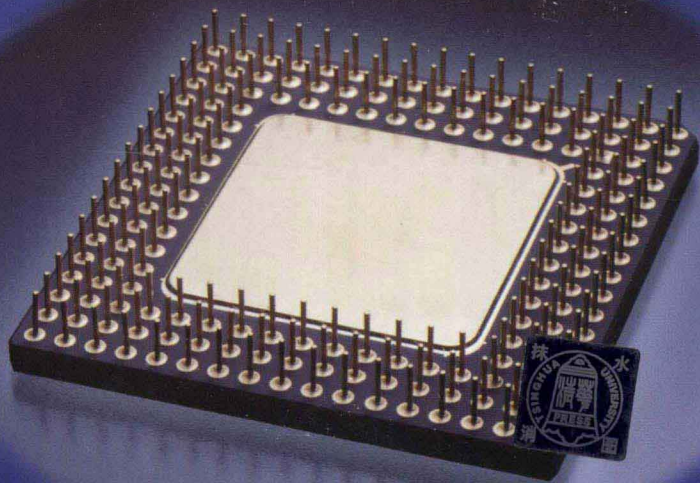


- 以模块化形式进行编写
- 软硬结合，波形仿真
- 原理图与VHDL语言输入设计并存
- 淡化理论，侧重实际设计
- 免费提供计算机主机系统源程序

基于EDA技术的单周期 CPU设计与实现

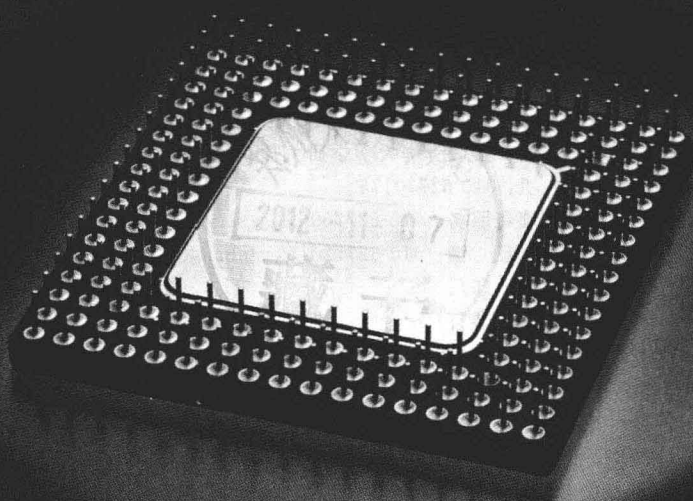
——计算机组成原理实践

蒋丽华◎编著



基于EDA技术的单周期 CPU设计与实现 ——计算机组成原理实践

蒋丽华◎编著



清华大学出版社
北 京

内 容 简 介

本书借鉴国内外最新的计算机组成实践的教学方法,结合作者多年指导计算机组成原理实验课的教学经验,并引用目前市面上主流的 MIPS 指令集,翔实、具体地讲解了计算机各个组成部件的设计原理和典型指令的执行过程。

全书首先介绍了 EDA 技术、Quartus II 软件以及 VHDL 的使用方法;然后讲解了 MIPS 体系结构中比较典型的指令,并使用 Quartus II 软件,通过原理图以及 VHDL 语言,设计出计算机组成的各个部件;最终形成了一个能执行 MIPS 指令集指令程序的计算机主机系统。书中的所有部件及 CPU 逻辑电路都能进行正确的功能仿真模拟及 FPGA 下载验证。

本书既可作为高等院校计算机相关专业的计算机组成原理课程的实验及课程设计教材,也可供计算机硬件开发人员、电子技术及工程技术人员参考使用。

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

基于 EDA 技术的单周期 CPU 设计与实现——计算机组成原理实践/蒋丽华编著. —北京:清华大学出版社,2012.6

ISBN 978-7-302-29129-9

I. ①基… II. ①蒋… III. ①电子电路-电路设计-计算机辅助设计 ②计算机组成原理 IV. ①TN702 ②TP301

中国版本图书馆 CIP 数据核字(2012)第 132754 号

责任编辑:朱英彪

封面设计:张秀岩

版式设计:文森时代

责任校对:张兴旺

责任印制:沈 露

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

印 刷 者:北京密云胶印厂

装 订 者:三河市溧源装订厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:11.5 字 数:266 千字

版 次:2012 年 6 月第 1 版 印 次:2012 年 6 月第 1 次印刷

印 数:1~4000

定 价:24.00 元

前 言

在数字化和网络化的信息技术革命浪潮中,电子技术获得了飞速发展,现代电子产品已渗透到了社会的各个领域。采用 EDA (Electronic Design Automation, 电子设计自动化) 平台进行数字电路系统及计算机系统的设计已成为硬件设计和实践的基本方法。在学习计算机组成原理课程的过程中,亲自动手进行硬件系统的设计及实现,对于相关理论的掌握是至关重要的。

