

新世纪电子信息课程系列规划教材

EDA技术及VHDL

EDA JISHU JI VHDL

主 编 蒋小燕 俞伟钧
张立臣



东南大学出版社
SOUTHEAST UNIVERSITY PRESS

新世纪电子信息课程系列规划教材

EDA 技术及 VHDL

主 编 蒋小燕 俞伟钧 张立臣
副主编 黄文生 徐 俊

东南大学出版社
· 南京 ·

内 容 简 介

本书全面介绍了 EDA 技术及 VHDL 概述和发展历史;讨论了 EDA 技术及 VHDL 与可编程逻辑器件的原理和应用;分析了 VHDL 结构与要素、VHDL 基本语句;介绍了 MAX+PLUS II 开发工具、VHDL 设计应用实例;研究了 EDA 技术及 VHDL 的实验方法,主要包括基本逻辑电路、可编程逻辑器件的应用。

本书编写力求反映应用型本科的要求和理工类专业的教学特点,内容力求由浅入深,循序渐进,通俗易懂,基本概念和基本知识准确清晰,说明简明扼要,注重将理论与实际应用有机地结合起来,并且特别注意以形象直观的形式来配合文字表述,重点突出,以帮助读者掌握计算机控制技术的主要内容。

本书可适应不同层次的读者选用,既可用于高等学校理工类本科教材,也适用于各类工程技术人员参考、阅读。

图书在版编目(CIP)数据

EDA 技术及 VHDL / 蒋小燕,俞伟钧,张立臣主编.
南京:东南大学出版社,2008.12
(新世纪电子信息课程系列规划教材)
ISBN 978-7-5641-1504-3

I. E… II. ①蒋…②俞…③张… III. ①电子电路—
电路设计:计算机辅助设计—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. TN702 TP312

中国版本图书馆 CIP 数据核字(2008)第 195112 号

EDA 技术及 VHDL

出版发行	东南大学出版社
出 版 人	江 汉
社 址	南京市四牌楼 2 号
邮 编	210096
经 销	全国各地新华书店
印 刷	盐城印刷总厂有限责任公司
开 本	787 mm×1092 mm 1/16
印 张	19.25
字 数	480 千字
版 次	2008 年 12 月第 1 版
印 次	2008 年 12 月第 1 次印刷
印 数	1—4000 册
书 号	ISBN 978-7-5641-1504-3/TP·254
定 价	40.00 元

(凡因印装质量问题,请与我社读者服务部联系。电话:025-83792328)

前 言

根据 2003 年 1 月教育部组织召开的“21 世纪中国高等学校应用型本科人才培养体系的创新与实践”课题审定会的有关精神,在原高等学校通用的计算机控制技术的基础上,根据理工类应用型本科专业的特点,编写了 EDA 技术及 VHDL 的教材。

EDA 技术及 VHDL 主要内容包括三部分:第一部分 EDA 技术及 VHDL 和可编程逻辑器件,主要包括 EDA 技术及 VHDL 概述及发展历史;可编程逻辑器件原理和应用;第二部分主要讨论 VHDL 结构与要素、VHDL 基本语句、MAX+PLUS II 开发工具、VHDL 设计应用实例;第三部分主要讨论研究 EDA 技术及 VHDL 的实验方法,主要包括基本数字逻辑电路、可编程逻辑器件的应用。

为了让读者能全面、系统地掌握计算机控制技术系统的知识,达到教育部对应用型本科的要求,根据应用型本科的特点,本书在编写过程中,力求由浅入深,循序渐进,通俗易懂,基本概念和基本知识准确清晰,本书结构条理清晰,重点要素突出。

本书共分 8 章。第 1 章介绍 EDA 技术及 VHDL 现状与发展(绪论);第 2 章主要介绍可编程逻辑器件结构、原理和应用技术;第 3 章主要研究讨论 VHDL 结构与要素;第 4 章主要研究讨论 VHDL 基本语句,第 5 章主要介绍 MAX+PLUS II 开发工具;第 6 章主要研究讨论基本逻辑电路设计方法及应用;第 7 章主要介绍 VHDL 设计与应用实例;第 8 章主要研究讨论 EDA 技术及 VHDL 实验原理与实验方法。

