

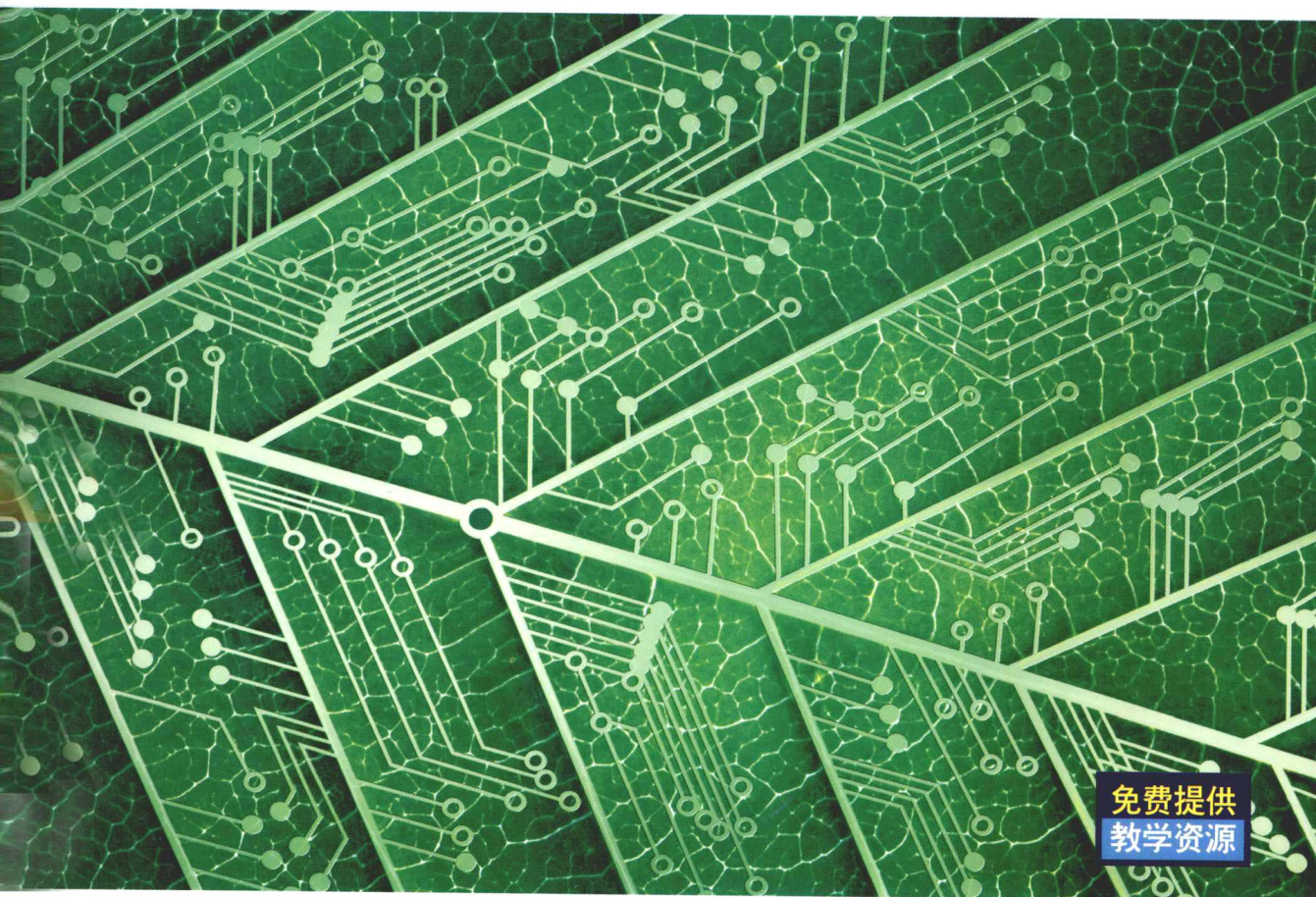


普通高等教育“十一五”国家级规划教材
普通高等教育国家级精品教材

现代计算机组成原理 (第二版)

——结构 原理 设计与SOC实现

——潘松 潘明 黄继业 编著——



免费提供
教学资源

 科学出版社

普通高等教育“十一五”国家级规划教材
普通高等教育国家级精品教材

现代计算机组成原理

——结构 原理 设计与 SOC 实现
(第二版)

潘 松 潘 明 黄继业 编著

科学出版社

北 京

内 容 简 介

本书提供了基于 HDL 和 EDA 技术的关于 CPU 和计算机系统硬件设计理论和设计技术方面较完整和丰富的内容。其中有基于微程序控制模式的 8 位 CISC 模型计算机原理和设计技术；基于状态机控制模式的 16 位实用 CISC CPU 的基本原理、设计技术和创新实践指导；基于流水线技术的 16 位 RISC CPU 设计技术；基于 32 位 OpenRISC1200 处理器系统的 SOC 软硬件构建和应用设计，以及基于经典处理器的 8051 CPU 核与 8088/8086 CPU 核构建 SOC 系统的基本理论和设计技术。在大部分章节后面，还提供了有针对性的实验与设计项目，甚至包括激发学习者创新意识和培养创新能力的 CPU 创新设计竞赛项目。

全书从授课内容到实验形式都能与目前国外计算机组成原理与计算机体系结构同类课程的教学和实验有较好的接轨。本书内容新颖实用，吸收了欧美许多高校的计算机组成原理同类课程教学和实验方面的基本要求和核心内容。首次为国内高校就这一课程的教学改革和相关实验内容的延拓方面提供了实用的教材。