EDA 技术的出现,使得电子系统设计工程师能够在一块通用的芯片上通过编写程序的方式来改变或定义芯片的硬件功能,从而设计出具有不同功能的产品。同时,EDA 技术可缩短硬件的开发周期,降低硬件的开发成本,减轻开发者的劳动强度。EDA 技术主要包括大规模可编程逻辑器件、硬件描述语言和软件开发工具等内容。目前,应用最广泛的大规模可编程逻辑器件是现场可编程逻辑门阵列 FPGA (Field Programmable Gate Array),硬件描述语言是 VHDL。

近年来的教学实践表明,在基于 EDA 的平台上进行计算机组成原理实践,不仅可以提高学生的硬件系统设计水平及硬件系统实践效率,打破以往使用特定实验台进行硬件设计的局限性,而且能够激发学生学习和实践的兴趣,以达到进一步提高学生实践能力及创新能力的目的。

本书是在多年的计算机组成原理本科教学实践、教研工作以及国外学习的基础上进行编写的,可作为本科生基于 EDA 技术进行计算机组成原理实践的指导性通用教材,也可以作为相关工程技术人员学习和使用 VHDL 硬件描述语言、Quartus II 进行数字电路系统设计、计算机硬件系统设计的参考书。

本书以计算机组成原理实践为例,使用 VHDL 硬件描述语言,以 Altera 公司的 Quartus II 作为 EDA 软件工具,介绍了利用 EDA 平台进行硬件系统设计的方法。具体介绍了多路选择器、加减器、移位运算器、算术逻辑运算器、寄存器堆、控制器以及存储器等各个部件,最终形成计算机的主机系统。其中以核心 CPU 的逻辑电路来实现市面上较流行的 MIPS 指令集中的 20 条指令,指令的执行要经过取指令、指令译码、指令执行、存储器读写以及结果写回等过程。最后用汇编语言编写了用于 CPU 测试的简单程序,对所设计的 CPU 逻辑电路进行功能仿真模拟,以验证 CPU 逻辑电路的正确性,同时也可下载到 FPGA 芯片中进行验证,读者根据现有的计算机组成原理知识,完全可以独立地设计出一个完整的计算机主机系统,这样一来,读者便不会再认为计算机深不可测,其硬件课程也不再那么乏味枯燥,期间所收获的那份开心和喜悦更是不可代替的。

全书共分为 9 章及附录 A 和附录 B。其中,部件实验的参考学时为 16~20 学时,计算机主机系统的参考学时为 30~40 学时。

本书在内容编排上由浅入深,结合实例边讲边练。根据 FPGA 的学习与开发经验,采

用 VHDL 语言与原理图输入设计并存的方式进行讲解,使读者能够很快掌握 EDA 技术,并运用数字电子技术的相关知识进行 FPGA 系统设计。通过 Quartus II 设计的原理图和 VHDL 程序代码,可进行波形的功能仿真,并可通过分析波形来验证设计思路正确与否。全书淡化理论知识,侧重于实际应用,避免知识的重复,同时也在一定程度上节省了初学者的学习成本。另外,读者只需对本书中的实验项目工程文件和程序源代码稍加修改,便可应用于自己的工作或完成自己的课题。另外,读者可联系 jianglihua@whpu.edu.cn 索取相关电子资源。

EDA 技术发展很快,新技术、新方法、新器件层出不穷,本书在编写时虽然采用的是当时最新的资料,但在读者阅读本书时,可能又出现了更新的方法和器件,所以本书主要是向大家提供有关 EDA 设计方面较为基础的内容,读者可以从 EDA 厂商的网站上获取更新的 FPGA 器件资料,也可以从销售商或可编程逻辑器件的相关网站获取有关信息和技术支持。

本书由蒋丽华编著,刘昌华、张聪、左翠华、夏祥胜、黄金仙、张喻平、姚蒙等相关师生参与了图书的编写工作,完成了书中大量实例的编程输入、仿真、验证以及文字的录入和校对工作,在此对他们表示衷心的感谢。此外,还要感谢我的家人,在编写本书的将近一年时间里,他们一直给我鼓励、支持和照顾。