本书由蒋小燕、俞伟钧、张立臣任主编,黄文生、徐俊任副主编;其中第 1 章至第 2 章由俞伟钧编写,第 3 章由黄文生编写,第 4 章由张立臣编写,第 5 章至第 7 章由蒋小燕编写,第 8 章和附录由徐俊编写,本书由蒋小燕统稿,由钱显毅主审。

为了方便教师教学和与作者交流,本书作者将向该教材的教学单位提供 PPT 及相关教学资料,联系方式 jiangxiaoyan@czu.cn。

由于作者水平有限,书中难免有错误或不足之处,敬请广大读者批评、指正。

编 者
2008 年 7 月

目 录

1	EDA 技术概述	(1)
1.1	EDA 技术及其发展	(1)
1.1.1	EDA 技术的发展阶段	(1)
1.1.2	EDA 的发展趋势	(2)
1.2	EDA 技术的主要内容	(3)
1.2.1	硬件描述语言	(3)
1.2.2	大规模可编程逻辑器件	(4)
1.2.3	软件开发工具	(4)
1.2.4	实验开发系统	(5)
1.3	EDA 技术的基本工具	(5)
1.3.1	设计输入编辑器	(6)
1.3.2	综合器	(6)
1.3.3	仿真器	(6)
1.3.4	适配器(布局布线器)	(7)
1.3.5	下载器	(7)
1.4	EDA 技术的基本设计思路	(7)
1.4.1	EDA 电路级设计	(7)
1.4.2	EDA 系统级设计	(8)
1.5	EDA 设计开发流程	(9)
1.5.1	设计输入	(10)
1.5.2	综合	(10)
1.5.3	适配	(11)
1.5.4	时序仿真与功能仿真	(11)
1.5.5	编程下载	(12)
1.5.6	硬件测试	(12)
	习题 1	(12)
2	可编程逻辑器件	(13)
2.1	可编程逻辑器件概述	(13)
2.1.1	PLD 发展历程	(13)
2.1.2	可编程逻辑器件的特点	(13)
2.1.3	可编程逻辑器件的分类	(14)
2.2	简单 PLD 原理	(15)

2.2.1	PLD 中阵列的表示方法	(15)
2.2.2	PROM	(16)
2.2.3	PLA 器件	(16)
2.2.4	PAL	(17)
2.2.5	GAL 器件	(18)
2.3	CPLD 的结构与工作原理	(18)
2.3.1	CPLD 的基本结构	(19)
2.3.2	Altera 公司 MAX 系列 CPLD 简介	(19)
2.4	FPGA 结构与工作原理	(23)
2.4.1	FPGA 的基本结构	(23)
2.4.2	Xilinx 公司 XC3000 系列的结构	(24)
2.5	常用 FPGA/CPLD 产品简介	(26)
2.5.1	ALTERA 公司的 FPGA/CPLD 器件系列	(26)
2.5.2	Lattice 公司 CPLD 器件系列	(27)
2.5.3	Xilinx 公司的 FPGA/CPLD 器件系列	(29)
2.6	FPGA/CPLD 的测试技术	(30)
2.6.1	内部逻辑测试	(30)
2.6.2	JTAG 边界测试技术	(31)
2.7	FPGA 和 CPLD 的比较	(33)
	习题 2	(33)
3	VHDL 结构与要素	(34)
3.1	实体	(34)
3.1.1	实体说明	(34)
3.1.2	类属(GENERIC)说明语句	(34)
3.1.3	端口说明	(35)
3.2	结构体	(36)
3.3	库、程序包及配置	(38)
3.3.1	库	(38)
3.3.2	程序包	(40)
3.3.3	配置	(42)
3.4	VHDL 文字规则	(42)
3.4.1	数字型文字	(42)
3.4.2	字符串型文字	(43)
3.4.3	标识符	(43)
3.4.4	下标名及下标段名	(45)
3.5	VHDL 数据对象	(45)
3.5.1	常量(CONSTANT)	(45)
3.5.2	变量(VARIABLE)	(46)
3.5.3	信号(SIGNAL)	(46)

