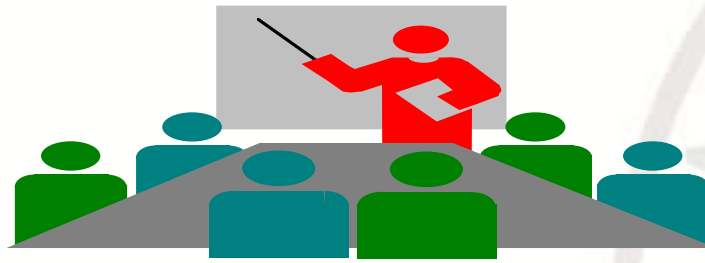




浙江大学
ZHEJIANG UNIVERSITY



计算机组成与设计

Computer Organization & Design

The Hardware/Software Interface

Say Something

施青松

Asso. Prof. Shi Qingsong

College of Computer Science and Technology, Zhejiang University

zjsqs@zju.edu.cn



Some information

□ TA:

- Miao Wang(王淼) 3150104650@zju.edu.cn
- Jianwen Liu(刘剑文), jianwen_liu219@163.com)
- 2018级学长

Teaching Web :

◆ 浙江大学: [学在浙大](#)

◆ <http://10.14.30.103>

<http://10.214.26.108>

◆ <ftp://10.14.30.103:10000>

<ftp://10.214.26.108:10000>



Background

It is **very easy** to design CPU **IP** Core!

It is **not easy** to design good CPU!

To design successfully is far more
difficult than one!

For example ➡



日本法政大学李亚民教授设计的单周期处理器 Within SOC

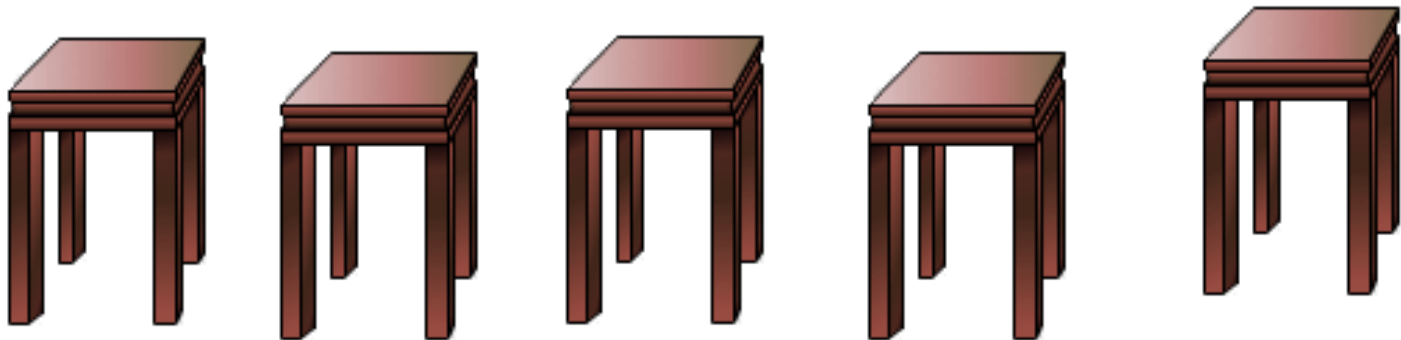
[illegible]

Course Status

■ Hardware and software interface

- One of the courses of the Graduate Record Examination

Soft: Assembly → compile → OS → ALGOL → **Software Engineering**



HW: digital circuits → **Organization** → Implementation → Interface → architecture

Software Professional

Computer professionals



立足基础、加强实践、服务专业、进入国际

- 数字逻辑课程：计算机组成相关部件的设计 **基础**
 - 组合电路设计、时序电路设计
- 计算机组成：设计简单 **RISC-CPU** 核 **核心**
 - ALU 部件
 - 单周期实现、多周期实现简单的32位-CPU
 - RISC: RISC-V/MIPS
 - 写入FPGA，用实验板卡做测试验证
 - 简易计算机系统（微控制系统）
- 计算机系统结构：设计流水线 **RISC-CPU** 核心 **提高**



