

文章编号: 1672-5913(2012)02-0062-05

中图分类号: G642

# “计算机组成与系统结构”课程建设思路与教学实践

袁春风, 张泽生, 杨若瑜, 王 帅, 蔡晓燕  
(南京大学 计算机科学与技术系, 江苏 南京 210093)

**摘 要:**“计算机组成与系统结构”是计算机专业最重要的核心基础课之一。根据该课程内容在计算机系统的位置, 针对南京大学计算机专业人才的培养目标, 阐述该课程的教学目标与教学理念、教学内容及其组织思路, 以及相关实验教学等方面的改革思路和实践情况。

**关键词:** 计算机组成与系统结构; 计算机组成原理; 计算机系统结构; 教学改革; 课程建设

## 1 背景

计算机组成与系统结构涉及的内容位于软件和硬件的交界处, 不仅与计算机系统层次结构中下层的数字逻辑电路设计密切相关, 还和上层的操作系统、编译器等软件技术紧密关联。此外, 本课程中数据的表示、Cache 的实现、虚拟存储器的实现、过程调用以及寄存器使用约定等内容与程序设计也密切相关。它在整个计算机专业教学中起着重要的承上启下的作用。该课程对于学生全面理解计算机系统层次结构, 建立计算机系统整机概念, 了解计算机的硬件和软件之间的关系, 培养对计算机系统的分析、应用、设计及开发能力都起着非常重要的作用。特别是多核/众核处理器技术的出现, 传统的通用计算平台向嵌入式、云计算和可重构等方向发展, 使得“计算机组成与系统结构”课程的教学工作在高等学校计算机专业人才的培养方面变得更加重要和突出。

由于该课程涉及知识面广, 内容多而繁杂, 课程难度大, 概念抽象, 不少学生觉得课程内容枯燥无味, 学习兴趣下降, 大部分学生学习时似懂非懂, 考试时死记硬背, 考完后一切全

忘。因此, 该课程被大多数学生看成是最难学的课程之一。

基于本课程的重要地位和课程内容的特点, 近年来, 我们对国外多个一流大学的课程网站<sup>[1-7]</sup>进行了广泛的调研, 并认真阅读了这些大学所用的主要教材或教学参考书<sup>[8-10]</sup>, 在多个方面进行了教学研究和探索。我们在课程本身和围绕本课程的课程群建设两个方面进行了一系列的教学研究和教学改革实践。在先进教学理念的指导下, 采用科学合理的教学方式, 优化教学内容, 强化实践教学环节, 取得了较好的教学效果。

根据我们在南京大学本课程教学过程中的思考和实践情况, 笔者着重对以下几个问题进行阐述: 本课程在课程群中的位置及其与其他相关课程的关联、本课程的教学目标和教学理念、本课程教学内容的组织和改革思路, 以及本课程涉及的实践教学内容和教学模式改革。

## 2 课程在相关课程群中的位置

“计算机组成与系统结构”课程主要针对寄存器传送级以上层来介绍单处理器计算机系统中各部件的内部工作原理、组成结构以及相互联系。

**作者简介:** 袁春风, 女, 教授, 研究方向为 Web 信息检索与本文挖掘、多媒体文档处理、计算机体系结构。

包括计算机系统概述、计算机系统性能基本评价方法、数的表示及其运算、存储器的层次结构、指令集体系结构、数据通路和指令控制流程、控制器的设计、微程序设计原理、总线互连、中断和输入输出组织等基本内容。也就是说,本课程的定位主要是单处理器计算机系统中有计算机组成和系统结构的基本内容,因此,各种指令级并行处理技术也包含在其中。而对于多核/众核处理器芯片技术、多处理器计算机系统、多计算机系统所涉及到的各种内容则由其他相关课程以选修课的形式提供给学生。我系规划的“计算机组成与体系结构”课程群如图1所示。其中,“计算机系统概论”、“数字逻辑与数字系统”、“数字逻辑电路实验”、“计算机组成与系统结构”和“计算机组成与设计实验”为必修课程。从图1可以看出,“计算机组成与系统结构”课程在“计算机组成与体系结构”课程群中处于核心地位,与课程群中其他所有课程都有极其密切的关系。

