

Introduction

Background

什么是 computer architecture

Computer architecture, like other architecture, is the art of determining the needs of the user of a structure and then designing to meet those needs as effectively as possible within economic and technological constraints

需要大量的工作，长久的经验累积，富有创造性.....

一个架构师的角色可以分为四个方面

- look backward (to the past): 理解和分析过去的设计
- look forward (to the future): 设计新的架构，推动技术发展
- look up (towards the problems to solve): 理解问题，解决问题
- look down (towards device technology): 理解现有技术的能力，预测将来技术的发展，超前设计

Introduction to Computer Architecture

Layered design

计算机架构的分层：软件-ISA-硬件

分层是为了分治，减少复杂度

体系结构中两个重要的层

- ISA (Instruction Set Architecture) 是软件和硬件的接口，编译器使用 ISA 告知微处理器需要的操作，微处理器使用 ISA 了解需要实现的内容
- Microarchitecture (uarch) 是实现 ISA 的硬件结构

作为架构师，需要有跨越层次的眼光，需要理解软件之下的硬件是如何工作以及影响软件的运行

Performance

衡量系统性能的两个重要指标

- 执行时间

- 吞吐量

除此之外还有衡量能耗，成本，可靠性等的指标

没有完美的设计，设计就是要做 tradeoff

提高一个指标的同时一般会降低其他的指标（Pareto Optimum）

边际效应

一个 tradeoff 的例子：考虑有 N 个不同的指标 x_i ，需要最大化

$$\prod x_i$$

等价于最大化

$$\sum \log x_i$$

且有限制

$$\sum c_i x_i = B$$

则最终的解为

$$c_i x_i = \frac{B}{N}$$

Quantitative Approach

Back of the envelope calculations

基本思想：简单但不简略的计算，留有余量

e. g. CPI (Clock Per Instruction)，通过用程序的时钟周期数除以程序的执行指令数得到

模拟也可以用于量化分析，但快速的模拟过于粗糙，而 RTL 模拟的时间消耗是不可接受的

有三个指标可以评估模拟器

- 速度
- 灵活性
- 准确性

高层次的模拟就是牺牲一部分准确性换来速度和灵活性的提升

Trends

过去的十年内 Moore's law 和 Dennard scaling 都达到了瓶颈，单核性能的提升极为有限

现在主要的系统：移动系统，大型的机房

现在主要的瓶颈：memory wall, power wall, ILP wall