

1. 简要分析 CISC 和 RISC 架构各自的优势和劣势

CISC 架构: 优点: 对编译器和程序存储空间的要求较低, 指令长度灵活
单个指令完成的任务量大且功能复杂

缺点: 硬件设计复杂, 测试验证难度较高

RISC 架构: 优点: 硬件设计较为简单, 适合利用流水线提升性能

缺点: 对编译器要求高, 程序代码密度低。

2. RISC-V 中的基本指令集为 RISC-V I 指令集

标淮扩展指令集: RISC-V M 包含乘法、除法、取模求余指令

RISC-V F 单精度浮点指令

RISC-V D 双精度浮点指令

RISC-V Q 四倍精度浮点指令

RISC-V C 压缩指令, 单指令长度 16 位, 改善程序大小

4. (1). RV32I add opcode 为 0110011 ... RV32I-add 与 RV64I-addw 不同

RV64I addw opcode 为 0111011 RV32I-add 与 RV64I-add 相同

RV32I add opcode 为 0110011

RV64I 为 RV32I 的扩展, 相同的指令名称一操作的, 具有可移植性

(2). 不需要 addi 和 addiw 指令将结果存入 rd 目标寄存器时已进行符号位扩展

addw $\Rightarrow rd \leftarrow \text{sign-extend}(tmp[31:0])$

addiw $\Rightarrow rd \leftarrow \text{sign-extend}(tmp[31:0])$

5. Hint 指令通常用于向微架构传达性能提示,除堆栈 PC 和可用的性能计数器外,并不改变任何体系结构可见的状态。

RV32I 为 HINT 指令保留了一大片编码空间,91% 的 HINT 空间保留给了标量 HINTS,剩余的则保留给定制 HINTS

6. $a2 = -3$ div 指令 $rs1/rs2$ 先进行有符号数扩展
 $a3 = 4$ divu 指令 $rs1/rs2$ 进行无符号数扩展
rem 指令 $rs1/rs2$ 有符号取余,余数为正
remu 指令 $rs1/rs2$ 无符号取余

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