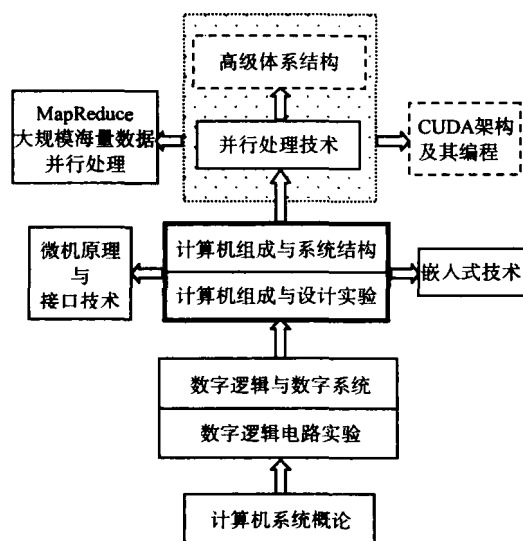


图1 “计算机组成与系统结构”在课程群中的位置

### 3 课程教学目标和教学理念

本课程为全日制本科二年级学生开设,其教学内容处于计算机系统层次结构中软件和硬件结合处,不仅涉及计算机硬件设计和指令系统

设计,还涉及到操作系统、编译程序和程序设计等软件设计技术,是整个计算机系统核心部分。根据本课程在计算机系统的位置,结合我系的生源情况和培养目标,我们将“培养学生现代计算机系统设计能力”作为本课程的核心目标,具体体现为:使学生掌握计算机系统各组成部分的工作原理、组成方法及相互关系;了解计算机系统的层次化结构概念,熟悉典型计算机系统结构,弄清硬件与软件之间的接口界面,建立计算机系统整机概念。要求学生能够设计完整的指令系统、基本功能部件、单周期CPU、多周期CPU和基本流水线CPU。为学生在后续课程的学习打下坚实的理论基础,并通过实验培养学生的动手能力,使他们在分析问题和解决问题的能力方面得到提高,为他们今后从事计算机系统结构和硬件设计、软件设计工作打下良好的基础。

根据“以培养现代计算机系统设计能力为核心”的教学目标,我们采用“从程序设计视角出发、强调软硬件关联与协同、以CPU设计为核心、以提高系统性能为目标”的教学内容组织思路,一改该课程教学中通常的就硬件讲硬件、软硬件分离的传统教学方式,从硬件设计与软件设计密切相关的角度,以系统的观念全面讲解相关的知识和技术,让学生在完成本课程基本知识体系学习的同时,重点培养学生掌握现代计算机系统的设计能力。

我们提出的教学目标和教学理念主要在以下三个方面有别于传统的教学理念。一是立足“现代”,二是强调“系统”,三是注重“设计能力”的培养。立足“现代”就是指教学内容要提升,不能仅仅停留在基本原理层面,而是要在基本原理的基础上,向学生展示现代技术的发展;强调“系统”就是指教学思路要延伸,不能仅仅局限在系统结构和硬件知识方面,也要包含相关的软件知识,要站在计算机系统的高度来阐述硬件和软件之间的接口和关联,以提高学生对于整个计

算机系统的认识水平;注重“设计能力”培养就是要让学生不仅知其然,而且知其所以然,不仅了解和掌握知识,而且会运用知识进行设计,使学生将学到的知识转化为一种运用能力。

#### 4 课程教学内容的组织和改革

根据教学目标和教学理念,我们全面系统地进行了教学内容的改革,出版了教育部普通高等教育“十一五”国家级规划教材《计算机组成与系统结构》以及配套的教学辅助教材《计算机组成与系统结构习题解答和教学指导》。在教材的内容组织以及教学过程中的内容组织方面,进行了以下几个方面的改革。

1) 由于本课程处于软件和硬件的交界处,因此,我们以“高级语言程序→汇编语言程序→机器指令→CPU设计”为主线组织教学内容,使学生深刻理解一个程序从编程设计、翻译转换、链接加载到最终在CPU上运行的整个过程,建立“从程序员视角理解计算机系统结构和硬件设计,从系统结构和硬件设计者的视角理解程序执行”的软硬件密切相关的计算机系统设计思想,从而全面地建立计算机系统的整体概念。

2) CPU设计部分在传统内容基础上大幅增加了指令流水线设计等现代计算机所用的先进技术。依照“最简单的IAS计算机CPU→总线式CPU→单周期CPU→多周期CPU→基本流水线CPU→动态超标量超流水线CPU”的次序,以“MIPS处理器”和“Pentium 4处理器”为蓝本,循序渐进地让学生了解CPU设计技术的发展过程,以及CISC和RISC处理器的不同特点。

3) 在介绍软、硬件系统之间协同工作的原理时,采用“将异常和中断的检测和响应、存储器访问和I/O访问等纳入到指令执行过程中”的教学思路,清晰说明CPU执行指令过程中硬件与操作系统相互切换和协同工作的处理过程,使