由于作者水平有限,加之计算机技术飞速发展,新的理念和技术层出不穷,因此本书难免有不足之处,恳请广大读者批评、指正。

蒋丽华
于武汉工业学院

目 录

第 1 章 绪论.....	1
1.1 实践内容.....	2
1.1.1 计算机组成原理实验.....	3
1.1.2 计算机主机系统设计.....	3
1.2 实践环境.....	4
第 2 章 EDA 设计工具.....	5
2.1 建立工程项目.....	5
2.1.1 启动 Quartus II.....	5
2.1.2 使用 Quartus II.....	6
2.1.3 Quartus II 文件后缀及其含义.....	11
2.1.4 原理图设计.....	11
2.2 VHDL 简介.....	14
2.2.1 VHDL 程序的基本结构.....	15
2.2.2 VHDL 客体及词法单元.....	24
2.2.3 VHDL 的基本描述语句.....	31
2.2.4 属性的描述与定义.....	40
2.2.5 VHDL 设计.....	45
2.3 原理图转换.....	46
2.3.1 原理图转换为 VHDL 文件.....	46
2.3.2 VHDL 文件转换为原理图.....	47
2.4 编译与器件封装.....	47
2.4.1 局部工程编译.....	47
2.4.2 封装成器件放入用户库.....	48
2.5 电路仿真.....	49
2.5.1 建立仿真文件.....	49
2.5.2 仿真设置工具.....	50
2.5.3 功能仿真.....	52
2.5.4 时序仿真.....	53
2.6 工程下载验证.....	53
2.6.1 引脚锁定设置.....	53
2.6.2 配置文件下载.....	55

第 3 章	MIPS 指令和 CPU 设计思路	57
3.1	MIPS 寄存器堆	58
3.2	指令格式和主机系统指令	58
3.2.1	指令格式	58
3.2.2	主机系统指令	59
3.3	CPU 设计思路	63
3.3.1	单周期 CPU 逻辑设计	63
3.3.2	R 类型指令	67
3.3.3	I 类型指令	68
3.3.4	J 类型指令	70
第 4 章	多路选择器	72
4.1	1 位 2 选 1 多路选择器	72
4.1.1	1 位 2 选 1 多路选择器原理图设计	72
4.1.2	1 位 2 选 1 多路选择器的 VHDL 设计	73
4.1.3	1 位 2 选 1 多路选择器的仿真验证	73
4.2	5 位 2 选 1 多路选择器	74
4.2.1	5 位 2 选 1 多路选择器的原理图设计	75
4.2.2	5 位 2 选 1 多路选择器的 VHDL 设计	75
4.2.3	5 位 2 选 1 多路选择器的仿真验证	76
4.3	8 位 2 选 1 多路选择器	76
4.3.1	8 位 2 选 1 多路选择器的原理图设计	77
4.3.2	8 位 2 选 1 多路选择器的 VHDL 设计	77
4.3.3	8 位 2 选 1 多路选择器的仿真验证	78
4.4	32 位 2 选 1 多路选择器	79
4.4.1	32 位 2 选 1 多路选择器的原理图设计	79
4.4.2	32 位 2 选 1 多路选择器的 VHDL 设计	80
4.4.3	32 位 2 选 1 多路选择器的仿真验证	80
4.5	32 位 4 选 1 多路选择器	81
4.5.1	32 位 4 选 1 多路选择器的原理图设计	82
4.5.2	32 位 4 选 1 多路选择器的 VHDL 设计	82
4.5.3	32 位 4 选 1 多路选择器的仿真验证	83
4.6	32 位 32 选 1 多路选择器	83
4.6.1	32 位 32 选 1 多路选择器的原理图设计	84
4.6.2	32 位 32 选 1 多路选择器的 VHDL 设计	84
4.7	lpm_mux 宏模块的设置	87
第 5 章	加减器	90
5.1	1 位加法器	90

5.1.1	1 位加法器的原理图设计	90
5.1.2	1 位加法器的 VHDL 设计	91
5.1.3	1 位加法器的仿真验证	91
5.2	1 位加减器	92
5.2.1	1 位加减器的原理图设计	92
5.2.2	1 位加减器的 VHDL 设计	92
5.2.3	1 位加减器的仿真验证	93
5.3	8 位加法器	93
5.3.1	8 位加法器的原理图设计	94
5.3.2	8 位加法器的 VHDL 设计	94
5.3.3	8 位加法器的仿真验证	95
5.4	32 位加减器	96
5.4.1	32 位加减器的原理图设计	96
5.4.2	32 位加减器的 VHDL 设计	97
5.4.3	32 位加减器的仿真验证	98
第 6 章	移位运算器	100
6.1	移位运算器的原理图设计	100
6.2	移位运算器的 VHDL 设计	101
6.3	移位运算器的仿真验证	103
第 7 章	算术逻辑运算器	105
7.1	0 操作数检测模块	105
7.2	算术逻辑运算器的原理图设计	106
7.3	算术逻辑运算器的 VHDL 设计	107
7.4	算术逻辑运算器的仿真验证	108
第 8 章	寄存器堆	110
8.1	寄存器号译码	110
8.1.1	寄存器号译码的原理图设计	111
8.1.2	寄存器号译码的 VHDL 设计	112
8.1.3	寄存器号译码的仿真验证	113
8.2	8 位触发器	113
8.2.1	8 位触发器的原理图设计	114
8.2.2	8 位触发器的 VHDL 设计	115
8.2.3	8 位触发器的仿真验证	115
8.3	32 位触发器	116
8.3.1	32 位触发器的原理图设计	117
8.3.2	32 位触发器的 VHDL 设计	117