本书可作为计算机专业本科生与研究生的教科书，或是作为传统的计算机组成原理课的教学与实验的补充教材，也可作为电子类专业高年级本科生、研究生进行系统设计的参考教材，还可作为相关领域工程技术人员面向片上系统开发应用的参考书。

图书在版编目(CIP)数据

现代计算机组成原理：结构、原理、设计技术与 SOC 实现/潘松，潘明，黄继业编著.—2 版.—北京：科学出版社，2013

普通高等教育“十一五”国家级规划教材·普通高等教育国家级精品教材

ISBN 978-7-03-036274-2

I. ①现… II. ①潘…②潘…③黄… III. ①计算机体系结构—高等学校—教材 IV. ①TP303

中国版本图书馆 CIP 数据核字 (2012) 第 308632 号

责任编辑：赵卫江 / 责任校对：王万红
责任印制：吕春珉 / 封面设计：多边设计工作室

科学出版社出版

北京东黄城根北街 16 号
邮政编码：100717

<http://www.sciencep.com>

百善印刷厂印刷

科学出版社发行 各地新华书店经销

*

2007 年 2 月第 一 版 开本：787×1092 1/16

2013 年 1 月第 二 版 印张：20

2013 年 1 月第一次印刷 字数：362 000

定价：33.00 元

(如有印装质量问题，我社负责调换〈百善〉)

销售部电话 010-62142129 编辑部电话 010-62138017 (H101)

版权所有，侵权必究

举报电话：010-64030229；010-64034315；13501151303

前 言

若非希望本书能更容易地被接受并融入传统“计算机组成原理”课程行列，本书的书名原本应该是“计算机组成原理与设计技术”。这显然是因为，在我国计算机高等教育的课程体系中，“计算机组成原理”这门课的名字，在长达数十年的时间内，竟与诸如“高等数学”、“大学物理”、“微机原理与接口技术”等课程的名字一样，成了对应课程的固有名词，没有人去增减一字，以致出现了在不同时段、出自众多不同作者、不同出版社、不同专业用途的面向此类课程的教材几乎取的是相同书名的现象。这似乎暗示计算机专业的学生只要是学习计算机组成的课程，就只能数十年一贯制地学原理、说原理，验证原理，且只能被动地认识计算机，使用计算机；似乎学习计算机就只能与软件打交道，只能学习软件设计而远离硬件设计，否则就脱离了计算机专业的“正业”。这对于计算机专业的课程设置与教学目标，显然是值得商榷的。以下将结合当前的计算机教学情况和本书的教学目标，对相关问题作一些探讨。最后对本书的编排和结构作一些说明。

1. 问题的提出

在多数高校计算机专业教学的课程设置结构中，不难发现其中有如下两点缺憾。

(1) 缺失基于工程实际的 CPU 硬件设计这一重要内容。

首先必须明确，CPU、嵌入式处理器、DSP 处理器乃至计算机系统的硬件设计技术及相关课程理应纳入计算机科学与技术专业中；计算机专业对于软硬件综合设计人才的培养是责无旁贷的。而电子信息、通信工程、工业自动化等专业从整个课程体系来说，都不可能插入这一课程。

如果只会使用计算机而不会设计计算机（或只会拼装计算机），只能永远依赖于别人的 CPU 或硬件平台，这显然绝非我国的办学宗旨。

随着科学技术的发展，核心技术已经越来越集中在集成电路芯片和软件这两项之中，其中 CPU 和 OS（操作系统）设计技术是最核心的两项技术。特别是高性能计算机技术一直是衡量国家实力的一个重要标志。美国、日本和西欧等国将此当作一种国家行为，不断加大这方面的资助力度。美国的许多高校本科计算机专业中都无一不是安排了 CPU 硬件设计方面的课程和实验内容。

例如美国 MIT 的一门相关课程就是“计算机系统设计”。学生在实验课中，须自主完成（即自行设计）ALU、单指令周期 CPU（single cycle CPU）、多指令周期 CPU（multi-cycle CPU），乃至实现流水线 32 位 MIPS CPU 和 Cache 的设计；Stanford 大学计算机系的本科生也有相似的课程和实验，即“计算机组成与设计”课。实验要求学生以各自独立的形式，用 VHDL 语言自主实现 CPU、VGA 显示控制模块等接口，最后实现于 FPGA 中，并完成软硬件调试。此外，如 University of California, Berkeley 和 Brigham Young University 等学校在基于 FPGA 的超级计算机研制方面也有大量成果。

然而我国在 CPU 和计算机系统设计方面还十分落后，具有成熟的完全自主知识产权的通用 8 位/16 位 CPU 产品基本没有，高端的通用 32 位到 64 位高性能处理器更是空白。国

产装备中的微处理器几乎全部采用进口的国际流行的通用或专用微处理器。这种受制于人的状况对于我国的 IT 产业、国家重要的经济军事战略乃至国家安全都十分不利!

