

《计算机组成原理》课程教学大纲

一、课程名称：计算机组成原理

二、课程代码：CST 31116

三、课程英文名称：Computer Organization and Design

四、课程负责人：钟 将

五、学时与学分：64 理论学时 + 16 实验学时 4.5 学分

六、课程性质：专业必修课

七、课程类型：专业基础课

八、适用专业：计算机科学与技术、网络工程、信息安全、物联网

九、选课对象：计算机学院的所有本科生，以及部分跨专业的硕士生，也适合于辅修和选修计算机为第二专业的本课程

十、预修课程：数字逻辑、汇编语言程序设计等

十一、使用教材：

[1] 计算机组成与设计:硬件/软件接口(第五版)，机械工业出版社，David A. Patterson, John L. Hennessy.

[2] Computer Orgnaization and Design (The Hardware /Software Interface), Elsevier, David A. Patterson, John L. Hennessy.

教材选用说明：

本课程采用双语教学，选用国际知名高校所采用的英文教材，为了方便到学习过程中便于对照，特给出中文翻译版本。该教材采用的指令集包括 ARM 指令集和 MIPS 指令集两个版本，为了便于学习，建议采用 MIPS 指令集版本的教材。

十二、参考书目：

[1] 蒋本珊, 计算机组成原理, 清华大学出版社.

[2] 白中英, 计算机组成原理 (第四版 立体化教材), 科学出版社.

[3] 李亚民, 《计算机组成与系统结构》, 清华大学出版社.

[4] 袁开榜, 《计算机组成原理》, 高等教育出版社.

十三、开课单位：计算机学院

十四、课程描述

《计算机组成原理》课程是计算机专业一门核心主干必修课。从课程的

地位来说,它是《计算机操作系统》、《微机接口原理》、《计算机系统结构》等课程的先导课程,在整个专业课的教学中,起着承上启下的作用。本课程的主要任务是以单机系统为研究对象,阐述计算机系统的硬件组成以及各部件的工作原理以及逻辑实现,课程使学生理解计算机基本工作原理包括:主要硬件部件的工作原理和实现方法、二进制算法、计算机内部的数据通路、取指令到指令执行的硬件实现、计算机内的程序执行过程。通过课程学习,最终建立起计算机硬件系统的完整概念,从而能够根据特定需求设计自定义指令集架构,通过 SOC 硬件实现能够运行特定指令集架构程序的 CPU。

十五、 课程目标

1. 本课程支撑的毕业要求

(1) R1.6 能够运用计算机专业知识,设计和实现计算机系统硬件及核心模块;

(2) R2.2 能够应用计算机软硬件技术,针对复杂计算机工程问题选择恰当的数学、自然科学和计算机科学等相关知识进行分析,得到相关工程问题的解决途径;

(3) R2.3 能够应用计算机科学与技术的基本原理,通过文献研究,深入分析复杂计算机工程问题,以获得有效的结论;

(4) R5.2 针对复杂计算机工程问题,能够运用图书馆、互联网、数据库等多种资源,检索和分析所需要的软硬件开发工具的相关资料,熟练掌握开发环境与工具的使用方法;

(5) R5.3 能够使用合适的软硬件开发工具对复杂工程问题进行预测和仿真模拟,并对结果进行合理评价。

(6) R9.1 具有团队意识,能够与其他学科的团队成員有效沟通,合作共事,理解一个团队中每个角色对于整个团队的意义和作用;

2. 本课程拟达到的教学目标

本课程是以单机系统为研究对象,阐述计算机系统的硬件组成以及各部件的工作原理以及逻辑实现,帮助学生建立起整机概念,从而能够在理解计算机工作原理的基础上,按照自定义的指令集架构实现完整的 CPU 主机的数据通路,具体的主要教学目标:

