

1. CISC 和 RISC 架构各自的优劣.

	CISC	RISC
优势	① 指令丰富、功能强大 ② 对编译器的要求低 ③ 指令长度灵活、寻址方式多 ④ 源代码短、内存需求低	① 指令精简、使用率均衡 ② 对硬件设计的要求低 ③ 指令长度固定，可在一个时钟周期内执行完成 ④ 易于切分流水线，易于实现
劣势	① 采用微码控制器设计，执行时间更长 ② CISC 处理器设计复杂，研发成本高	① 受限于访存的速度 ② 依赖于 RISC 的随机逻辑 CPU 需同步于指令集进行软、硬件设计成本高。

2. RISC-V 的基本指令集是什么？列举 5 个常见的标准扩展指令集并说明作用和应用范围。

基本：RV32I 基本 32 位 整数 指令

RV32E 只使用了寄存器的基本 32 位 指令

RV64I 基本 64 位 整数 指令

扩展：M 扩展了整数乘法和除法指令，应用范围广，且是其它扩展指令集的基础。

F 扩展了单精度浮点数运算指令，增加了 32 个 32 位 浮点寄存器，适用于单精度浮点数的运算，对提高计算速度有一定帮助。



A. 标准原子扩展，包括加载保留/条件存储器指令以及取和操作存储器指令，适用于多线程同步操作
C. 原子扩展，定义了部分指令而以位版本。适用于小内存的嵌入式应用。

V 向量扩展，包括向量运算和向量条件运算等指令，广泛应用于图像处理、机器学习等领域。

4. 17. 均具有相同的指令操作数。RV64I 中的 add 和 RV32I 中的 add 是一样的，addw 的不同之处在于，它可以操作 2 个 32 位数的加法，将高位截断，并进行符号位扩展。addw 的加入，是为了在 RV64I 中支持 32 位数。

3. 在 RV64I 中，32 位数将以符号扩展的形式存放放在 64 位寄存器中，addw 和 addiw 取计算结果应该是已经扩展过的，所以不再需要进行额外的符号扩展。

5. 指令对某些
当有一些操作数无效时，HINT 指令可以提示该操作码为未来的标准微架构保留，对于一些简单的实现，HINT 可以被当作正常的指令来操作而不会影响体系结构状态。

6. $a_2 = \frac{-3}{2}, a_3 = 1$

DIV rdq, rs1, rs2. rs1 是被除数，rs2 是除数，rdq 是商

REM rdr, rs1, rs2. rs1 是被除数，rs2 是除数，rdr 是余数

rdq 不能和 rs1 或 rs2 为同一个寄存器。



11. 0. jal ra, 0x88 立即教导地
22. jalr xo, ra, 0 寄存器间接寻址
32. addi ao, ai, 4 立即教导地
42. mul ao, ai, a2 寄存器直接寻址
52. ld ay, 1b(sp) 偏移量寻址



扫描全能王 创建