从我国高校教学方面来看也同样不容乐观。尽管也通常包含了“计算机组成原理”和“计算机体系结构”的课程,且从这两门课程的内容来说也理应肩负这一重任(计算机系统和 CPU 硬件设计)。但实际情况并非如此,调研表明,国内除诸如中国科技大学、北京大学软件学院、复旦大学软件学院、哈尔滨工业大学、成都电子科技大学等少数高校十分重视计算机系统设计课程群建设,深入广泛地采用 EDA 技术与硬件描述语言完成实验和设计训练,并将“计算机组成原理”和“计算机体系结构”课的实验明确为 CPU 设计训练外,其他学校则大多仅将计算机组成原理定位为计算机科学导论和计算机模型认知的层面上;而在实验与实践方面,作为探讨实用 CPU 硬件原理和设计的“计算机体系结构”课基本没有对应的硬件实验。计算机组成原理的实验则主要是在一些由分离元件构成的实验平台上,完成简单模型 CPU 的验证性实验,基本谈不上设计,更没有国外高校类同的自主创新型 CPU 设计实验任务。这显然难以满足教育部在《关于加强高等学校本科教学工作,提高教学质量的若干意见》中关于“高等学校要重视本科教学的实验环节,保证实验课的开出率达到本科教学合格评估标准,并开出一批新的综合性、设计性实验”的要求。

(2) 自主创新能力培养与训练方面的课程内容偏少。

计算机学科领域中自主创新能力的培养,包括卓越工程师的培养,其目标应该是拥有自主知识产权计算机部件或硬件系统设计技术及创新设计能力的人才的培养。这就要求包含“自主”这一重要因素。

“创新”未必包含“自主”。例如某项计算机软件的设计完成,某 DSP 算法的实现,某嵌入式系统软件的开发成功等,都可能包含一些前人未曾有过的创新,甚至可以有自己的知识产权。但我们却从来没有听说过,它们会是拥有完全自主知识产权的项目。这是因为,这些尽管属于创新型项目,但却都基于现成 CPU 平台上的软件,离开了这些 CPU,软件中的所有创新价值都将归于零,因为 CPU 是别人的。这就是说,创新能力的培养绝不能脱离自主创造设计能力的培养,没有了自主的创新便不是真正的创新。

胡锦涛在 2006 年全国科学大会上提出了到 2020 年将我国建设成为创新型国家的宏伟目标,并在讲话中多次强调:建设创新型国家,核心就是坚持自主创新,将增强自主创新能力作为发展科学技术的战略基点。江泽民也曾指出:原始创新孕育着科学技术质的变化和发展,是一个民族对人类文明进步作出贡献的重要体现,也是当今世界科技竞争的制高点。

显然,如果忽视了创新型人才的培养,其他一切都将是空话。

根据这些讨论,不难发现,在计算机技术的教学中,唯有“计算机组成原理”和“计算机体系结构”课中加入相关的教学内容和合理的实验设置才能够承担起除基本知识传授外,对于学生自主创新能力的培养的重任。

毫无疑问,离开了硬件设计,特别是符合工程实际的硬件设计训练,自主创新能力的培养便无从谈起。然而目前的状况有时确实令人堪忧。不少计算机专业学生存在“重软轻硬”,“欺软怕硬”,甚至“只软不硬”的现象,学生们只将注意力和兴趣集中在各种编程环境、开发工具、数据库、计算机网络的集成技术上面,对于硬件技术的学习和应用研究不感兴趣或望而生畏。甚至计算机教学领域的个别学者都认为,计算机专业的学生可以“只要用键盘、鼠标就能演奏出各种美妙的音乐”。这种现象和认识对于我国培养自主创新

型人才显然是极为不利的。

2. 探索解决问题的方法

基于以上问题的考虑,本书给出了相应的对策,将教学目标定位于以下三点:

(1) 通过利用与实际工程相吻合的 EDA 工具,如时序仿真工具和硬件实时测试工具等,以全新的角度和方法学习并掌握计算机的结构特点和基本原理。例如可以利用这些先进的工具将执行一段应用程序过程的 CPU 中所有主要的数据流和每一个硬件微控制信号全景实时展示出来,这是传统的教学方法和实验条件无法实现的。

(2) 通过基于 EDA 技术的建模方法和必要的实践训练,初步掌握简单结构的 8 位或 16 位实用 CPU,包括复杂指令与精简指令 CPU 的设计技术、测试技术与实现方法。

(3) 通过本书从不同角度展示的 CPU 设计方案和 SOC 实现技术的启发,以及潜心完成书中给出的诸多自主创新实践项目,在此基础上,有效地激发创新意识,提高面向计算机核心部件设计的自主创新能力。

为此,本书提供了基于 VHDL/Verilog 和最新 EDA 技术的关于 CPU 和计算机系统设计理论和设计技术方面较丰富与完整的内容。其中有基于微程序控制模式的 8 位模型机设计,基于状态机指令控制模式的 16 位 CISC 实用 CPU 设计,基于 8088/8086 IP 核的完整微机系统及 8051 IP 核的 SOC 片上系统的实现,以及基于流水线技术的 RISC CPU 设计和 32 位可编程嵌入式系统设计等内容。

由于所有示例和设计都是基于 Altera 较新的 Cyclone III 系列 FPGA 硬件平台和新版 Quartus II 工具软件平台的,从而使得整个设计,从每一个逻辑门至移位寄存器,从 RAM、ROM、锁相环、Cache、ALU、DMA、中断控制器直至硬件通信接口,从 8 位 CPU 至 32 位嵌入式系统,即从最基本的部件至整个宏观系统,几乎全部能用与工程实际吻合的 Verilog/VHDL 硬件描述语言、LPM 宏模块或嵌入式 IP 来表达,并实现于一个单片 FPGA 中。同时利用此平台提供的高效的软硬件调试和测试工具进行优化设计。显然这一切是过去多年以来传统的计算机组成原理教学内容、实验模式和实验手段难以企及的。