3.6 VHDL 数据类型	(47)
3.6.1 VHDL 的预定义数据类型	(47)
3.6.2 IEEE 预定义标准逻辑位与矢量	(49)
3.6.3 其他预定义标准数据类型	(50)
3.6.4 用户自定义数据类型方式	(51)
3.6.5 类型转换	(53)
3.7 VHDL 操作符	(55)
3.7.1 逻辑操作符	(56)
3.7.2 关系操作符	(57)
3.7.3 算数操作符	(58)
3.7.4 并置操作符	(59)
3.7.5 省略赋值操作符	(59)
习题 3	(60)
4 VHDL 基本语句	(61)
4.1 VHDL 顺序语句	(61)
4.1.1 赋值语句	(61)
4.1.2 转向控制语句	(61)
4.1.3 WAIT 语句	(70)
4.1.4 子程序调用语句	(72)
4.1.5 返回语句(RETURN)	(78)
4.1.6 断言语句(Assert)	(78)
4.1.7 REPORT 语句	(79)
4.1.8 NULL 语句	(79)
4.2 VHDL 并行语句	(81)
4.2.1 并行信号赋值语句	(82)
4.2.2 进程语句	(85)
4.2.3 块语句(BLOCK)	(93)
4.2.4 并行过程调用语句	(95)
4.2.5 元件例化语句	(97)
4.2.6 生成语句	(99)
4.3 属性描述与定义语句	(102)
习题 4	(106)
5 MAX+plus II 开发工具	(107)
5.1 Max+plus II 开发系统的特点	(107)
5.2 VHDL 文本输入设计方法初步	(108)
5.2.1 文本输入	(108)
5.2.2 系统的编译、综合、适配	(109)
5.2.3 时序仿真	(112)

5.2.4	引脚锁定	(117)
5.2.5	编程下载	(119)
5.3	原理图输入设计方法	(120)
5.3.1	原理图编辑工具	(120)
5.3.2	原理图方式设计步骤	(121)
5.3.3	设计流程归纳	(125)
5.3.4	较复杂电路的原理图设计	(125)
5.4	参数可设置 LPM 宏功能块应用	(129)
5.4.1	基于 LPM 数控分频器设计	(130)
5.4.2	编辑定制 LPM_FF 模块	(132)
5.4.3	编辑定制 LPM_FIFO 模块	(134)
5.4.4	基于 LPM_ROM 的 4 位乘法器设计	(138)
5.5	波形输入设计方法	(142)
5.6	VHDL 与原理图混合设计方式	(144)
习题 5		(146)
6	基本逻辑电路设计	(148)
6.1	组合逻辑电路设计	(148)
6.1.1	门电路	(148)
6.1.2	编码器	(150)
6.1.3	译码器	(154)
6.1.4	选择器	(155)
6.1.5	比较器	(157)
6.1.6	加法器	(158)
6.1.7	三态门及总线缓冲器	(160)
6.2	时序逻辑电路设计	(164)
6.2.1	时钟信号和复位信号	(164)
6.2.2	触发器	(165)
6.2.3	寄存器	(169)
6.2.4	计数器	(173)
6.3	有限状态机(FSM)	(174)
6.3.1	状态机的功能和基本结构	(175)
6.3.2	Moore 型有限状态机的设计	(176)
6.3.3	Mealy 型有限状态机的设计	(183)
习题 6		(187)
7	VHDL 设计应用实例	(189)
7.1	移位相加 8 位硬件乘法器电路设计	(189)
7.1.1	设计原理	(189)
7.1.2	硬件乘法器的设计	(190)