8.3.3	32 位触发器的仿真验证	118
8.4	32 位寄存器	119
8.4.1	32 位寄存器的原理图设计	119
8.4.2	32 位寄存器的 VHDL 设计	121
8.5	32 位寄存器堆	122
8.5.1	32 位寄存器堆的原理图设计	122
8.5.2	32 位寄存器堆的 VHDL 设计	123
8.5.3	32 位寄存器堆的仿真验证	123
第 9 章	计算机主机系统设计	125
9.1	跳转指令寄存器指定元件	125
9.1.1	跳转指令寄存器指定元件的原理图设计	126
9.1.2	跳转指令寄存器指定元件的 VHDL 设计	126
9.1.3	跳转指令寄存器指定元件的仿真验证	126
9.2	指令译码器	127
9.2.1	指令译码器的原理图设计	128
9.2.2	指令译码器的 VHDL 设计	130
9.2.3	指令译码器的仿真验证	131
9.3	控制部件设计	132
9.3.1	控制部件的原理图设计	134
9.3.2	控制部件的 VHDL 设计	135
9.3.3	控制部件的仿真验证	137
9.4	指令存储器	138
9.4.1	指令存储器的原理图设计	138
9.4.2	指令存储器的仿真验证	142
9.5	数据存储器	142
9.5.1	数据存储器的原理图设计	143
9.5.2	数据存储器的仿真验证	147
9.6	单周期中央处理器 CPU 设计	148
9.6.1	单周期中央处理器的原理图设计	148
9.6.2	单周期中央处理器的 VHDL 设计	150
9.7	计算机主机系统设计	152
9.7.1	计算机主机系统的原理图设计	153
9.7.2	计算机主机系统的 VHDL 设计	153
9.8	调试程序编制及主机系统的调试	154
9.8.1	加法调试程序	154
9.8.2	加法程序调试仿真	156
9.8.3	乘法调试程序	158

9.8.4 乘法程序调试仿真	159
附录 A DE2-70 简介及调试注意事项	167
A.1 DE2-70 硬件实验平台简介	167
A.2 DE2-70 引脚分配的一般性指导	168
A.3 实验板基本输入/输出引脚信号	168
附录 B 设计调试过程中的注意事项	169
参考文献	171

第 1 章 绪 论

随着大规模集成电路技术和计算机技术的不断发展,在涉及通信、国防、工业自动化、计算机应用、仪器仪表等领域的电子系统设计工作中,现场可编程(FPGA)的技术含量正以惊人的速度上升。电子类的新技术项目的开发也更多地依赖于FPGA技术的应用,特别是随着VHDL等硬件描述语言综合工具功能和性能的提高,计算机中许多重要的元件,包括CPU都用硬件描述语言来设计和表达,许多CPU(如8051单片机、8086等),硬核嵌入式系统(如ARM、Excalibue系列FPGA),软核嵌入式系统(如Nios),微机CPU,乃至整个计算机系统都用FPGA来实现,即所谓的单片系统SOC(System On a Chip)和SOPC(System On a Programmable Chip)。计算机和CPU的设计技术及其方法进入了一个全新的时代!不仅如此,传统的CPU结构模式,冯·诺依曼结构和哈佛结构也正在受到巨大的挑战。

FPGA芯片以操作灵活著称,可以重复擦写无限次,而微处理器均采用固定电路,只能进行一次设计。设计人员可通过改变FPGA中晶体管的开关状态对电路进行重写,即重配置。因此,尽管FPGA芯片的时钟频率要低于奔腾处理器,但是由于FPGA芯片可并行处理各种不同的运算,所以可完成许多复杂的任务如网页显示,全球天气建模及基因组组合核对等,而且处理速度比奔腾处理器或数字信号处理器快得多。

超级计算机是科技世界中的极品:售价奇高、速度飞快,集成了数以千计的微处理器。但这种超级计算机也浪费了非常多的芯片资源,每个处理器只能进行单任务操作,大部分功能难以充分发挥。现在有了另一种更为简洁的设计:设计工程师采用FPGA芯片来武装超级计算机,取代了原先大量的英特尔奔腾处理器。经过使用硬件描述语言和相关软件语言的设计,FPGA芯片可并行处理多项任务,从而使所有电路都能随时发挥作用。又因为FPGA芯片可以反复编程培植,所以几乎可瞬时完成。例如,通过利用FPGA的重配置功能,在某一时刻它可以用来预报全球天气状况,而下一时刻又可根据某公司做的主要利率对冲情况来评估债券市场的风险,或是转而去图像信息处理。