Curriculum Text Book

Computer Organization & Design

—The Hardware/Software Interface

John L. Hennessy

Stanford University

David A. Patterson

California University, Berkeley



Experimental teaching materials

Say something: How to learn? and ---- Cultivation and harvest



□ 孟子曰：

舜发于畎亩之中，傅说举于版筑之间，胶鬲举于鱼盐之中，管夷吾举于士，孙叔敖举于海，百里奚举于市。

□ 故天将降大任于斯人也，必先苦其心志，劳其筋骨，饿其体肤，空乏其身，行拂乱其所为，所以动心忍性，曾益其所不能。

□ 人恒过，然后能改。困于心，衡于虑，而后作。征于色，发于声，而后喻。入则无法家拂士，出则无敌国外患者，国恒忘。

□ 然后知生于忧患，而死于安乐也。



The secret of success

- 舜从田野之中被任用，傅说从筑墙工作中被举用，胶鬲从贩卖鱼盐的工作中被举用，管夷吾从狱官手里释放后被举用为相，孙叔敖从海边被举用进了朝廷，百里奚从市井中被举用登上了相位。 **出道**
- 所以上天将要降落重大责任在这样的人身上，一定要道先使他的内心痛苦，使他的筋骨劳累，使他经受饥饿，以致肌肤消瘦，使他受贫困之苦，使他做的事颠倒错乱，总不如意，通过那些来使他的内心警觉，使他的性格坚定，增加他不具备的才能。 **磨难**
- 人总有故，过而能改善莫大焉；心有困或，反复思索、思虑，然后才能有所作为；表现于身体，抒发于言语中，然后才被人了解。在内如果没有坚持法度的世臣和辅佐君主的贤士，在外如果没有敌对国家和外患，国终久是要亡的。 **历程**
- …担忧和患难使人生存，而安逸享乐使人萎靡死亡。 **祖训**



The role of classroom teaching

- 教学是双方互动的，不能一边倒。大学应素质教育为主，要鼓励学生在教师指导下的自学与动手。
- 课堂教学作用是：**引出**知识及相关知识点，**引导**学生猎取知识的方向，分析知识的难点，**学会**分析讨论解决问题的**方法与途经**，节省课余时间，**提高**自学的**效率**
- 学会‘**止于至善**’，知道‘**物极必反**’
 - **大学之道**，在明明德，在亲民，**在止於至善**。知止而後有定，定而後能静，静而後能安，安而後能虑，虑而後能得。物有本末，事有终始，知所先後，则近道矣。



Guidelines Practice: 知识的感性化

□ 注重知识的系统性、连贯性，强化实践能力

- 立足组成，三位一体：从**程序员视角俯视计算机组成结构**
- 知其然，知其所以然：从**工程师视角仰视计算机组成结构**

□ 培养自主学习能力

- 引出组成及相关知识的自主获取和消化方法
 - 力求充分体现培养学生硬件知识的自学方法
- 引导猎取知识的方向，给出分析问题的途经
 - 节省课余时间，提高预习、复习、自学的效率。

□ 启发式、鼓励式课堂交互

- 引出关键问题，开展提问和讨论
- 培养讨论，争论，辩论的学习气氛
- 核心、重要知识点学生上台
- *课程设计presentation

Experimental schedule



1	Assignments: Assembler Design*	Elective	MIPS汇编器设计(实验课外作业)
2	Exp01: 逻辑实验模块优化一		多路选择器与CPU辅助模块设计
3	Exp02: 逻辑实验模块优化二		七段显示部件(设备)扩展
4	Exp03: IP核集成SOC设计(2 Weeks)		建立调试、测试和应用环境
5	Exp04: 逻辑实验模块优化三		ALU与Register Files CPU设计辅助: IP核设计CPU
6	Integrated Experiment1: 单周期CPU设计实现 Exp05-08	Elective	数据通路设计 控制器设计 指令扩展设计 ARM模式中断设计*
7	Integrated Experiment2: 多周期CPU设计实现 Exp09-12		多周期测试框架建立 多周期数据通路设计 多周期控制器设计 多周期指令扩展设计
8	Final Project: CPU simple application 微控制器或简单SOC应用设计		基本I/O设备扩展设计与实现(简单IO接口) 含完整的应用程序
9	ALU功能扩展设计*	Elective	32位乘法器、除法器
10	MIPS中断扩展*	Elective	MIPS中断功能及应用实现