7.1.3 系统仿真·····	(195)
7.2 数字频率计的设计 ·····	(195)
7.2.1 设计原理·····	(195)
7.2.2 数字频率计的设计 ·····	(195)
7.2.3 系统仿真·····	(200)
7.3 电梯控制系统的设计 ·····	(200)
7.3.1 设计要求·····	(200)
7.3.2 设计思路·····	(201)
7.3.3 设计实现·····	(201)
7.3.4 系统仿真·····	(207)
7.4 多功能信号发生器的设计 ·····	(208)
7.4.1 设计思路·····	(208)
7.4.2 多功能信号发生器的设计实现 ·····	(208)
7.4.3 系统仿真·····	(213)
7.5 数字闹钟系统设计 ·····	(215)
7.5.1 闹钟系统的设计要求及设计思路 ·····	(215)
7.5.2 闹钟系统的总体设计思路·····	(216)
7.5.3 闹钟系统的控制器的设计·····	(216)
7.5.4 闹钟系统的预置寄存器的设计 ·····	(222)
7.5.5 闹钟系统的寄存器的设计·····	(223)
7.5.6 闹钟系统的分频器的设计·····	(224)
7.5.7 闹钟系统的时间计数器的设计 ·····	(225)
7.5.8 闹钟系统的显示驱动器的设计 ·····	(227)
7.5.9 闹钟系统的整体组装 ·····	(230)
7.6 基于 DDS 的数字移相正弦信号发生器设计 ·····	(230)
7.6.1 系统设计要求 ·····	(230)
7.6.2 系统设计方案 ·····	(230)
7.6.3 设计实现·····	(231)
7.7 乒乓球游戏电路设计 ·····	(236)
7.7.1 设计思路·····	(236)
7.7.2 各模块的设计 ·····	(237)
7.8 乐曲硬件演奏电路设计 ·····	(244)
7.8.1 设计要求·····	(244)
7.8.2 设计原理·····	(244)
7.8.3 乐曲硬件演奏电路的设计·····	(245)
7.8.4 乐曲硬件演奏电路的顶层设计 ·····	(252)
7.8.5 系统仿真·····	(253)
习题 7 ·····	(253)

8 实 验	(254)
实验一 原理图输入设计 8 位全加器	(254)
实验二 含异步清零和同步时钟使能的 4 位加法计数器	(254)
实验三 七段数码显示译码器设计	(256)
实验四 数控分频器的设计	(257)
实验五 4 位十进制频率计设计	(259)
实验六 用状态机实现序列检测器的设计	(262)
实验七 用状态机对 ADC0809 采样控制电路的实现	(264)
实验八 循环冗余校验(CRC)模块设计	(266)
附录 GW48 实验电路结构图	(269)
部分习题参考答案	(280)
参考文献	(298)

1 EDA 技术概述

电子设计自动化(Electronic Design Automation, EDA)技术是一种以计算机为工作平台,以 EDA 软件工具为开发环境,以硬件描述语言和电路图描述为设计入口,以可编程逻辑器件为实验载体,以 ASIC(Application Specific Integrated Circuits)、SOC(System On Chip)和 SOPC(System On Programmable Chip)嵌入式系统为设计目标,以数字系统设计为应用方向的电子产品自动化设计技术。它是融合了电子技术、计算机技术、信息处理技术、智能化技术等最新成果而开发的高新技术,是现代电子系统设计、制造不可缺少的技术。EDA 技术涉及面广,包含描述语言、软件、硬件等多方面知识。

1.1 EDA 技术及其发展

EDA 技术伴随着计算机、集成电路、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, CAE)和电子设计自动化(Electronic Design Automation, EDA)三个发展阶段。

1.1.1 EDA 技术的发展阶段

1) CAD 阶段

20 世纪 70 年代的电子线路的计算机辅助设计是 EDA 发展的初级阶段。它利用计算机的图形编辑、分析和存储等能力,协助工程师进行电子系统的 IC(Integrated Circuit)版图编辑和 PCB(Printed Circuit Board)布局布线,取代了手工操作。它可以减少设计人员的繁琐重复劳动,但自动化程度低,需要人工干预整个设计过程。这类专用软件大多以微机为工作平台,易于学习和使用,设计中小规模电子系统可靠有效。现仍有很多这类专用软件被广泛应用于工程设计,由此产生了计算机辅助设计的概念。

