

日期:

3. 14

①

CISC: 优点: 对编译器和程序员空间需求较低

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, 指令代码密度较低

②

基准指令集: RV32I/RV64I.

1. RV32/64I: 提供通用压缩指令集, 能在大部分应用领域提供良好的性能与代码密度
2. RV32/64M: 提供重叠宏指令, 适用于如DSP、ISP等高数据流运算场景
3. RV32/64A: 提供原子操作指令, 适用于多线程应用, 如多核服务器
4. RV32/64F: 提供浮点指令集, 高效处理浮点计算, 用于科学计算、DSP等
5. RV32/64C: 提供压缩指令集, 相对于I进一步提高代码密度, 性能

③

(1) 两者不同: addw 对于32位Int 加法运算并将结果符号扩展到64位, 而非对两个64位Int 或32位Int + 32位加法, 混用 opcode 会导致冲突

后者相同, 可能因为 RV64I 的 add 已经 RV32I 的 add 扩展, 保持兼容性, 从而易于移植, 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, 并将其存放回目标寄存器中, 因此后续计算可直接使用目标寄存器的值。

④

这一组专门为提供对处理器的提示和建议, 没有明确定义的指令

日期： /

作用：①为硬件软件提供额外信息，可用于系统检测、维修管理等

②为编译器和汇编器增加灵活性，可在不影响程序功能时插入 NOP

③为未来标准扩展预留空间

⑯

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

除法 div：向 0 取余。当被除数绝对值小于除数绝对值，结果为 0。然后若二者符号相同，结果为正；不同，则为负。

余数 rem：向 0 取余。如果整除则为 0，不整除则符号同被除数。

⑰

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。