本书中明确包含了符合现代工程设计技术的 CPU 设计理论,设计方法和实现技术,其意义是多方面的。首先,在传统的“计算机组成原理”和“计算机体系结构”课中增加了理论向工程实际转化的符合现代计算机系统工程设计规范的硬件设计内容。传统的实验模式中往往不是这样,因为在传统的实验模式中虽也有“设计”内容,但主要是根据不同实验系统各自规定的方法,用传统的分离元件(也有包括部分可编程器件)和接口器件进行拼装搭接而成。学生显然无法从这样的“设计”过程中了解真实的现代实用 CPU 基本设计技术。其次,本书能使学生在了解了计算机组成原理和软件设计技术的同时,学会计算机硬件设计技术。计算机的软硬件设计与调试技能是一个合格的计算机专业学生本应具备的基本知识和重要技能。

3. 创造能力的培养

从创造心理学的角度看,单纯的逻辑性思维和收敛性思维绝不可能产生创造,尤其是原始创造。创造型思维向来植根于发散性思维。美国心理学家吉尔福特认为,发散思维能

力是创造力的核心。一切创造和创新都是发散性思维与收敛性思维、非逻辑思维与逻辑思维、分析思维与直觉思维共同作用的结果。司马光砸缸、爱因斯坦相对论、薛定谔方程、达尔文进化论、德布罗意波粒二象性学说、麦克斯韦方程、魏格纳大陆板块漂移学说,等等,无一不是这些思维能力共同作用的经典范例。

从创造能力培养的角度看,软件设计人员只需拥有逻辑上的单向一维思维能力就能保证软件设计的成功;而过量单纯的软件设计,不断强化了设计者纯逻辑性思维和一维的收敛性思维。显然,这与培养植根于多维多向的发散性思维方式的创造能力是相违背的。因此,多数纯软件设计训练只能归类为技能性和知识性训练。

硬件系统设计则不同。首先,硬件系统可以由许多相关或互为独立的模块组成,相关模块的关系可以是同步,也可以是异步。其次,硬件系统设计本身并不能离开软件设计,因此硬件系统的构建是一个软硬综合的并发系统,设计和把握它自然必须拥有并发和多维的思想方法。而 HDL 正是描述和设计硬件系统的计算机语言,它的语句都是并发的。因此在 HDL 的 RTL 设计中,根本不存在“单步执行”的概念。除语句格式排错外, HDL 程序调试只能通过了解整个程序的时序仿真波形后才能实现。显然,一维单向的逻辑思维方式已远远不够了。因此,按照软件语言的常规设计思路是不可能用好 HDL 的。基于 EDA 技术和 HDL 的 CPU 设计训练无疑十分有利于强化发散性思维和自主创新能力的培养。

爱因斯坦说过:“想象力比知识更重要,因为知识是有限的,而想象力概括着世界上的一切,推动着进步,并且是知识进化的源泉。”因此计算机专业的教学除了传授知识外,激发学生的想象力,拓展学生对多课目的适应力,培养学生的创造力将更为重要。

4. 教材的结构、知识点与实践内容

为了进一步明确本书的教学目标,在第二版中,除了为原书名增加了副标题外,真正的变化表现在书中各章节的结构和内容的变化。对于适用范围,《现代计算机组成原理》一书仍可作为普通计算机专业的一门独立的专业基础课教材,也可以作为现在的“计算机组成原理”同类课的后续课程,或作为此类课程在教学内容和实验内容上的补充教材,或干脆用作满足新要求、赋予新任务、包含新内容和适应新时期的新的“计算机组成原理”课程的教材。

对于第二版中的结构与内容的变化及因此而凸显的特点主要表现在以下几方面:

(1) 尽量减少或删除与本书教学目标没有直接关系的内容。

考虑到目前 EDA 技术与硬件描述语言在计算机专业中有了较高的普及率,新版教材中删除了原来花大篇幅介绍硬件描述语言的章节;只是在第 2 章中简要介绍了与此后章节的实验与验证关系密切的 Quartus II 的使用方法及相关测试工具与测试技术,以便能使一些尚无任何 EDA 基础及相关预备知识的读者也能通过本书的内容而便捷入门。为此,教材还有意识地直到第 6 章之前尽可能地使用 Quartus II 的原理图设计技术而不涉及硬件描述语言的应用。此外还删除了第一版的第 9 章和第 10 章有关 Nios II 处理器的内容,这主要是因为基于 Nios II 的 SOPC 技术变化大内容多,且与“计算机原理与设计”主题的关系不是太密切,我们计划为此内容单独出版一本教材。

(2) 分散难点突出重点。

仅从讨论的对象和涉及的内容上看,第 3 章至第 5 章的基本内容与传统“计算机组成

原理”课程的对应内容十分接近，即介绍了基于微程序控制的 8 位复杂指令模型机的原理和设计；但在表述方式（LPM 宏模块或 HDL）、设计工具（Quartus II）、实现平台（大规模 FPGA）、测试工具和方法（嵌入式逻辑分析仪 SignalTap II、在系统源与信号探测器 In-System Sources and Probes 和在系统存储器实时读写器 In-System Memory Content Editor）、实现目标（工程级实用 CPU 或 SOC 系统设计）以及实验要求（以自主创新型设计项目为主要目标而非仅仅停留在原理性验证）等方面却与传统内容迥异。这三章将基于传统“计算机组成原理”课程中有关 8 位模型机原理与设计的内容，完整地放在了全新的 EDA 技术平台上来解析，并融入了规范的工程设计和大量与之配合的实验项目。这部分内容可作为本书的重点和基点。

其中，第 3 章将 LPM 宏模块调用和测试方法与 8 位模型机部件的设计融为一体，使第 4 章用较多的篇幅更详细、更宽口径地介绍此模型机的硬件构建及主要部件的设计技术，从而使第 5 章能更集中地描述模型机的工作原理、设计方法和测试技术，为后续内容的拓展和延伸奠定了基础。