2) CAE 阶段

伴随计算机和集成电路的发展,EDA 技术进入到计算机辅助工程设计阶段。20 世纪 80 年代初推出的 EDA 工具以逻辑模拟、定时分析、故障仿真、自动布局和布线为核心,重点解决电路设计没有完成之前的功能检测等问题。利用这些工具,设计师能在产品制作之前预知产品的功能与性能,生成产品制造文件,使设计阶段对产品性能的分析前进了一大步。

CAE 与 CAD 相比,除了纯粹的图形绘制功能外,还具备了自动设计的功能。其主要特征是既具备了自动布局布线和电路的计算机仿真、分析和验证功能;又增加了电路功能设计和结构设计,并且通过电气连接网络表将两者结合在一起,以实现工程设计。其作用已不仅仅是辅助设计,而且可以代替人进行某种思维。CAE 的主要功能有:原理图输入、逻辑仿真、电路分析、自动布局布线、PCB 后分析。

3) EDA 阶段

20 世纪 90 年代,设计师逐步从使用硬件转向设计硬件。从单个电子产品开发转向系统

级电子产品开发(System on a Chip, 片上系统集成), 因此, EDA 工具是以系统级设计为核心, 包括系统行为级描述与结构综合、系统仿真与测试验证、系统划分与指标分配、系统决策与文件生成等一整套的电子系统设计自动化工具。

过去传统的电子系统产品的设计方法是采用“自底而上”的顺序。依照这种顺序, 设计者先对系统结构分块, 直接进行电路级的设计。这种设计方式使设计者不能预测下一阶段的问题, 而且每一阶段是否存在问题, 往往在系统整机调试时才确定, 也很难通过局部电路的调整使整个系统达到既定的功能和指标, 不能保证设计一举成功。EDA 技术高级阶段采用一种新的“自顶向下”的设计顺序和“并行工程”的设计方法。依照这种方法, 设计者的精力主要集中在所要设计系统的准确定义上, EDA 系统去完成电子产品的系统级到物理级的设计。该阶段 EDA 技术的基本特征是, 设计人员以计算机为工具, 按照“自顶向下”的设计方法, 对整个系统进行方案设计和功能划分。设计人员再将划分后的各个模块用硬件描述语言等设计描述方法完成系统行为级设计, 再利用先进的开发工具自动完成逻辑编译、化简、分割、综合、优化、布局布线、仿真及特定目标芯片的适配编译和编程下载, 这也称为数字逻辑电路的高层次设计方法。

1.1.2 EDA 的发展趋势

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)成为集成电路设计的发展方向, 这一发展趋势表现在如下几个方面:

(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($0.13\ \mu\text{m}$ 、 $0.18\ \mu\text{m}$ 、 $0.09\ \mu\text{m}$)已经走向成熟, 在一个芯片上完成系统级集成已成为可能。

(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 工具得到迅速的发展, 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。同时, 设计的效率也成了产品能否成功的关键因素, 促使 EDA 工具和 IP 核应用更为广泛。

(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 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的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 SystemC、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(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 技术目标器件各种性能要求的提高, ASIC 和 FPGA(Field Programmable Gate Array)将更大程度相互融合。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(Programmable Logic Device, PLD)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“走到”一起,互相融合,取长补短。

在新一代产品的 PCB 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,或者由于开始的工作中没有条件完成足够的验证测试,稍后也可以根据要求对它编程,有了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,为已经完成设计并投入应用的集成电路留有多些可自由更改的功能是十分有价值的事,这在通信领域中的芯片设计方面尤为重要。

现在,传统 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 和集成电路设计工业都朝这个方向发展,这并非只是 FPGA 与 ASIC 制造商竞争的产物,对于用户来说,也意味着有了更多的选择。

1.2 EDA 技术的主要内容

EDA 技术主要包括四个方面:① 硬件描述语言;② 大规模可编程逻辑器件;③ 软件开发工具;④ 实验开发系统。其中,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,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,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,实验开发系统则是利用 EDA 技术进行电子系统设计的下载工具及硬件验证工具。