数字信息技术高速发展的IT时代,全新的计算机设计技术和实现技术向传统的计算机组成原理的教学内容和实验方式提出尖锐的挑战,对计算机硬件系统的学习、设计也提出了愈来愈高的要求。为了培养具有坚实的理论基础、具有创新精神和实践能力的计算机设计人才,必须重视计算机专业课程的教学和实践过程。在学习计算机组成原理及其相关硬件系统课程的过程中,利用EDA(Electronic Design Automation,电子设计自动化)平台的实验环境,使用中大规模集成电路等器件对计算机硬件系统的组成进行逻辑设计、实现及正确性测试,不仅可以加深对计算机硬件系统的组成、工作原理等概念的理解,而且还可以激发学生的学习兴趣,使设计者不受特定硬件平台的限制而设计出水平各异、具有创新性的计算机硬件系统,以达到进一步提高设计能力、动手能力及创新能力的目的。

在学习计算机组成原理等相关课程的过程中,为了加深对计算机组成及其工作原理等

概念的理解,需要进行与其相关的课程实践,这样才能够把学到的理论知识应用到实践中,做到理论与实践相结合。只有把学到的理论知识进行综合、融会贯通,才可以深刻理解计算机的组成原理以及工作原理。在此基础上,通过亲自设计、模拟仿真实现一台计算机主机,对于初学者而言,计算机不再具有神秘感,计算机的组成不再不透明,计算机的工作过程不再是“黑匣子”。

在以往学习计算机组成原理等相关课程的过程中,进行计算机的部件级设计实验以及计算机主机系统设计实验时,一般采用“固定功能集成电路+连线”的验证性教学模式,在特定硬件实验箱中进行实验。在实验中,经常存在着插线虚接、连线折断、器件不断损坏以及每个人的实验结果不能长久保留等问题。对于设计正确的逻辑线路,常常也会因为导线老化、虚接或器件损坏等问题而花费很长时间去寻找问题所在,使实验者不能把主要精力放在系统的功能设计上。

随着 EDA 技术的不断发展,对于复杂的数字电路系统设计和计算机硬件系统的设计可以采用 EDA 平台来实现,使用大规模 FPGA、EDA 软件工具和 IEEE 标准硬件描述语言构建的计算机组成原理实验系统取代传统的计算机组成原理实验系统已经势在必行。这样不仅可以更好地进行计算机硬件系统的逻辑功能设计,而且可以提高硬件设计的水平及硬件实践的效率。

利用 EDA 平台进行计算机系统的部件及主机系统设计,其实质是利用运行在计算机上的软件所提供的虚拟硬件环境,在该环境中进行计算机硬件系统的设计及系统测试。在进行计算机硬件系统的设计、实现过程中,如同用真实器件做实验一样,每个人可以根据需要选择相应的器件及芯片,按照自己设计出的逻辑电路在所选择的器件、芯片及输入/输出引脚之间进行连线;线路一旦连接好,不必再担心线路折断、导线虚接等问题,并且随时可以保存自己未做完的实验,待下次做实验时继续完成设计、布线及存储操作。在此基础上,可以针对设计的部件以及计算机主机系统进行编译、模拟仿真测试,以验证逻辑设计的正确性。另外,可以对设计出的部件进行封装,封装后的元件可供后续的设计使用。在实验结束后,可以根据逻辑设计图及存储的测试结果提交给检查者,以进行设计正确性的验证。

本书作为计算机专业基础课程之一的计算机组成原理实践的指导教材,重点介绍利用 EDA 技术,使用原理图和 VHDL 硬件描述语言两种方式,进行计算机各部件设计,最后完成一台基于 MIPS 指令集主机系统的逻辑设计、实现方法以及设计指令正确性验证的全过程。

1.1 实 践 内 容

在计算机组成原理的学习过程中,结合课程理论知识的学习,需要进行相应的以计算机功能部件为基础的课程实验;在完成计算机组成原理的学习后,为了加深对计算机硬件系统工作原理的理解,增强设计计算机硬件系统的实践能力,需要继续计算机主机系统的设计实践。因此,在计算机组成原理实践中引入 EDA 计算机辅助设计技术是一种非常有益的尝试。

1.1.1 计算机组成原理实验

“计算机组成原理”是计算机与信息工程系的一门核心专业基础课程,它从层次结构的观点和信息输入、处理和输出的顺序讲述计算机的结构及工作原理,使学生掌握计算机常用的逻辑器件、部件的原理、参数及使用方法,学习计算机设计中的入门性知识,以及简单、完备的单台计算机的基本组成原理,培养学生掌握硬件系统的分析、设计、开发、使用和维护能力。

