

作业2

4.1 CISC

优势: 1. 复杂指令可以在一个时钟周期内执行多个操作

2. 可以使用少量代码来实现复杂功能

3. 支持大量的寻址模式和数据类型

劣势: 1. 指令长度不确定, 导致了难以预测性能

2. 硬件设计比较复杂, 成本高昂

3. 易出现新指令增加导致的性能问题

RISC

优势: 1. 指令长度固定, 并且只执行基本操作码, 可提高效率降低成本

2. 简化CPU结构, 减少了芯片面积、功耗等开销

3. 更好地支持流水线技术, 在相同频率下比CISC有更高的运行速度

劣势: 1. 编写程序需要更多代码实现相同数量功能

2. 不支持某些特殊用途或旧式应用程序所需的部分操作码

2. 是 RV32I 和 RV64I

扩展

1. RV32M / RV64M: 乘法扩展, 并支持整数乘、除法和取模运算。

适用于需要高效执行数字信号处理

2. RV32A / RV64A: 原子操作扩展, 允许在单个时钟周期内执行原子读-修改-写操作, 以确保数据完整性, 适用于并发编程中需要对资源进行同步或互斥访问的应用程序

3. RV32F / RV64F: 浮点数运算扩展, 支持单精度和双精度浮点运算, 适合于图形计算等需要大量使用浮点数运算符号的应用程序

4. RV32D / RV64D: 双精度浮点数运算扩展, 类似于3, 但专注于更高精度双精度数据

扫码使用

夸克扫描王



5. RV32G/RV64G: 压缩指令集执行环境: 将 RISC-V 汇编语言转换成紧凑格式, 从而使可执行文件占据更少空间, 降低带宽需求, 在网络传输节省空间。

4. 1) RV32I 中的 `add` 指令和 RV64I 中的 `addw` 指令有相同的 opcode, 都是 0x33. 原因是 RISC-V 采用了可扩展设计, 使得不同数位版本之间共享大部分操作码。

在 RV32I 和 RV64I 中, 其采用了不同大小的寄存器, 但其操作完全相同。

2) 是, 在 RISC-V 中, 所有的整形运算都默认按补码方式进行。

5. 包含了用于提示处理器执行优化的指令。

这些指令不会直接影响程序的正确性和结果, 但可以提高性能和效率。

6. `addi a0, a1, 16`, `addi a1, a1, -15`

`div a2, a0, a1` # $a_2 = \frac{16}{-15} = -1-3$

`rem a3, a0, a1` # $a_3 = 1$

对整数除法和取余运算都采用了有符号整数运算的规则。若两个操作数异号, 则结果向零舍入; 否则保持与被除数相同的符号。

11. 1) 偏移量寻址。

2) 内存直接寻址

3) 立即数寻址

4) 寄存器直接寻址

5) 偏移量寻址。

扫码使用

夸克扫描王