1.2.1 硬件描述语言

硬件描述语言(HDL)是各种描述方法中最能体现 EDA 优越性的描述方法。所谓硬件描述语言,实际就是一个描述工具,其描述的对象就是待设计系统的逻辑功能,实现该功能的算法,选用的电路结构以及其他各种约束条件等,通常要求 HDL 既能描述系统的行为,又能描述系统的结构。

HDL 的使用与普通的高级语言相似,编制的 HDL 程序也需要首先经过编译器进行语法,语义的检查,并转换为某种中间数据格式,但与其他高级语言相区别的是,用硬件描述语言编制程序的最终目的是要生成实际的硬件,因此 HDL 中有与硬件实际情况相对应的并行处理语句。此外,用 HDL 编制程序时,还需注意硬件资源的消耗问题(如门,触发器,连线等的数目),有的 HDL 程序虽然语法,语义上完全正确,但并不能生成与之相对应的实际硬件,其原因就是要实现这些程序所描述的逻辑功能,消耗的硬件资源将十分巨大。目前主要有以下两种 HDL 语言:

(1) Verilog - HDL

Verilog - HDL 语言是在 1983 年由 GDA(Gateway Design Automation)公司首创的,主要用于数字系统的设计。设计者可以用它来进行各种级别的逻辑设计,可以用它进行数字逻辑系统的仿真验证,时序分析,逻辑综合等。它是目前应用最广泛的硬件描述语言之一。其最大优点是工艺无关性,这使得工程师在功能设计,逻辑验证阶段可以不必过多考虑门级电路及其工艺实现的具体细节,只需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。实际上,这是利用 EDA 工具,把逻辑验证与具体工具库匹配,把

布线及延时计算由计算机自动完成,从而减轻了设计者的劳动。

Verilog - HDL 把数字系统当作一组模块来描述,每一个模块具有模块接口以及关于模块内容的描述,一个模块代表一个逻辑单元,这些模块网络相互连接,相互通信。由于 Verilog - HDL 是标准化的,所以能把完成的设计移植到不同厂家的不同芯片中去。又由于 Verilog - HDL 设计的信号位数很容易改变,所以可以通过对信号位数的修改,来适应不同的硬件规模,而且在仿真验证时,仿真测试用例可以用同一种描述语言来完成。

(2) VHDL

VHDL 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对电路的功能与性能进行描述。无论哪种级别的描述,都可以利用综合工具将描述转化为具体的硬件结构。VHDL 的基本结构包含有一个实体和一个结构体,而完整的 VHDL 结构还包括配置,程序包与库,各种硬件描述语言中,VHDL 的抽象描述能力最强,因此运用 VHDL 进行复杂电路设计时,往往采用自顶向下结构化的设计方法。VHDL 还具有以下优点:

① VHDL 的宽范围描述能力使它成为高层次设计的核心,将设计人员的工作重心提高到了系统功能的实现与调试,只需花较少的精力用于物理实现。

② VHDL 可以用简洁明确的代码描述来进行复杂控制逻辑的设计,灵活且方便,而且也便于设计结果的交流、保存和重用。

③ VHDL 的设计不依赖于特定的器件,方便了工艺的转换。VHDL 是一个标准语言,为众多的 EDA 厂商支持,因此移植性好。

比较而言,VHDL 语言是一种高级描述语言,适用于电路高级建模,综合的效率和效果较好。Verilog - HDL 语言是一种低级的描述语言,适用于描述门级电路,容易控制电路资源,但其对系统的描述能力不如 VHDL 语言。

1.2.2 大规模可编程逻辑器件

可编程逻辑器件(PLD)是一种由用户编程以实现某种逻辑功能的新型逻辑器件。FPGA(Field Programmable Gate Array)和 CPLD(Complex Programmable Logic Devices)分别是现场可编程门阵列和复杂可编程逻辑器件的简称。高集成度、高速度和高可靠性是 FPGA/CPLD 最明显的特点,其时钟延时可小至 ns 级。结合其并行工作方式,在超高速应用领域和实时测控方面有着非常广阔的应用前景。在高可靠应用领域,如果设计得当,将不会存在类似于 MCU 的复位不可靠和 PC 可能跑飞等问题。FPGA/CPLD 的高可靠性还表现在几乎可将整个系统下载于同一芯片中,实现所谓片上系统,从而大大缩小了体积,易于管理和屏蔽。

