师常常采用 MAX+PLUS II 和 Quartus II 软件结合进行针对不同复杂度的项目开发。但由于其支持的元件种类和门数较少,综合效率较低,Altera 公司已不再对该软件进行更新。

2. Quartus II

Quartus II 是 Altera 公司开发的第四代综合性 PLD 集成开发软件,支持多种设计输入方式,内嵌自有的综合器和仿真器,可以完成从设计输入到硬件配置的完整的 PLD 设计流程。该软件支持 Altera 公司的 IP 核,包含了 LPM/MegaFunction 宏功能模块,使用户可以充分利用成熟的模块,简化设计复杂性、加快设计速度。Quartus II 对第三方 EDA 工具的良好支持也使用户可以在设计流程的各个阶段使用熟悉的第三方 EDA 工具。

Quartus II 通过和 DSP Builder 工具与 Matlab/Simulink 相结合,可以方便地实现各种 DSP 应用系统。它支持 Altera 公司的片上可编程系统(SOPC)开发,集系统级设计、嵌入式软件开发、可编程逻辑设计于一体,是一种综合性的开发平台。

3. ISE Design Suite

ISE 设计套件(ISE Design Suite)是 Xilinx 公司的 EDA 集成开发工具,在为嵌入式、DSP 和逻辑设计人员提供 FPGA 设计工具和 IP 产品方面确立了业界新标准。作为 Xilinx 目标设

计平台战略的一个重要里程碑, ISE 设计套件的逻辑版本支持快速访问和使用从前端直到后端的完整 FPGA 设计流程, 它提供的工具和基础 IP 覆盖设计输入、引脚分配、综合、验证(包括片上调试)、实施、布局/分析、位流生成以及器件编程功能。ISE 设计套件的逻辑版本包括 ISE Foundation 软件、ISE Simulator、PlanAhead 设计分析工具、ChipScope Pro Analyzer、ChipScope Pro Serial I/O Toolkit、Base-level IP。

4. ispLEVEL

Lattice 是 ISP(在线可编程)技术的发明者, 该技术极大地促进了 PLD 产品的发展。ispLEVEL 是一款用于所有 Lattice 可编程逻辑产品的集成开发环境。各种不同的 ispLEVEL 版本都包含一组全方位的功能强大的工具, 包括项目管理、IP 综合、设计规划、布局布线和在系统逻辑分析等。

5. 第三方工具

1) Synplify

Synplify 是由 Synplicity 公司(现被 Synopsys 公司收购)开发设计的逻辑综合工具。它在综合优化方面的优点非常突出, 是应用范围较广的综合优化工具之一。它支持用 Verilog HDL 和 VHDL 描述的系统级设计, 具有强大的行为及综合能力。综合后, 能生成 Verilog HDL 和 VHDL 网表文件, 能进行功能仿真。

2) ModelSim

ModelSim 软件是 Mentor 公司的产品, 该产品是一款强大的仿真软件, 具有速度快、精度高和便于操作的特点, 此外还具有代码分析能力, 可以看出不同代码段消耗资源的情况。其功能侧重于编译和仿真, 没有综合和适配能力。在 ModelSim 中可以进行的仿真有 Simulate Behavioral Model(行为仿真)、Simulate Post-Translate VHDL Model(转换后仿真)、Simulate Post-Map VHDL Model(映射后仿真)和 Simulate Post-Place & Route VHDL Model(布局布线后仿真及时序仿真)。ModelSim 有专门针对 Altera 公司产品的 ModelSim-Altera 版。

3) Synario

Synario 软件是 Lattice 公司和 Data I/O 公司合作开发的一种运行于 PC Windows 环境下的通用电子设计工具软件。该软件继承和发扬了 PLD 器件开发软件 ABEL 的特点。它有一个包括各种常用逻辑器件和模块的较完善的宏库。设计中能进行逻辑图输入和 ABEL 硬件描述语言输入, 并包括功能模拟显示和波形显示。

1.3 EDA 设计流程

1.3.1 设计输入

利用 EDA 技术进行一项工程设计, 首先需利用 EDA 工具的文本编辑器或图形编辑器将设计思路用文本方式或图形方式表达出来, 进行排错编译, 形成 VHDL 文件格式, 为进