- (1) 具备系统硬件设计和实现能力。理解计算机的基本工作原理,理解计算机各主要部件的硬件结构、相互联系和作用,具备设计主机系统的数据通路的能力,最后能够应用硬件描述语言来设计和实现计算机存储器及存储系统、运算器、寄存器堆、控制器等核心部件,并能够将这些部件构成一个完整的主机系统;
- (2) 具备体系结构的研究能力。能够理解指令和数据编码方法,并

能够设计自定义指令集架构（ISA），并通过设计和实现数据通路来实现该指令集架构，理解和运用精简指令 MIPS32，通过文献研究和存储程序的原理来理解精简指令集的发展与演化过程和最新进展，从而学习利用文献研究来发现精简指令集 CPU 设计的思路；

- （3）具备系统硬件研究及方案设计能力。通过文献研究，理解流水线处理器中各种冒险机制和解决方案，利用互联网资源设计和实现经典五级流水线处理器和性能优化方案，了解计算机硬件系统面临的能耗问题，理解国内外计算机硬件技术的最新的进展情况，理解缓存技术优化原理，能够应用量化分析方法对计算机的性能、功耗和成本，以及改进系统性能的方法；
- （4）具备芯片设计与验证等工具链的应用能力。熟练掌握硬件描述语言开发软件系统，能有对设计的系统进行调并下载到实际开发板上运行，学生需要通过查阅互联网资源和软件开发使用指南，从过自主学习掌握最新的设计工具和 IP 核的使用，能够自主学习汇编程序工具调试和开发 MIPS 汇编程序设计任务；
- （5）具备利用硬件电路的仿真与验证能力。熟练掌握硬件描述语言开发软件系统对逻辑功能仿真和时序仿真，了解仿真结果与中的功耗估计和性能估计与实际电路之间的差异；
- （6）具备专业领域的书面沟通和团队协作能力。通过小组协作方式完成设计型实验报告和课程报告，在团队协作联系其团队成员之间的有效沟通和合作共事能力，并理解每个成员在合作中的分工、理解团队的意义和作用。

3. 课程教学目标与毕业要求的关系

课程教学目标	毕业要求二级指标点					
	R1.6	R2.2	R2.3	R5.2	R5.3	R9.1
子目标 1	✓					
子目标 2		✓				
子目标 3			✓			
子目标 4				✓		
子目标 5					✓	
子目标 6						✓

课程教学的子目标 1 是培养学生理解计算机基本工作原理,对于计算机运算器、控制器、存储器等主要硬件的工作原理,并通过数据通路联接这些部件来实现取指令和执行指令,从而理解如何通过硬件来执行程序,学生能够应用硬件描述语言(HDL)设计和实现计算机存储器及存储系统、运算器、寄存器堆、控制器等核心部件,并通过 SOC 技术实现完整的 CPU 和主机系统,从而支撑**毕业要求指标点 R1.6** 能够运用计算机专业知识,设计和实现计算机系统硬件及核心模块;

子目标 2 是培养学生利用文献研究和存储程序原理来理解精简指令集的发展与演化过程和最新进展,从而学习利用文献研究来发现精简指令集 CPU 设计的思路,通过联系利用文献研究方法来寻找复杂工程问题的解决途径,从而支撑**毕业要求指标点 R2.2**;

子目标 3 是通过文献研究,理解流水线处理器中各种冒险机制和解决方案,利用互联网资源设计和实现经典五级流水线处理器和性能优化方案,从而培养学生对于复杂工程问题中解决方案的设计能力,支撑**毕业要求指标点 R2.3**;

教学子目标 4 是培养学生掌握硬件描述语言开发工具,通过小组协作方式完成设计型实验报告和课程报告,实现 CPU 和主机的设计,并通过仿真测试系统的正确性,将设计文件下载到实际开发板上运行。学习过程中学生需要自主收集各种资料学习和掌握指令集架构(ISA)设计能力、汇编程序的编程能力、FPGA 硬件板卡使用等能力,从而支撑**毕业要求指标点 R5.2**。

