

1. CISC 单个指令完成的任务量大且功能复杂，指令长度灵活。

优点：对编译器和程序存储空间的要求较低。

缺点：硬件设计复杂，测试验证难度较高。

RISC：单个指令完成的任务量少且功能单一，指令长度相对固定。

优点：硬件设计较为简单，有利于流水线提升性能。

缺点：对编译器设计的要求较高，程序的代码密度较低。

2. RISC-V 指令集是目前较为流行的一种 RISC 架构指令集，同时它也是寄存器一寄存器型指令集，即整数的基本指令集和一系列扩展指令集组成。这种灵活的组织方式使 RISC-V 指令集能够适应多种应用场景，小到简易嵌入式设备，大到高端多核处理器，都可胜任。

M：扩展整数乘法和除法指令，作用范围：整数

F：扩展了 IEEE 标准单精度浮点数运算指令，增加到了 32 个 32 位浮点寄存器。

D：用于浮点运算和转换，算术和逻辑运算，乘除运算，浮点类型转化。

~ 32 个 64 位浮点寄存器

范围：浮点数运算需要浮点数的统一，需要 32 位通用寄存器，需要浮点 CSR。

V：向量操作，向量并行运算向量读取和写回依靠代码的条件执行。

范围：需要专用向量寄存器，适用于向量化和管道化指令存储，并行化执行。

原子操作：内存原子操作，加载保留，条件存储。

范围：与内存一致性模型相匹配的多核体系中便于信息交流。

4. RV32I add opcode: 0110011 不同

RV64I addw opcode: 0111011

RV64I add opcode 0110011 相同

原因：①应用范围不同：addw 为 64 位半字指令，将结果低 32 位保留，高位进行符号扩展在寄存器中，而 add 指令在 32 位中为商 32 位相加，高位溢出在入寄存器中，64 位亦如此，因此 add 与 32 位 addw 操作数不同。



扫描全能王 创建

④灵活性，RV-32I add 和 RV-64I add 具有相同的操作模式，使用相同的操作码有利于提升指令的兼容性和灵活性

2) 要

在 RV-64I 中，addw 和 addiw 分别用于 32 位有符号数相加，和立即数加法，并将结果截断为 32 位有符号整数，因此，如果需要将其用于后续的 64 位计算，需要进行字节扩展。

5. HINT 空间是一组预留给实现者使用，用于定义特定的指令集序列，以提高处理器的性能和能效的空间

作用：为处理器实现者提供一组可供自定义的指令空间，以期处理器的性能和能效优化。

e.g.: HINT 指令可实现一些特殊操作：预取数据，跳转预测，分支预测等。

$$b: u_2 = -3$$

$$d_3 = 1$$

div：有符号除法 为0报错

divu 无 ~ —

rem：有符号数取余 余数符号和被除数一致

remu：无 ~ — 取正

11) ~~立即数寻址~~ ~~直接量寻址~~ 内存直接寻址

12) 寄存器寻址

13) 立即数寻址

14) 寄存器，直接寻址

15) 间接量寻址



扫描全能王 创建