

第一章

1.6 性能

- 性能 = 1 / 执行时间;
- 性能的度量: 时间;
- CPU时间 = 用户CPU时间 + 系统CPU时间;
- 阿姆达尔定律: 提升后性能 = 受影响时间 / 影响因素 + 不受影响时间;

第二章

- 设计原则1: 简单来自于规整;
- 设计原则2: 更少则更快;
- 设计原则3: 优秀的设计需要折中;
- 指令格式:
 - funct7 rs2 rs1 funct3 rd opcode (r-type)
 - imm rs1 funct3 rd opcode (i-type)
 - imm7 rs2 rs1 funct3 imm5 opcode (s-type)
 - 000000 imm6 rs1 funct3 rd opcode (srli)
 -
- slli sll 立即数左移;
- xori 按位异或;
- jalr rd, rs1, offset: $x[rd] = pc + 4; pc = (x[rs1] + sext(offset)) \& \sim 1;$
- jalr 12位, jal 20位没bit0,
- lb sb字节加载
- lh sh半字加载
- lw sw字加载
- lui 左移12位加入
- beq 12位
- 编译器: 将C语言转化为汇编
- 汇编器: 将汇编转化为机器语言模块 (处理伪指令)
- 连接器: 将各个独立模块合并, 解析未定义标签生成可执行文件
- 加载器: 将目标程序放到内存中已准备执行

第三章

- 浮点偏移: 127、1023

第四章

- 单周期缺陷:
 - 指令数据存储器必须分开;
 - 所有操作必须在一个时钟周期内完成;
 - 电路频率受延时最大的指令限制, 使用DRAM时限制极大
- ID计算beq目标地址, EX分支完成, 访存地址计算, MEMlw读 store, WBlw写
- 流水线特点:
 - 没有缩短单个任务时间;
 - 提高整体吞吐量;
- 分支预测移动至ID
- 异常时
 - 保存发生PC值;
 - 保存错误原因;
 - 跳转到异常处理函数。
- 非精确异常: 阻塞流水线, 保存状态;

- 依赖性限制了并行性的挖掘，复杂性导致功耗墙。

第五章

- 时间局部性：某个数据项被访问，将来可能再次被访问；
- 空间局部性：某个数据项被访问，相邻数据项可能很快也将被访问。
- 旋转延迟：转半圈的时间；
- 增大块大小可以减少失效率；
- 对于固定的cache，增加大小导致块数量减少，导致更多竞争，增加失效率；
- 块越大，失效造成的延时越大；
- 可用性 = 平均失效时间 / (平均失效时间 + 平均维修时间)
- 强制失效：没有在cache中出现过的块进行第一次访问产生的失效；
- 容量失效：不可能容纳所有请求而导致的失效
- 冲突失效：组相联和直接映射竞争同一组，全相联中不存在
- 增加cache容量，降低失效率，可能演唱访问时间；
- 增加相连度，减少冲突失效，降低失效率，可能延长访问时间；
- 增加块容量，降低失效率，增大失效损失，还可能增大失效率；

第六章

- 异常由内部引起，中断由外部引起；
- 中断源提出请求：请求标记INTR；
- 中断优先级：软硬件实现；
- CPU相应
 - 相应条件：EINT = 1，允许中断触发器；
 - 乡音时间：指令执行周期结束，CPU发查询信号；外部中断异步，内部中断同步（马上相应）；
 - 中断银指令：包含程序断点（PC和PSW），寻找服务程序入口地址，硬件关中断（EINT=0，INT=1），转ISR；
- 恢复；
- 寻找入口地址：
 - 硬件向量法；
 - 软件查询发；
- 多重中断：中断屏蔽技术，提前设置开中断（中断服务开始之后）；
- RISCv处理方式：
 - 处理程序地址由中断原因决定；
 - 向量表及地址寄存器加上异常向量地址；
 - 处理中断或者跳到实际处理程序。
- 非精确异常：允许已经进入流水线中的指令执行完再去执行中断处理；
- 精确异常：完整保存当前状态；
- 查询法 向量法
- ISR：入口地址
- 断点，现场

第八章

- 总线：构成计算机系统二点互联结构，连接系统中多个部件的信息传输线；
- 宽度：数据总线位数；
- 贷款：每秒传输字节数；
- 单总线：要求其上部件必须高速运行完成操作；
- 多总线：将IO与主存分离；
- 现行总线结构：
 - 数据传送总线；
 - 仲裁总线；
 - 集中式仲裁：链式查询，计数定时，独立请求；
 - 分布式仲裁：自举分布式仲裁，冲突检测，并行竞争；
 - 中断和同步总线；
 - 公用线；

- 广播：一个主设备对多个从设备的写操作；
- 广集：多个从设备对一个主设备的读操作。
- 总线周期：
 - 申请分配；
 - 寻址阶段；
 - 传送阶段；
 - 结束阶段。
- 总线通信：
 - 同步通信：用于总线长度较短，各部件存取时间一致；
 - 异步通信：；
 - 半同步通信：工作速度差异较大；
 - 分离式通信：大型计算机系统。
- IO编址方式：
 - 独立编址：不占用主存，需专用指令；
 - 统一编址：减少主存容量，无需专用指令；
- IO接口：可以是两种硬设备之间的连接电路，也可以是两个软件之间的共同逻辑边界；也叫**适配器**；
- 信息交换方式：
 - 程序查询方式；
 - 程序中断方式；
 - DMA方式；
 - 数据不经过CPU，直接与主存交换；
 - 周期挪用：DMA优先于CPU，适合IO读写周期大于主存周期的情况；
 - 交替访问；
 - 通道方式；
- 中断 DMA
- 程序 硬件
- 指令执行结束 存取周期结束
- 能 不能
- 传送数据 溢出、后处理
- 低 高