显然，这三章也是传统“计算机组成原理”课的对应内容向现代计算机组成原理过渡最好的切入点，它使传统条件下以原理验证为主的实验教学方案，平稳而合理地转向了基于现代 EDA 工程的以自主创新型实验与设计实践为主的教学方案。

（3）突出创新能力和实用工程技术的掌握。

第二版在原第 6 章内容的基础上，将其改造成实用 CPU 创新设计的学习内容。即首先给出一个基本版的有限状态机指令系统的 16 位 CPU 的设计方案，在将其核心原理与设计技术进一步深化解析，以及展示了面向 CPU 软硬件测试和实现方法后，针对此设计项目所有可能的优化和升级的方向，给出了多侧面的提示。然后根据这些提示，以及第 5 章学习内容的铺垫，在实验部分为读者列出了多项自主创新设计实验项目，并在最后提出了针对本章内容的 CPU 创新设计竞赛命题，为更能动地学习计算机原理和硬件设计以及自主创新能力的提高启动了一个新的增长点。这样的流程也为后续课程的学习作了良好的示范。

为了体现 Verilog 的简洁易用和 VHDL 的严谨与行为逻辑描述的优势，第 6 章对于 CPU 中不同部件的设计使用了不同的 HDL 硬件描述语言。例如对于控制器使用了最擅长状态机描述的 VHDL；而其他模块的功能描述都用 Verilog。这种方式也延伸到了此后的一些章节。作者认为，既然是计算机专业，无论是软件语言还是硬件语言都有必要进行全面的了解，乃至熟悉。当然作者也准备了针对各部件的这两种语言描述的程序以供交流。相关的内容可参阅参考文献[1]和[2]。

（4）循序渐进，将计算机结构原理和硬件设计技术引向深入。

第 7 章与第 8 章主要讨论基于 EDA 技术和 FPGA 平台的流水线构架 RISC CPU 的设计理论、设计技术和实现方法。第 7 章介绍流水线 CPU 的基本原理和基础理论，最后增加了较多实验；第 8 章完整介绍了流水线 CPU 的设计及 FPGA 的实现方法。这两章可作为传统“计算机体系结构”课的补充，这主要是指硬件设计和实现方面。当然也可作为“计算机组成原理”课程的延伸教学内容和深入的实践项目。

（5）增加了先进而实用的 SOC 技术应用教学内容。

这部分内容包括第 9 章和第 10 章，它们主要介绍基于不同处理器核的 SOC 片上系统的结构特点、基本原理、系统构建和应用技术。

第9章介绍了32位嵌入式可编程处理器核OpenRISC 1200的系统结构和基于OpenRISC 1200的SOC系统的软硬件构建和应用设计。此章以一个SOC系统设计引入主要内容,对基于WISHBONE片内互联总线的OpenRISC 1200应用系统结构作了介绍,其中包括CPU、存储器、VGA/LCD显示控制器、编程接口、串行通信端口等模块,以及系统的指令集、WISHBONE总线结构、基于WISHBONE总线的开源IP等。此章可作为学生课外科技活动或毕业设计的内容。

第10章则基于人们十分熟悉而又经典的8051单片机与8086/8088计算机系统所对应的IP核,展示了SOC系统的构建和实用技术。此章首先介绍了基于8051单片机CPU核和大规模FPGA的SOC实现与实用技术,即将整个单片机系统置入一片FPGA中进行设计。此后的内容是基于8088核的SOC设计,首先给出了计算机系统中各主要IP部件的构建和使用方法,其中包括8088/8086 CPU IP核、8253定时器IP核、8237 DMA IP核、8259中断控制IP核、8255可编程I/O IP核和8250 UART串行通信IP核等,以及相关的存储器;最后将它们组合起来构成一个完整的计算机系统,且在一片FPGA中实现,直至实现VGA显示、PS/2键盘和鼠标控制以及启动耳熟能详的MS-DOS操作系统和Windows操作系统。本章的内容对自主设计、测试和调试自己的SOC计算机系统有很好的启发性和指导性。

事实上,在最近几年中,认可、接纳并实践本书基本教学理念和教学实验内容的高校不断增加。许多高校的计算机专业选择本书作为教材或选择教材中提到的FPGA实验系统来完成对应的硬件实验,其中有西安交通大学、西北大学、中国人民大学、吉首大学、哈尔滨工业大学、北京航空航天大学、中国民航大学和南京邮电大学等近30所高校。最难得的是,最早(2007年)加入这一教改行列的竟然是一所外语学校,即广州外语学院。该校有关教师对于外语专业学生兼修计算机软件技术和基于FPGA的计算机硬件设计技术,在拓宽就业口径、提高就业率方面所表现出的作用和优势有很高的评价。

现代计算机技术发展的速度异常迅猛,高等院校计算机科学与技术的教学将面临越来越大的挑战,这主要表现在两个方面:更多更新的知识有待传授;学生在该领域的自主创新能力有待更有效地提高。为了迎接这个挑战,本书力图在这两个方面都有所作为,但限于知识面,难免力不从心。作为抛砖引玉,望业内专家同行不吝斧正(作者E-mail: pmr123@sina.cn)。

本书作者潘明是桂林电子科技大学教师,其他二位都是杭州电子科技大学教师。

特别感谢杭州中天微系统有限公司的黄欢欢工程师为本书的重要章节增色良多!