与 ASIC 设计相比,FPGA/CPLD 显著的优势是开发周期短、投资风险小、产品上市速度快、市场适应能力强和硬件升级回旋余地大,而且当产品定型和产量扩大后,可将在生产达到充分检验的 VHDL 设计迅速实现 ASIC 投产。

1.2.3 软件开发工具

目前比较流行的、主流厂家的 EDA 的软件工具有 Altera 公司的 MAX+plus II、Quartus

II, Lattice 公司的 ispEXPERT, Xilinx 公司的 Foundation Series、ISE/ISE - WebPACK Series。这些软件的基本功能相同,主要差别在于:① 面向的目标器件不一样;② 性能各有优劣。下面简单介绍 MAX+plus II 和 Quartus II。

(1) MAX+plus II:是 Altera 公司推出的一个使用非常广泛的 EDA 软件工具,它支持原理图、VHDL 和 Verilog 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,并支持这些文件的任意混合设计。它具有门级仿真器,可以进行功能仿真和时序仿真,能够产生精确的仿真结果。在适配之后,MAX+plus II 生成供时序仿真用的 EDIF、VHDL 和 Verilog 这三种不同格式的网表文件。它界面友好,使用便捷,被誉为业界最易学易用的 EDA 的软件,并支持主流的第三方 EDA 工具,支持除 APEX20K 系列之外的所有 Altera 公司的 FPGA/CPLD 大规模逻辑器件。

(2) Quartus II:是 Altera 公司的新近推出的 EDA 软件工具,其设计工具完全支持 VHDL、Verilog 的设计流程,其内部嵌有 VHDL、Verilog 逻辑综合器。第三方的综合工具,如 Leonardo Spectrum、Synplify Pro、FPGA Compiler II 有着更好的综合效果,因此通常建议使用这些工具来完成 VHDL/Verilog 源程序的综合。Quartus II 可以直接调用这些第三方工具。同样,Quartus II 具备仿真功能,但也支持第三方的仿真工具,如 Modelsim。此外,Quartus II 为 Altera DSP 开发包进行系统模型设计提供了集成综合环境,它与 MATLAB 和 DSP Builder 结合可以进行基于 FPGA 的 DSP 系统开发,是 DSP 硬件系统实现的关键 EDA 工具。Quartus II 还可与 SOPC Builder 结合,实现 SOPC 系统开发。

1.2.4 实验开发系统

实验开发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),以供硬件验证用。一般包括:① 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;② 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;③ 监控程序模块,提供“电路重构软配置”;④ 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。

1.3 EDA 技术的基本工具

EDA 工具在 EDA 技术应用中占据极其重要的位置,EDA 的核心是利用计算机完成电子设计全程自动化。因此,基于计算机环境的 EDA 软件的支持是必不可少的。

由于 EDA 的整个流程涉及不同技术环节,每一环节中必须有对应的软件包或专用 EDA 工具独立处理,包括对电路模型的功能模拟,对 VHDL 行为描述的逻辑综合等,因此单个 EDA 工具只涉及 EDA 流程中的某一步骤。EDA 工具大致可以分为如下五个模块:

- (1) 设计输入编辑器;
- (2) HDL 综合器;
- (3) 仿真器;
- (4) 适配器;
- (5) 下载器。

1.3.1 设计输入编辑器

通常专业的 EDA 工具供应商或各可编程逻辑器件厂商都提供 EDA 开发工具,在这些 EDA 开发工具中都含有设计输入编辑器,如 Xilinx 公司的 Foundation、Altera 公司的 QuartusII 和 MAX+plusII 等。

一般的设计输入编辑器都支持图形输入和 HDL 文本输入。图形输入通常包括原理图输入、状态图输入和波形图输入三种常用方法。图形输入方式与 PROTEL 作图相似,设计过程形象直观,而且不需要掌握硬件描述语言,便于初学。但图形输入方式存在没有标准化、图形文件兼容性差、不便于电路模块的移植和再利用等缺点。HDL 文本输入方式与传统的计算机软件语言编辑输入基本一致,HDL 文本输入方式比图形输入方式简单得多,用普通的文本编辑器即可完成。