开设计算机组成原理实验,能够使学生在已有计算机知识的基础上,对计算机组成有一个较全面、系统的了解,从而提高学生的计算机硬件基本知识、基本理论和实际操作的能力,提高学生对计算机组成的认识、维护和应用技能,并能够进一步掌握计算机各功能部件的设计、实现方法,以增强对计算机组成原理及工作原理的理解,提高进行计算机硬件系统设计的能力。

在本书中,主要介绍了32位多路选择器、32位加减器、移位运算器、算术逻辑运算器以及32位寄存器堆的设计与实现这5个部件级实验。

在上述实验的基础上,根据需要还可以开设难度更高的实验。

1.1.2 计算机主机系统设计

计算机组成原理课程设计比课程教学实验复杂一些,涉及的深度要广些,并具有一定的实用性。其目的是通过课程设计的综合训练,培养学生分析问题和解决问题的能力,使学生掌握整机概念。同时进行计算机系统部件的EDA设计,即借助于EDA工具软件Quartus II完成主机系统逻辑设计,以帮助学生系统地掌握计算机组成原理这门课程的主要内容,为以后进一步学习计算机体系结构打下一个良好的基础。

在计算机组成原理课程设计中要求设计基于MIPS指令集的单周期中央处理器CPU,从而构建计算机主机。它使以往令学生望而生畏的计算机组成原理课程设计变得轻松而有吸引力,并节约了资金,缩短了设计的时间周期;给学生提供了极大的创造空间,激发和培养了学生的创新思维能力;使学生设计的作品质量和难度系数都得到了提高。通过Quartus II软件在计算机上进行计算机主机系统设计,学生可以采用不同的设计方案,选用软件库中的一般器件或者自行用原理图或者VHDL语言设计的专用器件,而这些器件实验室往往又无法提供。通过计算机仿真的课程设计结果可以使学生会从不同角度思考同一个问题,提出不同的解决方案,从而提高学生思考实际问题的能力。计算机主机系统的设计主要包括计算机主机系统硬件结构设计和指令系统设计,并通过运行存储在主存储器中利用MIPS指令系统中的指令所编制的程序,来验证主机系统设计的正确性。

本书重点介绍进行计算机主机系统的设计、实现及调试的方法。通过设计、实现一台计算机主机系统,使大家了解计算机主机系统硬件结构设计与计算机指令系统设计之间的关系,掌握控制计算机运行的核心部件的设计原理,学会设计、调试一台计算机主机系统。以此为基础,可以进一步培养设计、调试计算机硬件系统的能力。

1.2 实践环境

采用 EDA 技术进行计算机组成原理实践，实验的环境是值得提倡和推广的。

本书重点介绍利用 Altera 公司的 Quartus II 设计软件作为计算机组成原理实践平台的实现方法。Quartus II 平台是完全集成化、易学易用的可编程逻辑设计环境，主要用于设计新器件和中大规模 CPLD（复杂可编程逻辑器件）/FPGA（现场可编程逻辑器件）。该设计软件有原理图输入、硬件描述语言等多种设计方式，利用其所提供的标准门电路、芯片等逻辑器件，可以实现数字电路系统和计算机硬件系统从设计输入、编辑、编译、模拟仿真测试、封装到下载的全过程。Quartus II 平台可以保证所设计系统的可靠性、高效性和灵活性，其强大的图形界面和完整的帮助文档，使设计者能够轻松快速地掌握和使用该平台。在进行实验时，只需要在计算机上安装 Quartus II 设计软件即可进行计算机各功能部件及计算机主机系统的设计。

在基于 EDA 技术进行数字电路系统、计算机硬件系统等相关设计实践时，可以根据需求采用原理图输入方式或使用 VHDL（硬件描述语言）进行逻辑设计。采用传统的原理图输入方式进行系统的设计，使设计者更接近于使用真实的物理器件进行设计。这种设计方法设计、实现的计算机硬件系统，比较直观；对于设计输入具有较好的可控性，并且实现的效率较高。采用 IEEE 标准的 VHDL（Very High Speed Integrated Circuit Hardware Description Language）硬件描述语言进行系统设计，比较适用于较复杂的数字电路系统的设计。这种设计方法易于学习、掌握，使用较方便；其设计的可修改性、可移植性较强。对于采用原理图输入方式进行的设计，需要了解在设计中所选择的器件的内部结构及其外部特性，才能够进行更好的设计；然而，对于较复杂的数字电路系统，利用硬件描述语言进行设计，可以简化设计的复杂性。这两种设计方式的特点及其关系类似于 80X86 汇编语言及 C 语言这两种语言的特点及其相互关系，二者之间可以相互调用、相互结合。在进行系统设计等实践的过程中，可以采用原理图输入及硬件描述语言两种设计方式结合的方法，充分发挥各自的长处，使设计出的系统具有更高的性能提升。