本书的PPT配套教学课件与实验课件、实验程序、附录的mif工具软件、实验示例源文件、所涉及各类CPU IP软核等相关资料的索取可浏览网址www.kx-soc.com,也可与科学出版社(www.abook.cn)联系。

编著者

2012年11月

目 录

第 1 章 概述.....	1
1.1 EDA 技术及其优势.....	1
1.2 面向 FPGA 的数字系统开发流程.....	2
1.2.1 设计输入.....	3
1.2.2 综合.....	3
1.2.3 适配（布线布局）.....	4
1.2.4 仿真.....	4
1.3 可编程逻辑器件.....	5
1.4 FPGA 的结构与工作原理.....	6
1.4.1 查找表逻辑结构.....	6
1.4.2 Cyclone III 系列器件的结构原理.....	7
1.5 硬件描述语言.....	9
1.6 Quartus II.....	10
1.7 CISC 和 RISC 处理器.....	11
1.8 FPGA 在现代计算机领域中的应用.....	13
第 2 章 系统设计与测试基础.....	15
2.1 原理图输入设计方法的特点.....	15
2.2 原理图输入方式基本设计流程.....	16
2.2.1 建立工作库文件夹和存盘原理图空文件.....	16
2.2.2 创建工程.....	17
2.2.3 功能初步分析.....	19
2.2.4 编译前设置.....	20
2.2.5 全程编译.....	21
2.2.6 时序仿真测试电路功能.....	22
2.3 引脚设置和编程下载.....	24
2.3.1 引脚锁定.....	24
2.3.2 配置文件下载.....	25
2.3.3 JTAG 间接编程模式.....	26
2.3.4 USB-Blaster 编程配置器安装方法.....	28

2.4 层次化设计	28
2.5 SignalTap II 的使用方法	31
习题	35
实验与设计	35
 第 3 章 CPU 宏功能模块调用方法	38
3.1 计数器宏模块调用	38
3.1.1 调用 LPM 计数器及参数设置	38
3.1.2 对计数器进行仿真测试	39
3.2 寄存器与锁存器的调用	40
3.2.1 基于 D 触发器的寄存器的调用	40
3.2.2 基于锁存器的寄存器的调用	41
3.3 ROM/RAM 宏模块的调用与测试	41
3.3.1 存储器初始化文件	42
3.3.2 ROM 宏模块的调用	44
3.3.3 ROM 宏模块的测试	45
3.3.4 LPM 存储器在系统读写方法	46
3.3.5 RAM 宏模块的调用	47
3.3.6 RAM 宏模块的测试	48
3.4 信号在系统测试与控制编辑器用法	49
3.5 嵌入式锁相环使用方法	52
实验与设计	54
 第 4 章 计算机功能模块的原理与设计	57
4.1 CPU 基本功能与结构	57
4.2 计算机中的基本部件	59
4.2.1 算术逻辑单元	59
4.2.2 数据缓冲寄存器	61
4.2.3 移位运算器	61
4.2.4 程序存储器与数据存储器	63
4.2.5 程序计数器与地址寄存器	63
4.2.6 指令寄存器	65
4.2.7 微程序控制器	65
4.2.8 微程序控制器电路结构	68
4.2.9 时序发生器	71

4.3 数据通路设计	73
4.3.1 模型计算机的数据通路	73
4.3.2 模型机的电路结构	74
习题	75
实验与设计	75
 第 5 章 8 位模型计算机原理与设计	82
5.1 8 位模型 CPU 结构	82
5.2 指令系统结构及其功能的确定	85
5.2.1 模型机指令系统	85
5.2.2 拟定指令流程和微命令序列	86
5.2.3 微程序设计方法	89
5.3 CPU 硬件系统设计	91
5.3.1 CPU 顶层设计	92
5.3.2 取指令和指令译码	98
5.3.3 设计微代码表	100
5.3.4 建立数据与控制通路	100
5.3.5 控制执行单元	101
5.3.6 在模型机中运行软件	102
5.3.7 模型机整机系统时序仿真	104
5.3.8 模型机系统硬件功能测试	106
5.4 具有移位功能的 CPU 设计	110
5.4.1 移位运算器与 ALU 的结合设计	110
5.4.2 测试程序设计和模型机时序仿真	113
5.5 含更多指令的模型机设计	115
5.5.1 指令系统的格式与指令	116
5.5.2 微程序控制流程图设计	117
5.5.3 程序编辑与系统仿真	119
习题	120
实验与设计	121
 第 6 章 16 位实用 CPU 原理与创新设计	132
6.1 KX9016 结构原理及其特色	132
6.2 KX9016 基本硬件系统设计	134
6.2.1 单步节拍发生模块	136
6.2.2 运算器 ALU	136

6.2.3 比较器 COMP	137
6.2.4 基本寄存器与寄存器阵列组	138
6.2.5 移位器	140
6.2.6 程序与数据存储器	141
6.3 指令系统设计	141
6.3.1 指令格式	142
6.3.2 指令操作码	143
6.3.3 软件设计实例	144
6.3.4 KX9016v1 控制器设计	146
6.3.5 指令设计实例详解	150
6.4 KX9016 的时序仿真与硬件测试	152
6.4.1 时序仿真与指令执行波形分析	152
6.4.2 CPU 工作情况的硬件测试	154
6.5 KX9016 应用程序设计实例和系统优化	156
6.5.1 乘法算法及其硬件实现	156
6.5.2 除法算法及其硬件实现	157
6.5.3 KX9016v1 的硬件系统优化	158
习题	159
实验与设计	159
 第 7 章 流水线 CPU 原理	 164
7.1 流水线的一般概念	164
7.1.1 DLX 指令流水线结构	165
7.1.2 流水线 CPU 的时空图	167
7.1.3 流水线分类	169
7.2 与流水线技术相关的问题及处理方法	169
7.2.1 资源相关及其冲突	170
7.2.2 数据相关及其分类	170
7.2.3 数据竞争的处理技术	171
7.2.4 控制相关	177
7.2.5 流水实现的关键技术	178
7.3 流水线的性能评价	179
7.3.1 流水线的性能指标	179
7.3.2 流水线性能评估举例	181
7.3.3 Amdahl 定律	187
习题	188
实验与设计	189

