

日期:

3.14

①

CISC: 优点: 对编译器和程序存储空间要求较低

缺点: 硬件设计复杂, 测试验证难度较大

RISC: 优点: 硬件设计较为简单, 适合利用流水线提升性能

缺点: 对编译器设计的要求较高, 程序代码密度较低

②

指令集: RV32I/RV64I.

1. RV32I/64G: 提供通用压缩指令集, 能在大部分应用场景提供良好的性能与代码密度
2. RV32I/64M: 提供垂直扩展指令, 适用于如 DSP、ISP 等高效率阵列处理器
3. RV32I/64A: 提供原子操作指令, 适用于多线程应用, 如高性能服务器
4. RV32I/64F: 提供浮点指令集, 高效处理浮点计算, 用于科学计算、ISP 等
5. RV32I/64C: 提供压缩指令集, 相对于 G 进一步提升代码密度, 性能

④

(1) 前者不同, `addw` 对 32 位 `int` 加法运算并将结果符号扩展到 64 位, 而非对 2 个 64 位 `int` 或 32 位 `int` 的简单加法, 混用 `opcode` 会导致冲突
后者相同, 可能因为 RV64I `add` 已在 RV64I `mode` 扩展, 保持兼容性, 从而易于扩展, 提高可移植性。

(2) 不用, 这 2 指令执行时已对 32 位结果符号扩展至 64 位, 并将其存放在目标寄存器中, 因此后续计算可直接使用目标寄存器的值。

⑤

这是一组专门用于提供对处理器的提示和建议, 没有明确语义的指令

日期: /

作用: ①为硬件软件提供额外信息, 可用于系统故障预测, 性能管理等

②为编译器和汇编器添加灵活性, 可在不影响程序行为时插入 NOP

③为未来标准扩展预留空间

⑥

$$a_2 = -3, a_3 = 1$$

除法 div: 向0舍入。当被除数绝对值小于除数绝对值, 结果为0。然后, 若两
符号相同, 结果为正; 不同, 则为负。

乘数 mul: 向0舍入。如果除数为0, 不整除则符号同被除数。

⑪

内存直接寻址。

寄存器间接寻址。

立即数寻址。

寄存器直接寻址。

偏移量寻址。