本书利用原理图和 VHDL 语言输入方式设计、实现计算机各功能部件及主机系统。根据不同的需要，可以选择相应的设计方法在 Quartus II 平台上进行计算机组成原理的实践。

第 2 章 EDA 设计工具

EDA 工具在 EDA 技术应用中占有着重要的位置。EDA 的核心是利用计算机实现电子设计的全程自动化,因此基于计算机环境下的 EDA 工具软件是必不可少的。本章将主要介绍集成的 EDA 开发工具 Quartus II 和硬件描述语言 VHDL。

Quartus II 是 Altera 公司推出的新一代 EDA 开发软件,适用于大规模逻辑电路设计。Quartus II 软件的设计流程概括为设计输入、设计编译、设计仿真和设计下载等过程。Quartus II 支持多种编辑输入法,包括图形编辑输入法, VHDL、Verilog HDL 和 AHDL 的文本编辑输入法,符号编辑输入法,以及内存编辑输入法。

2.1 建立工程项目

2.1.1 启动 Quartus II

Quartus II 软件安装好后,在桌面上双击图标,或执行“开始”→“程序”→Altera→Quartus II→Quartus II 9.0 命令,打开如图 2-1 所示的界面。单击该界面中的 Create a New Project 按钮,可创建一个新的项目;单击 Open Existing Project 按钮,可打开一个已创建的项目;单击 Open Interactive Tutorial 按钮,可打开交互式指南,进入 Quartus II 的使用指南(此实验教程以 Quartus II 9.0 为例进行讲解,其他版本的操作过程与之类似)。

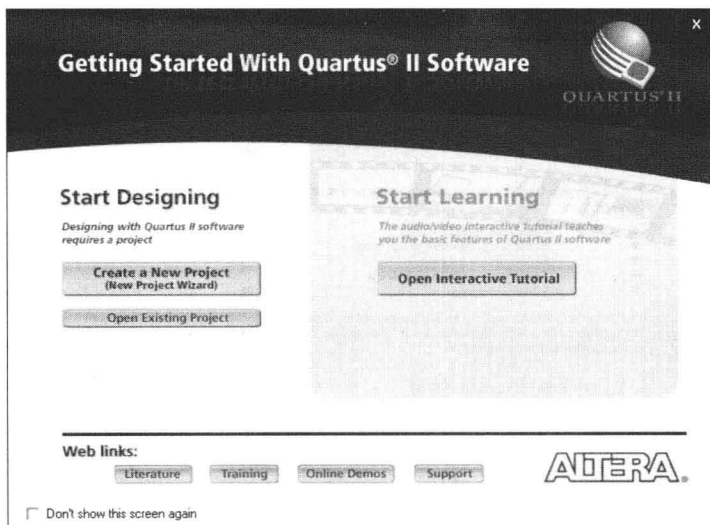


图 2-1 Quartus II 9.0 软件启动界面

单击图 2-1 中右上角的 × 按钮，可关闭当前界面进入 Quartus II 的主界面，如图 2-2 所示。如果选中图 2-1 左下角的 Don't show this screen again 复选框，以后启动 Quartus II 时，将直接进入 Quartus II 的主界面。

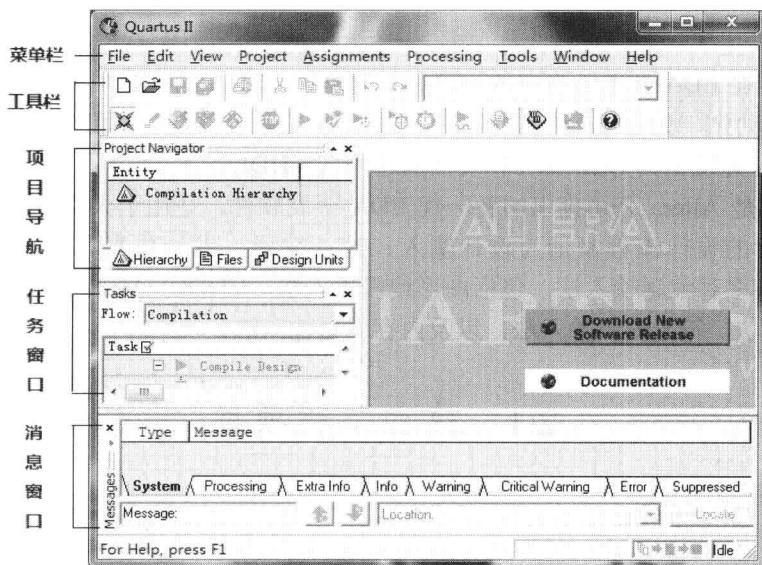


图 2-2 Quartus II 9.0 主界面

2.1.2 使用 Quartus II

1. 工程项目的创建

(1) 在图 2-1 中单击 Create a New Project 按钮，或在图 2-2 中执行 File→New Project Wizard 命令（注意，New 是新建文件，不是这里的工程），弹出如图 2-3 所示的新建项目向导对话框。