第 8 章 流水线 CPU 设计	196
8.1 流水线 CPU 的结构	196
8.2 指令系统设计	197
8.3 数据通路设计	200
8.4 流水线各段的功能描述与设计	201
8.4.1 Stage 1 取指令段	201
8.4.2 Stage 2 译码段 ID	205
8.4.3 Stage 3 执行有效地址计算段 (EXE)	210
8.4.4 Stage 4 访存段 (MEM)	215
8.4.5 Stage 5 回写段 (WB)	217
8.4.6 一些关键功能部件的设计	217
8.4.7 控制单元	222
8.4.8 中断与异常	223
8.4.9 流水线 CPU 系统构建与测试	224
实验与设计	227
第 9 章 32 位 OpenRISC 软核结构及应用	239
9.1 OpenRISC1200 处理器核概述	239
9.1.1 OpenRISC1000/1200 处理器的体系结构	239
9.1.2 OR1200 指令集及指令流水线	241
9.1.3 OR1200 核的异常模型和可编程中断控制器	243
9.1.4 OR1200 核的寄存器	244
9.1.5 OR1200 核的 Tick 定时器	245
9.2 WISHBONE 片上总线	246
9.2.1 WISHBONE 总线概述	246
9.2.2 WISHBONE 接口信号说明	247
9.2.3 WISHBONE 总线协议与数据传输	249
9.3 OpenRISC 的软件开发环境	250
9.3.1 OpenRISC 的 GNU 工具链	251
9.3.2 使用 Makefile 管理工程	252
9.4 一个简单的 OR1200 核的 SOC 设计示例	252
9.4.1 KX_OR1200_SOC 概述及设计流程	253
9.4.2 KX_OR1200_SOC 的存储器结构及初始化	254
9.4.3 GPIO 通用输入输出端口	255
9.4.4 uart16550 串行通信模块应用	255
9.4.5 VGA/LCD 显示控制器设计	257

9.4.6	外设的初始化及系统的启动.....	258
9.4.7	KX_OR1200_SOC 的 $\mu\text{C}/\text{OS II}$ 移植.....	260
9.4.8	基于 SignalTap II 的硬件实时调试.....	260
9.5	基于 OR1200 的 ORPSoC 设计.....	261
	实验与设计.....	261
第 10 章	基于经典处理器 IP 的 SOC 实现.....	263
10.1	基于 8051 单片机核的 SOC 系统实现.....	263
10.1.1	K8051 单片机软核基本功能和结构.....	264
10.1.2	单片机扩展功能模块的 SOC 设计.....	264
10.2	基于 8088 IP 核的 SOC 系统实现.....	269
10.2.1	8088 IP 核 SOC 系统.....	269
10.2.2	基于 8088 CPU IP 软核的最小系统构建.....	272
10.2.3	可编程并行接口 8255 IP 核.....	274
10.2.4	8255 IP 核基本功能测试.....	276
10.2.5	8255 IP 在 8088 IP 核系统中的应用示例.....	278
10.2.6	8254/8253 IP 核可编程定时器/计数器.....	280
10.2.7	8254 IP 核基本功能测试.....	281
10.2.8	8254 IP 核在 8088 系统中的应用示例.....	283
10.2.9	8259 IP 中断控制器的功能和用法.....	285
10.2.10	8259 IP 在 8086/8088 系统中的应用.....	287
10.2.11	8237 DMA 控制器.....	288
10.2.12	16550 IP 核可编程串行通信模块.....	290
10.3	基于 8086 IP 软核的 SOC 微机系统设计.....	292
10.3.1	8086Z CPU 性能特点.....	292
10.3.2	KX86Z 微机系统的结构与功能.....	293
10.3.3	KX86Z_FULL 系统上 MS-DOS 的使用.....	295
10.3.4	在 KX86Z_FULL 系统进行 C 程序或 BASIC 程序编程.....	297
10.3.5	在 KX86Z_FULL 上启动 Windows 3.0.....	297
	习题与设计实验.....	298
附录	现代计算机组成与创新设计实验系统.....	300
1.1	KX_DN8 系列实验开发系统.....	301
1.2	mif 文件生成器使用方法.....	304
	参考文献.....	306

第 1 章

概 述

本章主要对后面各章中可能涉及的一些基本概念, 以及在此后的实验与设计中将要用到的软硬件平台作一简述。更详细的情况需要参阅书后所列的参考文献[1]和[2]。

1.1 EDA 技术及其优势

现代电子设计, 包括 CPU 和片上系统 SOC (System On a Chip) 的设计, 技术的核心已转向基于计算机的电子设计自动化技术, 即 EDA (Electronic Design Automation) 技术。EDA 技术就是依赖功能强大的计算机, 在 EDA 工具软件平台上, 对以硬件描述语言 HDL (Hardware Description Language) 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、化简、分割、综合、布局布线以及逻辑优化和仿真测试, 直至实现既定的电子线路系统功能。EDA 技术使得设计者的主要工作仅限于利用软件的方式来完成对系统硬件功能的实现, 这是电子设计技术的一个巨大进步。

