

计算机组成原理
的挑战。
结构在功

1. 解: CISC 优点: 指令丰富, 功能强大

寻址方式灵活

以微程序控制器为核心, 指令存储器与数据存储器共享同一个物理存储空间, 性能强大

人们通
常效率
计设计
以通用
的析

缺点: 指令使用率不均衡

不利于采用先进结构提高性能

结构复杂不利于 VLSI 实现

RISC 优点: 结构简单, 易于设计

指令精简, 使用率均衡

程序执行效率高

缺点: 指令数较少, 功能不及 CISC 强大

寻址方式不够灵活

2. 解: RISC-V 基础指令集为 RV32I

扩展指令集: M 包含乘法、除法、求模、取余指令

F 单精度浮点指令

D 双精度浮点指令

A 原子操作指令, 比如比较并交换、读改写等指令

B 位操作指令

4. 解: (1) RV32I 中 m_add 和 RV64I 中 m_addw 具有不同 M 操作数, RV32I 中 m_add 为 0110011, RV64I 中 m_addw

为 0111011

RV32I 中 m_add 和 RV64I 中 m_addw 具有相同的操作数, 即 0110011, 这样设计可以增强指令集架构的灵活性

和灵活性, 可以在不同位宽的处理器上使用相同的译码和执行逻辑, 在实现不同位宽的处

处理器时尽可能地减少硬件设计和软件编写的工作量

(2) 需要, 因为 `addw, addiw` 只操作低32位的数据, 而丢弃高位的数据, 以此需先进行符号位扩展成64位才能进行^补64位计算

5解: `HINT` 是用于提供一些提示和调试信息的指令, 不会改变程序的执行结果, 但可以影响程序的性能和调试信息的收集。

有 `nop` 即 `no operation`, 表示不执行任何操作, 用于插入空指令

`fence` 用于控制内存访问的顺序和可见性

`ecall` 和 `ebreak` 用于系统调用和断点

`csrrw, csrrs, csrrc, csrrwi, csrrsi, csrrci`, 用于读取和修改控制/状态寄存器(CSR)

6解: `div a2, a0, a1` 后 `a2` 存值为 -3 , `rem a2, a0, a1` 后 `a2` 存值为 1

对于 `div`, 当 `a0, a1` 符号相同时 `a2` 为正数, `a0, a1` 符号相异时 `a2` 为负数

对于 `rem`, `a2` 的符号与被除数(即 `a0`) 符号相同

11解 (1) 偏移量寻址

(2) 偏移量寻址

(3) 立即数寻址

(4) 寄存器直接寻址

(5) 偏移量寻址