教学子目标 5 是培养学生掌握硬件开发工具的仿真评估,通过逻辑仿真、时序分析、关键路径分析、能耗估计以及资源使用情况,不断优化设计方案。此外,利用 MIPS 汇编模拟仿真开发环境 MARS 来开发和调试汇编程序,理解模拟器工作原理和局限性。通过上述开发工具链的使用,完成相应的实验报告和课程设计项目来**支撑毕业要求指标点 R5.3**;

通过教学子目标 6 的培养过程中,通过小组协作方式完成设计型实验报告和课程报告,在团队协作联系其团队成员之间的有效沟通和合作共事能力,并理解每个成员在合作中的分工、理解团队的意义和作用,从而支撑**毕业要求指标点 R9.1**。

十六、 课程教学方式/方法

对于课程教学目标 1 主要采用课堂教学的方式,通过知识教授、问题讨论、课后习题等途径,使学生能够理解和应用计算机系统硬件和核心模块的设计方法。

对于课程教学目标 2 和目标 3,目标 4 和目标 5 主要采用设计型的实验、课程综合设计报告,通过 CPU 核心部件、数据通路的设计型,使学生能够熟练使用硬件设计工具。

对于课程教学目标 5,主要通过分组实验和分组综合设计实验项目来

锻炼团队合作与沟通能力，特别是使用实验报告和项目报告的撰写培养学生的书面沟通能力。

课程采用双语教学的方式，授课老师可选择采用翻转课堂的方式。

十七、课程教学内容与学时

1. 课堂教学（40 学时）

一级知识点	二级知识点	三级知识点	掌握层次	学时
1. 计算机系统概论	1.1 计算机的分类以及发展简史	1.1.1 计算机的分类及特性	记忆	6
		1.1.2 计算机的发展简史	理解	
	1.2 程序概念	1.2.1 计算机系统层次结构	掌握	
		1.2.2 高级语言与硬件语言	理解	
	1.3 硬件概念	1.3.1 计算机五大部件	应用	
		1.3.2 处理器与存储器制造技术	理解	
		1.3.3 计算机性能评价	应用	
		1.3.4 功耗墙概念	理解	
	1.4 计算性能测试基准	1.4.1 SPEC 测试基准程序	理解	
		1.4.2 SPEC 功耗测试基准程序	理解	
	1.5 阿姆达尔定律	1.5.1 系统性能改进的基础	理解	
		1.5.2 阿姆达尔定律	应用	
2. 指令系统	2.1 指令分类	2.1.1 基本指令	理解	12
		2.1.2 指令的基本格式	应用	
		2.1.3 高级语言编译成机器指令	理解	
	2.2 指令中操作数	2.2.1 变量（数据）表示	应用	
		2.2.2 数据对齐方式	应用	

		2.2.3 内存的读写操作	理解	
	2.3 指令的表示	2.3.1 指令中操作码	应用	
		2.3.2 数据的寻址方式	应用	
		2.3.3 指令的寻址方式	应用	
	2.4 常用指令	2.4.1 逻辑操作指令	记忆	
		2.4.2 决策指令	记忆	
		2.4.3 过程处理指令	记忆	
		2.4.4 同步处理	记忆	
	2.5 程序的翻译与执行	2.5.1 程序翻译的层次	应用	
		2.5.2 编译器	理解	
		2.5.3 链接器	理解	
		2.5.4 加载器	理解	
	2.6 机器指令与程序举例	2.6.1 简单的过程	记忆	
		2.6.2 排序过程	记忆	
	2.7 数组与指针	2.7.1 数组	应用	
		2.7.2 指针	应用	
		2.7.3 数组和指针性能	记忆	
	2.8 ARM 指令集	2.8.1 ARM 指令集简介	记忆	
		2.8.2 ARM 寻址方式	记忆	
	2.9 X86 指令集	2.9.1 x86 指令集简介	记忆	
		2.9.2 x86 指令寻址方式	记忆	
	2.10 指令集设计的原则	2.10.1 简单整性原则	应用	
		2.10.2 越小越快	应用	
		2.10.3 好的设计需要折中	应用	
	2.11 汇编程序设计	2.11.1 汇编程序设计方法	应用	
3. 数的表示与运算	3.1 定点数运算	3.1.1 补码加法	应用	10
		3.1.2 补码减法	应用	
		3.1.3 加法器设计与实现	记忆	
		3.1.4 乘法器设计与	记忆	

