

嵌入式第四次理论课作业

陆畅 21307140118

1. 简要分析 Cisc 和 risc 架构各自的优势和劣势

Cisc: 复杂指令计算机，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要求较低

缺点：代码复杂，硬件设计复杂，测试验证难度较高

Risc: 精简指令集计算机，单个指令完成任务量少且功能单一

优点：硬件设计较简单，适用于利用流水线提升性能

缺点：对编译器设计要求高，程序的代码密度较低

2. RISCv 中的基本指令集是什么？列举五个常见的 RISCv 扩展指令集

并简要说明其作用和应用范围。

基本指令集：寄存器-寄存器型指令集

基本指令集

Base	Version	Status
RVWMO	2.0	Ratified
RV32I	2.1	Ratified
RV64I	2.1	Ratified
<i>RV32E</i>	<i>1.9</i>	<i>Draft</i>
<i>RV128I</i>	<i>1.7</i>	<i>Draft</i>

扩展指令集：

(1) 乘法和除法扩展(M 扩展)

用于将两个整数寄存器中的值进行相乘或相除

具体包含了 MUL、DIV、REM 等指令及对应于无符号数操作或 64 位操作相关的各种变体指令

(2) 原子指令扩展(A 扩展)

用于支持相同内存空间中的多个硬件线程间的同步

主要包含了对内存进行原子性读取、修改、写入的指令

(3) 单精度浮点扩展(F 扩展)

用于支持单精度浮点运算

添加了兼容 IEEE 754-2008 算术标准的单精度浮点运算指令, 以及

32 个 32 位宽的浮点寄存

器 $f_0 \sim f_{31}$, 和一个浮点控制与状态寄存器 $fcsr$

(4) 双精度浮点扩展(D 扩展)

用于支持双精度浮点运算

它依赖于 F 扩展, 需要与 F 扩展共同使用, D 扩展添加了兼容

IEEE 754-2008 算术标准的双精

度浮点运算指令, 并把浮点寄存器 $f_0 \sim f_{31}$ 拓宽到 64 位

(5) 四精度浮点扩展(Q 扩展)

用于支持四精度浮点运算

它依赖于 D 扩展, 需要与 D 扩展和 F 扩展共同使用, Q 扩展添加

了兼容 IEEE 754-2008 算术

标准的四精度浮点运算指令, 并把浮点寄存; 器的位宽扩展到 128

位

4. (1) 操作数不相同，操作数相同。分析这样设计是为了使得操作时不会引起混乱

(2) 当扩展太大，不能在 32 位固定长度指令格式中容纳时，最简单的方法就是加入自然对齐的 64 位指令。实现仍然必须支持 32 基本指令格式，但可以要求 64 位指令都在 64 位边界对齐，以简化指令取指，可在需要时，使用 32 位 NOP 指令来作为对齐填充。

为了简化标准工具的使用，64 位指令应当按照图 1.1 中描述那样进行编码。然而，实现可以为 64 位指令选择一个非标准的指令长度编码，同时保持为 32 位指令的标准编码。例如，如果不需要压缩指令，那么一条 64 位指令可编码为在它最前面 2 位设置 1 位或者更多位 0。

5. HINT 指令可以被记作 no operation 指令，让 core 进入低功耗状态，core 在执行这些指令的时候没有任何特定操作，core 功耗显著降低。

HINT 指令被设计成支持未来增加微体系结构提示，这些提示可能影响性能，但是不能影响体系结构状态。HINT 编码已经被选定，因此简单的实现可以忽略 HINT 编码，并将 HINT 指令作为常规指令执行，不改变体系结构状态。

例如，C.ADD 指令的目标寄存器如果是 x0，那么它是一条 HINT 指令，此处 5 位的 rs2 字段编码了 HINT 的细节。然而，一个简单的实现可以简单地把 HINT 当作一条目标是 x0 的加法指令执行，这时将没有效果（即 NOP 指令）。

6. $a_2:-3, a_3:1$ 规定：除法指令按照被除数与除数的正负关系得出，
 $a \% b = r$, $a > 0$ 则 $r > 0$, $a < 0$ 则 $r < 0$

11.

- 1) 偏移量寻址
- 2) 立即数寻址
- 3) 立即数寻址
- 4) 寄存器直接寻址
- 5) 寄存器间接寻址（偏移量寻址中偏移量为 0）