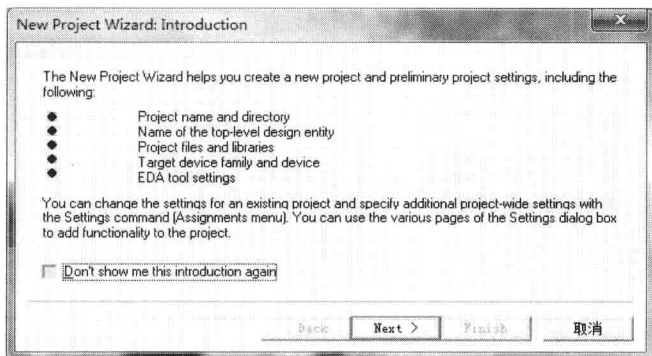


图 2-3 新建项目向导对话框

(2) 单击 Next 按钮，进入如图 2-4 所示的界面。其中，第 1 栏用于指定项目所在的工作库文件；第 2 栏用于指定项目名，项目名可以取任意名字，也可以直接用顶层文件的实

体名作为项目名（建议使用）；第 3 栏用于指定顶层文件的实体名。本例中项目的路径为 E:\cpu，项目名与顶层文件的实体名同为 cpu。

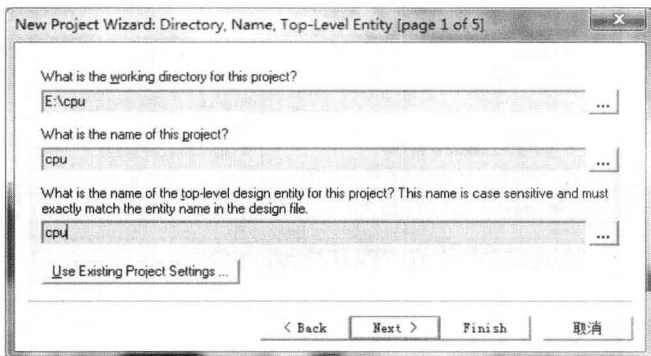


图 2-4 设置项目路径、项目名称和顶层文件名称

(3) 单击 Next 按钮，进入如图 2-5 所示的添加文件界面。由于本设计为新建的项目，还没有输入文件，所以可以不做任何操作。如果已经有文件或需要调用以前的文件，则最好先将所需文件复制到本项目文件夹下，然后再单击 Add All 按钮进行添加。

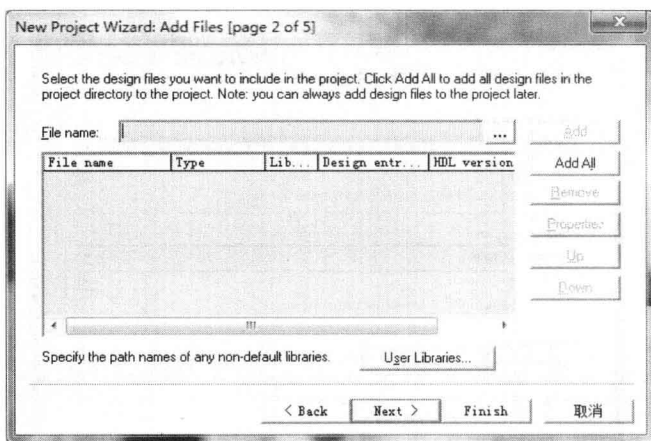


图 2-5 添加文件

注意：任何一项设计都是一个项目（Project），因此必须为其建立一个专门放置相关文件的文件夹。此文件夹将被 Quartus II 默认为工作库（Work Library）。一般，不同的设计项目最好放置在不同的文件夹中，而同一项目的文件则必须放置在同一个文件夹中，且不能将文件夹设在计算机已有的安装目录中，也不能将项目文件直接放在安装目录中。如果输入的路径是一个不存在的文件夹，则会弹出一个提示对话框，此时，单击“是”按钮即可。

(4) 单击 Next 按钮，进入如图 2-6 所示的指定目标器件界面。在 Device family 栏的 Family 下拉列表框中可选择目标器件系列；在 Show in ‘Available device’ list 栏中可指定器件的封装（Package）、引脚数（Pin count）和器件速度等级（Speed grade）来加快器件查找