一步的逻辑综合作准备。

常用的源程序输入方式有以下三种。

(1) 原理图输入方式: 利用 EDA 工具提供的图形编辑器以原理图的方式进行输入。原理图输入方式比较容易掌握, 直观且方便, 所画的电路原理图(注意, 这种原理图与利用 Protel 画的原理图有本质的区别)与传统的器件连接方式完全一样, 很容易被人接受, 而且编辑器中有许多现成的单元器件可以利用, 也可以根据设计需要自行设计特定元件。然而原理图输入法也有它的缺点: 随着设计规模增大, 设计的易读性迅速下降, 对于图中密密麻麻的电路连线, 极难搞清电路的实际功能; 一旦完成, 电路结构的改变将十分困难, 因而几乎没有可再利用的设计模块; 移植困难、入档困难、交流困难、设计交付困难, 因为不可能存在一个标准化的原理图编辑器。

(2) 状态图输入方式: 以图形的方式表示状态图进行输入。当填好时钟信号名、状态转换条件、状态机类型等要素后, 就可以自动生成 VHDL 程序。这种设计方式简化了状态机的设计。

(3) VHDL 软件程序的文本方式: 最一般化、最具普遍性的输入方法, 任何支持 VHDL 的 EDA 工具都支持文本方式的编辑和编译。

1.3.2 综合

综合过程将 VHDL 的软件设计与硬件的可实现性挂钩, 利用 EDA 软件系统的综合器进行逻辑综合。

综合器的功能就是将设计者在 EDA 平台上完成的针对某个系统项目的 HDL、原理图或状态机的描述, 针对给定硬件结构组件进行编译、优化、转换和综合, 最终获得门级电路甚至更底层的电路描述文件。由此可见, 综合器工作前, 必须给定最后实现的硬件结构参数, 它的功能就是将软件描述与给定硬件结构用某种网表文件的方式联系起来。显然, 综合器是软件描述与硬件实现的一座桥梁。综合过程就是将电路的高级语言描述转换成低级的、可与 FPGA/CPLD 或构成 ASIC 的门阵列基本结构相映射的网表文件。

由于 VHDL 仿真器的功能仿真是面向高层次的系统仿真, 只能对 VHDL 的系统描述做可行性的评估测试, 不针对任何硬件系统, 因此基于这一仿真层次的许多 VHDL 语句不能被综合器所接受。这就是说, 这类语句的描述无法在硬件系统中实现(至少是现阶段), 这时, 综合器不支持的语句在综合过程中将被忽略。综合器对源 VHDL 文件的综合是针对某一 PLD 供应商的产品系列的, 因此, 综合后的结果可以为硬件系统所接受, 具有硬件可实现性。

1.3.3 适配

逻辑综合通过后必须利用适配器将综合后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、布线与操作,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。

适配器的功能是将由综合器产生的网表文件配置于指定的目标器件中,产生最终的下载文件,如 JEDEC 格式的文件。适配所选定的目标器件(FPGA/CPLD 芯片)必须属于综合器指定的目标器件系列。对于一般的可编程模拟器件所对应的 EDA 软件来说,一般仅需包含一个适配器就可以了,如 Lattice 的 PAC-DESIGNER。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 FPGA/CPLD 供应商自己提供,因为适配器的适配对象直接与器件的硬件结构相对应。

注意:综合器和适配器分别是一套软件系统,而不是真实存在的硬件器件。

1.3.4 仿真

在编程下载前必须利用 EDA 仿真工具对适配生成的结果进行模拟测试。仿真就是让计算机基于一定的算法和仿真库对所设计的电路进行模拟,以验证设计的正确性,排除错误。仿真工具可以采用 PLD 公司的 EDA 开发工具,也可以选用第三方的专业仿真工具。

1. 功能仿真

功能仿真是直接对 VHDL、原理图描述或其他描述形式的逻辑功能进行测试模拟,以了解其实现的功能是否满足设计要求,仿真过程不涉及任何具体器件的硬件特性,最显著的特征是仿真信号没有延迟。在进行项目设计时,一般首先进行功能仿真,待确认设计文件所表达的功能满足设计要求后,再进行综合、适配和时序仿真,以便发现设计项目的功能性设计缺陷。

2. 时序仿真

时序仿真是接近真实器件运行特性的仿真,仿真文件中已包含了器件硬件特性参数,因而仿真精度高。但时序仿真的仿真文件必须来自针对具体器件的综合器与适配器。综合后得到的 EDIF 等网表文件通常作为 FPGA/CPLD 适配器的输入文件,产生的仿真网表文件中包含了精确的硬件延迟信息。时序仿真可以发现设计中由于硬件特性而产生的时序错误,结合功能仿真,设计者可以区分哪些是功能错误,哪些是时序错误。

1.3.5 编程下载

如果编译、综合、布线/适配和行为仿真、功能仿真、时序仿真等过程都没有发现问题,即满足设计要求,则可以将由 FPGA/CPLD 布线/适配器产生的配置/下载文件通过编程器或下载电缆载入目标芯片 FPGA 或 CPLD 中。