

2023.3.14

1. CISC架构单个指令完成的任务量大且功能复杂、指令长度灵活
其优点为对编译器和程序存储空间的要求较低
缺点是硬件设计复杂，测试验证难度较高
RISC架构单个指令完成的任务量少且功能单一，指令长度相对固定
优点是硬件设计较为简单，适合利用流水线提升性能
缺点是对编译器设计的要求较高，程序代码密度较低
2. RISC-V基本指令集：RV32I、RV32E、RV64I、RV64E基础指令集
 - ① M标准扩展指令集 包含对两整数做乘除法的指令 从而简化低端硬件实现
 - ② A标准 ... 包含对寄存器执行原子性读写以支持不同线程之间的同步
 - ③ F 增加32个32bit的浮点寄存器和一个浮点控制状态寄存器csr
 - ④ D 增加双精度浮点计算的指令
 - ⑤ Q 增加四精度浮点计算的指令

4. 1) RV32I add指令操作数是0b10011

RV64I addw指令操作数是0b11011 与RV32I add不一样

RV64I add指令操作数是0110011 与RV32I add一样

因为RV64I是RV32I的超集，且RV64I中寄存器扩展到64bits

RV64I包括RV32I的所有指令 addw是在RV64I中新增的指令。

而为了让add指令能普适于RV32I、RV64I指令集且与addw区分

...采用add指令操作数一致而与addw不一致

2) 不需要。RV64I中addw和addiw指令最终会把符号结果截断为32位并把符号位扩展的结果写入x[rd]，所以不需要做额外的符号扩展。

5 HINT 指令是提示指令, 通常用于向微架构传达性能提示, 且其除推动 pc 以及任何可用性能计数器外, 并不改变任何体系结构可见的状态。具体实现可选择忽略这些提示编码

6 $\text{div } a2, a0, a1 \Rightarrow \text{reg}[a2] = -3$

$\text{rem } a3, a0, a1 \Rightarrow \text{reg}[a3] = 1$

DIV 除法: $\text{div } rd, rs1, rs2$ 即用 $x[rs1]$ 的值除以 $x[rs2]$ 的值向零舍入 (正数向下舍入, 负数向上舍入) 将这些数视为二进制补码输入 $x[rd]$ 。

REM 取余 $\text{rem } rd, rs1, rs2$ $x[rs1]$ 除以 $x[rs2]$ 向零舍入, 余数写入 $x[rd]$

1) $\text{jal } ra, 0x88$: 偏移量寻址

2) $\text{jalr } x0, ra, 0$: 寄存器间接寻址

3) $\text{addi } a0, a1, 4$: 立即数寻址

4) $\text{mul } a0, a1, a2$: 寄存器直接寻址

5) $\text{ld } a4, 16(sp)$: 偏移量寻址