		实现		
		3.1.5 二进制除法	记忆	
		3.1.6 快速乘法	理解	
	3.2 浮点数运算	3.2.1 浮点数的表示	应用	
		3.2.2 IEEE754 单/双精度表示方法	应用	
		3.2.3 浮点数的加/减运算	应用	
		3.2.4 浮点数的乘/除运算	应用	
		3.2.5 浮点数运算器设计	记忆	
		3.3.6 浮点数运算的精度问题	理解	
	3.3 浮点运算指令	3.3.1 浮点指令和寄存器	记忆	
		3.3.2 浮点运算器	记忆	
		3.3.3 浮点运算指令的优化	理解	
4. 处理器设计	4.1 基本的MIPS指令集	4.1.1 指令功能与编码	应用	18
		4.1.2 寄存器和寻址方式	应用	
		4.1.3 简单CPU实现模型	应用	
	4.2 单周期处理机	4.2.1 单周期处理机数据通路	应用	
		4.2.2 单周期处理机控制器设计	应用	
		4.2.3 指令执行时数据通路上的操作	应用	
	4.3 多周期处理机设计	4.3.1 多周期处理机数据通路	记忆	
		4.3.2 多周期处理机控制器设计	记忆	
		4.3.3 多周期处理机中指令执行时数据通路上的操作	记忆	

	4.4 微程序控制器设计	4.4.1 微程序控制器原理和特点	应用	
		4.4.2 微指令、微命令	应用	
		4.4.3 微程序控制器的工作过程	应用	
		4.4.4 微程序的设计	应用	
	4.5 流水线处理器设计	4.5.1 流水线的基本概念	理解	
		4.5.2 流水线相关	应用	
		4.5.3 流水线控制器设计	应用	
5. 存储系统	5.1 程序局部性原理	5.1.1 时间局部性	应用	8
		5.1.2 空间局部性	应用	
		5.1.3 不同存储器的性能分析	记忆	
		5.1.4 多层次存储系统	应用	
	5.2 Cache 系统	5.2.1 Cache 的基本概念	应用	
		5.2.2 Cache 访问过程	应用	
		5.2.3 Cache 的缺失处理	应用	
		5.2.4 Cache 写操作过程	记忆	
		5.2.5 Cache 的性能评估	记忆	
		5.2.6 Cache 系统的设计	记忆	
	5.3 虚拟存储系统	5.3.1 虚拟存储器概念	应用	
		5.3.2 段式管理、页式管理和段页式内存管理	应用	
		5.3.3 虚拟地址到物理地址的转换过程	记忆	
		5.3.4 页表与 TLB 快表	应用	
		5.3.5 TLB 缺失和缺页处理	记忆	

6. I/O 系统	6.1 总线系统	6.1.1 总线概念与分类	应用	8
		6.1.2 总线接口	应用	
		6.1.3 总线的仲裁机制	应用	
		6.1.4 总线定时与数据传输机制	记忆	
		6.1.5 PC 机总线系统的演化	理解	
		6.1.6 多级总线系统	应用	
		6.1.7 现代总线系统	记忆	
	6.2 接口系统	6.2.1 接口概念	应用	
		6.2.2 接口构成	应用	
		6.2.3 接口分类	记忆	
	6.3 I/O 系统中数据交换	6.3.1 程序方式	应用	
		6.3.2 中断概念	应用	
		6.3.2 中断处理过程	应用	
		6.3.3 中断实现的机制	应用	
		6.3.4 多级中断实现方式	记忆	
		6.3.5 DMA 概念	应用	
		6.3.5 DMA 处理过程	应用	
		6.3.6 DMA 实现的方式	记忆	
7. 复习与展望			记忆	2

