

第四周作业

1. CISC: 优势: 少量指令实现大量功能, 同时节省了程序本身的大小, 降低内存空间的占用, 对编译器和程序存储空间的要求较低。
劣势: 指令集中包含的指令非常多, 而且功能复杂, 难以理解、学习, 且硬件设计复杂, 测试验证难度较高。

RISC: 优势: 一个时钟周期能同时运行多条指令, 效率得以提升, 同时指令集相对较少, 便于理解, 且硬件设计较为简单, 适合利用流水线提升性能。

劣势: 完成同样的功能, 相比CISC要使用更多的指令, 导致内存空间要消耗的更多, 且对编译器设计的要求较高, 程序的代码密度较低。

RV28I

2. RISC-V中的基本指令集有RV32I, RV32E, RV64I。扩展指令集M, 扩展了整数乘法和除法指令; 扩展指令集A, 扩展了并发操作中的原子指令; 扩展指令集F, 扩展了单精度浮点数运算指令; 扩展指令集D, 扩展了双精度浮点数运算指令; 扩展指令集Q, 扩展了四精度浮点数运算指令。

4. (1) RV32I中的add指令与RV64I中的addw指令的opcode不同。add为0110011, 而addw为0111011。

RV32I中的add指令与RV64I中的add指令有相同的opcode, 均为0110011。

这样的设计是因为可以使得RV64I可以兼容RV32I中的add指令(及其他指令)。因为RV32I是RV64I的子集, 且同时又能区分32位加法与64位加法。

(2) 需要进行额外的符号扩展。因为进行符号扩展后可以使得数据在后续的64位计算中保持原本的正负性以及大小。

指令

5. RISC-V的I标准指令集中的HINT^{指令}空间是为了提供一组用于调试和性能分析的指令。这些指令不执行任何操作, 而是将一些信息反馈给处理器或调试器, 或者触发一些硬件行为, 以便优化代码或调试程序。

作用: HINT指令能允许程序员和调试器更好地分析和调试程序, 同时还能提高性能, 在

处理器层面上进行更好的协同处理, 并更加有效地解决多处理器场景下的内存共享

6. 答: $\text{div } a2, a0, a1$ $a2 = a0 \div a1 = -3$

$\text{rem } a3, a0, a1$ $a3 = a0 \% a1 = 1$

RISC-V的标准指令集中, 除法和余数指令都有符号和无符号两种。无符号指令应为 divu, remu 。而有符号的除法和余数指令为默认表示, 且有符号除法遵循负负得正, 而有符号取余则是结果与被除数同号。

11. (1) $\text{jal } ra, 0x88$ 立即数寻址
(2) $\text{jalr } x0, ra, 0$ 立即数寻址
(3) $\text{addi } a0, a1, 4$ 立即数寻址
(4) $\text{mul } a0, a1, a2$ 寄存器直接寻址
(5) $\text{ld } a4, 16(sp)$ 偏移量寻址