学生深刻理解软、硬件系统之间的关系以及计算机系统内部软、硬件协同工作的过程。

4) 在涉及到与程序设计有关的内容(如数据表示、信息存放方式、指令寻址方式、cache中程序访问局部性等)时,通过举例和让学生编程实验阐明硬件设计和程序设计的关系,使学生建立“从程序员视角理解计算机系统结构和硬件设计,从系统结构和硬件设计者的视角理解程序执行”的软、硬件密切相关的设计思想,在提高学生硬件设计能力的同时,也增强学生进行高效的和系统化的程序设计的能力。

5) 结合指令流水线阐述编译优化技术,“编译原理”课程通常不涉及基于流水线的编译优化技术。本课程详细讲解了在流水线中遇到的各种数据相关、控制冒险和资源冲突等影响指令流水线执行效率的问题,进而阐述如何在编译过程中使用各种优化技术来避免或减少影响流水线执行效率问题的出现。

#### 5 课程实践内容安排

根据本课程内容抽象、概念枯燥,以及实践性强的特点,我们采用了“编程实验”、“硬件设计实验”和“创新实验”三阶段递进式的创新型实验教学方案。一方面,通过编程实验使学生能从程序员角度深刻理解数据的表示、异常、cache、系统性能评测等重要内容,使得课程内容对学生来说不再是抽象、枯燥的概念,而是能通过程序的执行和调试,变成容易理解的看得见的具体数据。另一方面,硬件设计实验要求学生用硬件描述语言在FPGA板上实现一个实际的流水线CPU,通过亲自体验从设计、实现到验证的整个过程,使学生将所学知识转化成运用知识的能力。此外,我们意识到,国家迫切需要一批能掌握先进的CPU和系统软件等计算机核心技术的尖端人才,这是高校计算机人才培养的重要任

务之一。基于本课程在计算机系统中的重要核心地位和计算机人才培养中应起的作用,我们为少数对计算机底层软/硬件设计感兴趣的同学开设了创新实验课程,为这些学生提供全方位的实验支持。

### 5.1 编程实验

目前为止,有三届学生完成了以下7个编程实验。

1) hello.c 可执行程序的编译、汇编和链接生成。

通过了解高级语言源程序和目标机器代码的不同表示及其相互转换,深刻理解高级语言和机器语言之间的关系,以及机器语言和不同体系结构之间的对应关系。

2) 无符号数和带符号整数的表示。

通过无符号数和带符号整数之间的相互转换来理解无符号数和补码整数在计算机中的表示及其相互之间的差别和联系;并深刻理解高级语言中常量或变量的真值与寄存器或内存单元中的机器数表示之间的对应关系。

3) 浮点数的表示。

通过对 float 和 double 型数据的各种编程实践,了解 IEEE 754 浮点数在计算机中的表示、以及对一些特殊运算(如除数为0、负数开方等)的处理方法;深刻理解 IEEE 754 标准中各种特殊位序列表示的作用,以及浮点数运算中各种“异常”事件的真实含义。

4) 数据的存放顺序和对齐方式。

通过对结构型变量中各种不同长度数据类型变量及其地址的打印输出,了解数据在机器中的存放方式和对齐方式,并深刻理解编译器与计算机体系结构之间的关系。

5) 类型转换和移位操作运算。

通过高级语言中数据类型的转换和移位操作结果,更好地理解指令系统设计和计算机硬件设

计所需满足的要求和需要考虑的问题。

6) 整数和浮点数的算术运算。

通过检查高级语言程序中不同数据运算的相应结果,进一步理解高级语言中的运算表达式与运算电路之间的对应关系、机器代码在 CPU 中的执行过程、“溢出”异常的产生条件、浮点数运算中的“大数吃小数”问题,以及 gcc 编译器和 x86 架构中对浮点数处理不足的地方。

7) cache 和程序访问的局部性。

通过对二维数组设置不同的访问顺序、不同的地址空间区域和不同的数组元素个数,来验证局部数据块大小对程序执行效率的影响,从而了解程序访问的局部性对带有 cache 的计算机系统性能的影响。

通过编程实验,大大增强了学生对课程内容的感性认识,使枯燥、抽象的课程内容变得具体而有趣,从而培养学生对课程学习的兴趣,也使学生更加深刻理解了课程内容与程序设计之间的关系。

### 5.2 硬件设计实验

近年来我们参考国外名校本课程的教学思路,对教学和实验内容进行了大幅更新,在课堂教学内容中加大了对 CPU 设计、特别是流水线 CPU 设计的力度,使学生打下了很好的理论和技术基础,将实验内容与课堂教学完全结合起来,改变了原先课堂教学和实验教学内容脱节的现象,因此,除了实验教材外,课堂教学的教材本身也成为了实验教材的延伸。