EDA 技术在硬件实现方面融合了大规模集成电路制造技术、IC 版图设计、ASIC 测试和封装、FPGA/CPLD (Field Programmable Gate Array/Complex Programmable Logic Device) 编程下载和自动测试等技术; 在计算机辅助工程方面融合了计算机辅助设计 (CAD)、计算机辅助制造 (CAM)、计算机辅助测试 (CAT)、计算机辅助工程 (CAE) 技术以及多种计算机语言的设计概念; 而在现代电子学方面则容纳了更多的内容, 如电子线路设计理论、数字信号处理技术、数字系统建模和优化技术等。因此 EDA 技术为现代电子理论和设计的表达与实现提供了可能性。正因为 EDA 技术丰富的内容以及与电子技术各学科领域的相关性, 其发展的历程同大规模集成电路设计技术、计算机辅助工程、可编程逻辑器件应用, 以及电子设计技术和工艺的发展是同步的。

就过去 40 多年的电子技术发展历程, 可大致将 EDA 技术的发展分为三个阶段。

20 世纪 70 年代, 在集成电路制作方面, MOS 工艺已得到广泛的应用。可编程逻辑技术及其器件已经问世, 计算机作为一种运算工具已在科研领域得到广泛应用。而在后期, CAD 的概念已见雏形, 这一阶段人们开始利用计算机取代手工设计, 辅助进行集成电路版图编辑、PCB 布局布线等工作, 这是 EDA 技术的雏形。

20 世纪 80 年代, 集成电路设计进入了 CMOS (互补场效应管) 时代。复杂可编程逻辑器件已进入商业应用, 相应的辅助设计软件也已投入使用。而在 80 年代末, 出现了 FPGA; CAE 和 CAD 技术的应用也更为广泛, 它们在 PCB 设计方面的原理图输入、自动布局布线及 PCB 分析、逻辑设计、逻辑仿真、布尔代数综合和化简等方面担任了重要的角色。特别是各种硬件描述语言的出现、应用和标准化方面的重大进步, 为电子设计自动化

必须解决的电子线路建模、标准文档及仿真测试奠定了基础。

进入 20 世纪 90 年代, 计算机辅助工程、辅助分析和辅助设计在电子技术领域获得更加广泛的应用。与此同时, 电子技术在通信、计算机及家电产品生产中的市场需求和技术需求, 极大地推动了全新的电子设计自动化技术的应用和发展。特别是集成电路设计工艺步入了纳米阶段, 近千门的大规模可编程逻辑器件的面世, 以及基于计算机技术的面向用户的低成本大规模 ASIC 设计技术的应用, 促进了 EDA 技术的形成。更为重要的是各 EDA 公司致力于推出兼容各种硬件实现方案和支持标准硬件描述语言的 EDA 工具软件的出现, 都有效地将 EDA 技术推向成熟和实用。

与传统的数字电子系统或 IC 手工设计相比, EDA 技术具有明显的优势:

(1) 用 HDL 对数字系统进行抽象的行为与功能描述到具体的内部线路结构描述, 从而可以在电子设计的各个阶段、各个层次进行计算机模拟验证, 保证设计过程的正确性, 可以大大降低设计成本, 缩短设计周期。

(2) EDA 工具之所以能够完成各种自动设计过程, 关键是有各类库的支持。如逻辑仿真时的模拟库、逻辑综合时的综合库、版图综合时的版图库、测试综合时的测试库等。这些库都是 EDA 公司与半导体生产厂商紧密合作、共同开发的。

(3) 某些 HDL 本身也是文档型的语言 (如 Verilog/VHDL), 极大地简化了设计文档的管理。

(4) EDA 技术中最为瞩目的功能, 即最具现代电子设计技术特征的功能是日益强大的逻辑设计仿真测试技术。EDA 仿真测试技术只需通过计算机就能对所设计的电子系统根据各种不同层次的系统性能特点完成一系列准确的测试与仿真操作。在完成实际系统的安装后, 还能对系统上的目标器件进行所谓边界扫描测试, 或直接使用嵌入式逻辑分析仪进行测试。这一切都极大地提高了大规模系统电子设计的自动化程度。

(5) 无论传统的应用电子系统设计得如何完美, 使用了多么先进的功能器件, 都掩盖不了一个无情的事实, 即该系统对于设计者来说, 没有任何自主知识产权可言。因为系统中的关键性的器件往往并非出自设计者之手, 这将导致该系统在许多情况下的应用受到限制。基于 EDA 技术的设计则不同, 由于用 HDL 表达的成功的功能设计在实现目标方面有很大的可选性, 它既可以用不同来源的通用 FPGA 实现, 也可以直接以 ASIC 来实现, 设计者拥有完全的自主权, 再无受制于人之虞。

(6) 传统的电子设计方法自今没有任何标准规范加以约束, 因此, 设计效率低, 系统性能差, 规模小, 开发成本高, 市场竞争能力小。相比之下, EDA 技术的设计语言是标准化的, 不会由于设计对象的不同而改变; 它的开发工具是规范化的, EDA 软件平台支持任何标准化的设计语言; 它的设计成果是通用性的, IP (Intellectual Property) 核具有规范的接口协议。良好的可移植与可测试性, 为系统开发提供了可靠的保证。

1.2 面向 FPGA 的数字系统开发流程

完整地了解利用 EDA 技术进行设计开发的流程对于正确地选择和使用 EDA 软件、优化设计项目、提高设计效率十分有益。一个完整的、典型的 EDA 设计流程既是自顶向下设