Course Grade

- 平时占10% (Assignments) ;
 - 作业、讨论、阅读：光盘 + 一篇论文
 - 期中闭卷(英文)考试, 10%(统一时间段)
 - **5.4 A Simple Implementation Scheme**
 - 期末闭卷(英文)考试, , 30%;
 - 基本实验占10%;
 - Assignments: 10%
 - Integrated Experiment1~2: 占20%;
 - Final Project, 占20% 2019级教学案例采用RISC-V, Project选做
 - 附加分奖励: 优秀作品
- 注: 卷面成绩不到45分(满分100分)者, 总评不及格。



Final Practical examination

□ 实验考查：分4项内容进行现场考核

■ 1. 现场验收或检查综合性课程设计

□ 结合前期验收；

■ 2. 从20个非常简单实验题中抽取1题

□ 现场完成；

■ 3. 现场口试

□ 针对学生已做实验和考试题提问，现场回答；

■ 4. 综合能力

□ 平时实验过程、现场操作和口试等实验相关的综合能力（熟练程度）。

Content at Classroom



□ Chapter One: Introduction

□ Chapter Two: Instructions: Language of the Computer

2.1 Introduction	62
2.2 Operations of the Computer Hardware	63
2.3 Operands of the Computer Hardware	67
2.4 Signed and Unsigned Numbers	74
2.5 Representing Instructions in the Computer	81
2.6 Logical Operations	89
2.7 Instructions for Making Decisions	92
2.8 Supporting Procedures in Computer Hardware	98
2.9 Communicating with People	108
2.10 RISC-V Addressing for Wide Immediates and Addresses	113
2.11 Parallelism and Instructions: Synchronization	121
2.12 Translating and Starting a Program	124
2.13 A C Sort Example to Put it All Together	133
2.14 Arrays versus Pointers	141
2.15 Advanced Material: Compiling C and Interpreting Java	144

Content at Classroom-2



□ Chapter Three: Arithmetic for Computers

3.1 Introduction	174
3.2 Addition and Subtraction	174
3.3 Multiplication	177
3.4 Division	183
3.5 Floating Point	191
3.6 Parallelism and Computer Arithmetic: Subword Parallelism	216

□ Chapter Four: The Processor: Datapath and Control

4.1 Introduction	236
4.2 Logic Design Conventions	240
4.3 Building a Datapath	243
4.4 A Simple Implementation Scheme	251
4.5 An Overview of Pipelining	262
4.6 Pipelined Datapath and Control	276
4.7 Data Hazards: Forwarding versus Stalling	294
4.8 Control Hazards	307
4.9 Exceptions	315
4.10 Parallelism via Instructions	321

Content at Classroom-3



□ Chapter Five: Large and Fast: Exploiting Memory Hierarchy

5.1 Introduction	366	
5.2 Memory Technologies	370	
5.3 The Basics of Caches	375	
5.4 Measuring and Improving Cache Performance		390
5.5 Dependable Memory Hierarchy	410	
5.6 Virtual Machines		416
5.7 Virtual Memory		419
5.8 A Common Framework for Memory Hierarchy		443
5.9 Using a Finite-State Machine to Control a Simple Cache	449	

□ Chapter Last: Storage, Networks, and Other Peripherals

8.1 Introduction	
8.2 Dependability, Reliability and Availability	
8.3 Disk Storage	8.4 Flash Storage
8.5 Connecting Processors, Memory, and I/O Devices	
8.6 Interfacing I/O Devices to the Processor, Memory, and Operating System	
8.7 I/O Performance Measures: Examples from Disk and File Systems	
8.8 Designing an I/O system	
8.9 Parallelism and I/O: Redundant Arrays of Inexpensive Disks	

- ❑ **How does Hardware support HLL?**
- ❑ **Arithmetic for Computers**
- ❑ **Datapath and Control**
- ❑ **Exploiting Memory Hierarchy**
- ❑ **Storage, Networks, and Other Peripherals**
- ❑ **Experimental**



◎ THINKS!