注释 1：课程内容按照相对独立性，用层次结构列出 1、2、3 级知识点单元。

注释 2：对知识学习的要求一般分为“记忆，理解，应用”三个层次。

2. 实验教学（16 学时）

（1）实验目的

通过计算机组成原理实验课程学习，掌握简单运算器的数据传送通路，验证运算功能发生器的组合功能。加深对静态随机存储器芯片的工作原理以及工作时序的理解。掌握微程序控制器的工作原理。在掌握部件单元实验的基础上，进一步组成一台基本模型计算机，加深对计算机整机理解，从而巩固课堂知识、深化学习内容、完成教学大纲要求，学好计算机科学与技术专业的专业基础课。

(2) 实验内容及学时分配

序号	实验名称	实验内容	实验软硬件环境要求	学时数分配
1	运算器与存储器实验	1) 掌握运算器的组成结构和工作原理; (2) 学习 Verilog 不同形式的编程方式, 理解 assign 和 always 的区别; (3) 掌握存储器阵列 (memory array) 的原理, 并使用 Block Memory Generator 设计一个合适大小的 RAM; (4) 掌握存储器的读写的过程。	(1) PC 机一台; (2) 计算机系统能力培养实践平台 (MINISYS 定制开发板); (3) Xilinx Vivado 开发套件 (2017.x 版本)。	4
2	单周期 CPU 取指译码实验	(1) 掌握单周期 CPU 控制器的工作原理及其设计方法。 (2) 掌握单周期 CPU 各个控制信号的作用和生成过程。 (3) 掌握单周期 CPU 执行指令的过程。 (4) 掌握取指、译码阶段数据通路执行过程。	同上	4
3	简单单周期 CPU 实验	(1) 掌握不同类型指令在数据通路中的执行路径。 (2) 掌握 Vivado 仿真方式, 实现指令执行的仿真和验证。 (3) 实现一个单周期的 CPU, 支持 11 条类似 MIPS32 简化指令	同上	4
4	流水线 MIPS 处理器设计	(1) 掌握流水线 (Pipelined) 处理器的思想; (2) 掌握单周期处理中执行阶段的划分; (3) 了解流水线处理器遇到的	同上	4

		冒险； (4)掌握数据前推、流水线暂停等冒险解决方式。		
--	--	--------------------------------	--	--

备注：有关各次上机实验的具体内容、实验开展方式等要求，请参见“附录：《计算机组成原理》实验报告”中的详细说明。

(3) 实验要求

- a) 学生每 2 人一组。
- b) 学生课前自学掌握 Windows 操作系统使用和基本命令操作。
- c) 学生实验前预习计算机组成原理实验指导书中关于实验版以及 FPGA 开发软件的相关介绍。
- d) 课内实验与课外实验结合。

(4) 实验教材（讲义、指导书）

同课程教材。

(5) 实验对课程教学目标的支持

实验教学内容围绕单机系统中的核心部件：运算器、存储器、控制器，并利用这些部件构造出一个能够实现简单指令集的原型机。通过实验教学，建立起建立计算机硬件系统的完整概念，以及软硬件如何协同的机制。从而具备计算机工作原理的专业核心知识。

在实验环节要求学生熟练掌握 VHDL（Verilog HDL）语言的开发工具、MIPS 汇编编程仿真工具(Mars 或者 WinMIPS)。

可以通过应用课程知识达成课程教学的各项目标。

十七、 能承担此课的教师

钟将、叶春晓、冯永、黄仁、肖春华、李季等老师。

教学大纲制订者：钟 将

教学大纲审定者：冯 永