为了加强对学生的计算机系统,尤其是 CPU 设计能力的培养,提出并进行了“以流水线 CPU 设计为核心”的高起点创新性实验教学改革,采用“基于硬件描述语言和 FPGA,以功能部件设计为先导,单周期 CPU 和多周期 CPU 设计为过渡,最终实现流水线 CPU”的实验教学方法。

CPU 设计实验分成以下两个阶段完成:功能

部件设计实验阶段(不分组)和综合设计实验阶段(3~4人/组)。内容主要包括:寄存器组、ALU和ALU控制器、桶形移位器、乘法器、除法器、单周期CPU、多周期CPU和流水线CPU的设计。其中,最终的流水线设计为综合设计实验。

从最近三届学生实验结果来看,大部分学生都能完成一个具有20多条基本指令的单周期或多周期CPU;部分同学能完成一个五段基本流水线CPU设计;有些能力强的学生甚至完成了带转发和阻塞处理功能(包括Load-use冒险、分支冒险、Jump指令阻塞、乘法指令阻塞)、并能处理异常和中断的五段流水线CPU设计。

通过CPU硬件设计实验,大大加深了学生对课程内容的理解,不仅掌握了CPU设计的理论知识,也深刻理解了整个计算机系统的设计思想和方法,同时还培养了部分学生对计算机硬件设计的兴趣和信心。

### 5.3 创新实验课程

创新实验课程以学生自愿为主,在完成课堂教学和实验教学阶段以后,专门为少数学生开设,由课程实验小组的老师和硕士研究生指导。创新实验课程的范围主要涉及数字电路设计、计算机功能部件设计、CPU设计、ISA模拟器设计

等。例如06级的一个学生小组在2008年实验课程结束后,我们继续为其开设了“MIPS处理器设计”创新实验课程,继续开展流水线CPU设计创新项目研究,设计实现了一个与MIPS32 2.62兼容的82条指令(包括21条算术运算、13条跳转、14条存取、8条逻辑运算、6条移位、6条数据传送以及14条自陷指令)的五级流水线CPU,运算部件包括超前进位加法器、桶形移位器、基于基4-Booth编码和Wallace树压缩的乘加器以及基于SRT的除法器。

## 6 结语

“计算机组成与系统结构”课程是计算机专业最重要的核心课程之一。近年来,我们在该课程本身和围绕该课程的课程群建设两个方面进行了相应的教学研究和教学改革实践。通过借鉴国内外先进的教学理念,采用科学合理的教学方式,优化教学内容和实验教学设计,在本课程及其课程群建设方面取得了一系列教学成果。我们将继续对本课程的教学进行深入改革,进一步完善课程的理论和实验教学内容,并编写出版相应的实验教材,使本课程的教学质量达到国际一流大学的水平。

#### 参考文献:

- [1] MIT. Computation Structures [EB/OL]. [2011-09-01]. <http://6004.csail.mit.edu/>.
- [2] UC Berkeley. Machine Structures [EB/OL]. [2011-09-01]. <http://inst.eecs.berkeley.edu/~cs61c/sp11/>.
- [3] UC Berkeley. Components and Design Techniques for Digital System [EB/OL]. [2011-09-01]. <http://inst.eecs.berkeley.edu/~cs150/sp11/>.
- [4] Stanford University. Computer Organization and Systems [EB/OL]. [2011-09-01]. <http://www.stanford.edu/class/cs107/>.
- [5] Stanford University. Digital Systems II [EB/OL]. [2011-09-01]. <http://www.stanford.edu/class/ee108b/>.
- [6] CMU. Introduction to Computer Systems [EB/OL]. [2011-09-01]. <http://www.cs.cmu.edu/~213/>.
- [7] CMU. Introduction to Computer Architecture [EB/OL]. [2011-09-01]. <http://www.ece.cmu.edu/~ece447/>.
- [8] Yale N. Patt, Sanjay J. Patel, Introduction to Computing Systems From Bits and Gates to C and Beyond [M]. 2nd ed. McGraw-Hill Education (Asia) Co. and China Machine Press, 2006.
- [9] Randal E. Bryant, David R. O'Hallaron. 深入理解计算机系统 [M]. 龚奕利, 雷迎春, 译, 北京: 中国电力出版社, 2011.
- [10] David A. Patterson, John L. Hennessy. 计算机组成和设计硬件/软件接口 [M]. 3版. 郑纬民, 译. 北京: 机械工业出版社, 2007.

(编辑: 郭小明)