当然,在用 EDA 技术设计电路时,也可以利用图形输入与 HDL 文本输入方式各自的优势,将它们结合起来,实现一个复杂的电路系统的设计。

1.3.2 综合器

HDL 综合器是一种用 EDA 技术实施电路设计中完成电路化简、算法优化、硬件结构细化的计算机软件,是将硬件描述语言转化为硬件电路的重要工具。HDL 综合器在把可综合的 HDL(Verilog 或 VHDL)转化为硬件电路时,一般要经过两个步骤:

第一步是 HDL 综合器对 Verilog 或 VHDL 进行处理分析,并将其转换成电路结构或模块,这时是不考虑实际器件实现的,即完全与硬件无关,这个过程是一个通用电路原理图形成的过程。

第二步是对应实际实现目标器件的结构进行优化,并使之满足各种约束条件,优化关键路径等。

HDL 综合器的输出文件一般是网表文件,是一种用于电路设计数据交换和交流的工业标准化格式的文件,或是直接用硬件描述语言 HDL 表达的标准格式的网表文件,或是对应 FPGA/CPLD 器件厂商的网表文件。

HDL 综合器是 EDA 设计流程中的一个独立的设计步骤,它往往被其他 EDA 环节调用,完成整个设计流程。它的调用方式一般有两种:一种是前台模式,在被调用时,显示的是最常见的窗口界面;另一种是后台模式(也称为控制台模式),被调用时不出现图形界面,仅在后台运行。

1.3.3 仿真器

在 EDA 技术中仿真的地位非常重要,行为模型的表达、电子系统的建模、逻辑电路的验证以及门级系统的测试,每一步都离不开仿真器的模拟检测。在 EDA 发展的初期,快速地进行电路逻辑仿真是当时的核心问题,即使在现在,各个环节的仿真仍然是整个 EDA 设计流程中最重要、最耗时的一个步骤。因此,仿真器的仿真速度、仿真的准确性和易用性成为衡量仿真器的重要指标。

按仿真器对硬件描述语言不同的处理方式,可以分为编译型仿真和解释型仿真器。编译型仿真器速度较快,但需要预处理,因此不能及时修改;解释型仿真器的速度一般,但可以

随时修改仿真环境和条件。

按处理的硬件描述语言类型分,可以分为:

- (1) VHDL 仿真器;
- (2) Verilog 仿真器;
- (3) Mixed HDL 仿真器(混合 HDL 仿真器,同时处理 Verilog 和 VHDL);
- (4) 其他 HDL 仿真器(针对其他 HDL 语言的仿真)。

按仿真电路描述级别的不同,HDL 仿真器可以单独或综合完成以下各仿真步骤:

- (1) 系统级仿真;
- (2) 行为级仿真;
- (3) RTL 级仿真;
- (4) 门级时序仿真。

按仿真是否考虑硬件延时分类,可以分为功能仿真和时序仿真,根据输入仿真文件的不同,可以由不同的仿真器完成,也可由同一个仿真器完成。

1.3.4 适配器(布局布线器)

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)必须属于原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结构相对应。

适配器最后输出的是各厂家自己定义的下载文件,用于下载到器件中以实现设计。适配器输出如下多种用途的文件:

- (1) 时序仿真文件,如 MAX+plusII 的 SCF 文件;
- (2) 适配技术报告文件;
- (3) 面向第三方 EDA 工具的输出文件,如 EDIF、VHDL 或 Verilog 格式的文件;
- (4) FPGA/CPLD 编程下载文件,如用于 CPLD 编辑的 JEDEC、POF、ISP 等格式的文件;用于 FPGA 配置的 SOF、JAM、BIT 等格式的文件。

1.3.5 下载器

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,即满足原设计的要求,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。

1.4 EDA 技术的基本设计思路

1.4.1 EDA 电路级设计

电路级设计工作流程如图 1.1 所示,电子工程师接受系统设计任务后,首先确定设计方