

CH6 附录A 附录B

课程基于

《计算机组成与设计：硬件/软件接口》5e

Patterson & Hennesy 著

B站 翼云图灵

CH6 从客户端到云的并行处理器

CH3讨论了子字并行（数据级并行），CH4讨论了指令级并行
本章介绍了更高级的**任务级并行**，主要通过**多核**和**多线程**实现
然而**并行编程**面临着调度、任务分割、负载均衡、同步时间、通信开销等一系列问题

Flynn根据指令流和数据流的数量，将计算机分为SISD、SIMD、MISD、MIMD
其中，**单指令流多数据流（SIMD）**计算机对向量数据进行操作，性能优于标量机

游戏市场的快速发展促进了**图形处理单元（GPU）**性能的高速提升
GPU采用数量更多、功能更专用化的核心进行计算，已广泛用于科学计算、AI计算

无论是计算机**集群**、WSC还是多核处理器，都需要某种网络实现多个处理机之间的通信
Roofline模型被广泛应用于并行浮点性能的评价

通过使用OpenMP中的pragma语句启用多线程，矩阵乘法的性能提升了14倍

作者寄语

软硬件接口上的并行变革也许是过去60年来所面临的最大挑战。你也可以像我们快速浏览各章章节时所展示的那样，把它当做最大的机遇。这个变革在IT界内外提供了大量的研究和商业前景，并且主导多核的公司并不一定与主导单处理器的公司相同。在理解了硬件发展的潮流、学会了如何根据硬件来改进软件之后，也许你就能抓住其中的机会，成为创新者中的一员。我们期待从你的发明创造中获益！

附录A 汇编器、链接器和SPIM仿真器

汇编语言拥有**编译器输出语言**和**编程语言**两种角色

汇编语言依赖与特定机器的指令集体系结构

且因为长度过大，汇编程序开发效率极端低下、阅读维护极端困难

出于这些缺点，汇编语言现已很少用作编程语言

但是，在以下四种情况中，仍然有必要使用汇编语言进行编程：

- ①性能要求极端苛刻，如控制汽车刹车的嵌入式系统
- ②需要深入硬件底层优化程序
- ③需要使用定制指令
- ④为没有可用编译器的老旧计算机编写程序

上述内容属于A.1节，后续关于MIPS汇编语言程序设计的内容不再涉及

附录B 逻辑设计基础

数字逻辑研究基于二进制数学即布尔代数的逻辑

我们可以用门电路、真值表、逻辑方程式（组）来表达数字逻辑

本书采用与门-或门阵列实现SOP两级表示

PLA、ROM、译码器、多路选择器是常见的组合逻辑单元（B.1~B.3）

Verilog是基于C的硬件描述语言，广泛应用于硬件设计（B.4）

从1位加法器开始逐步扩充、连接，构建了32位MIPS ALU这一典型的组合单元
并将行波进位加法器改进为超前进位加法器，将16位加法性能提高约6倍（B.5~B.6）

内部含有存储器的状态单元依赖时钟进行读写，从触发器和锁存器开始
构建了带纠错机制的SRAM、DRAM和有限状态机三种状态单元（B.7~B.11）

现场可编程器件普遍用于实现非定制的复杂逻辑功能（B.12）

B站 翼云图灵