Date.

嵌入式第四周作业

1. 简要分析CISC和RISC架构各自的优劣势.

解：1) CISC 复杂指令集计算机，单个指令完成任务量大且功能复杂，指令长度灵活，如x86指令集即为CISC架构.

优点：对编译器和程序存储空间要求较低.

缺点：硬件设计复杂，测试验证难度较高.

2) RISC，精简指令集计算机，单个指令完成任务量少且功能单一，指令长度相对固定，如RISC-V，MIPS及ARM(移动常见).

优点：硬件设计较为简单，适合利用流水线提升性能.

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基本指令集是什么？列举5个常见的RISC-V标准扩展指令集并简要说明它们的作用和应用范围.

基本指令集：

RV32I：使用32位寄存器的基本32位整数指令

RV32E：只使用16个寄存器的基本32位指令，适用于低端的嵌入式应用.

RV64I：使用64个寄存器的基本64位整数指令.

2) 标准扩展指令集：

M：扩展了整数乘法和除法指令.

A：扩展了并发操作中的原子指令.

F：扩展了IEEE标准单精度浮点数运算指令，增加32个32位浮点寄存器.

D：扩展了双精度浮点数运算指令.

V：扩展了向量操作指令.

C：压缩指令扩展，将某些指令进行压缩，提高代码密度；

D：双精度扩展，扩展双精度浮点寄存器，计算指令、L/S指令.

I+C+M+F+A+D 缩写为"G".

4. 以RV32I中的add指令和RV64I中的addw指令均为32位整型加法指令，但具有不同的指令操作数(opcode)，但和RV64I中的add指令具有相同的指令操作数，是为了保持指令集的简洁性和一致性，同时提高编码和执行效率.

在RV64I中，addw和addiw指令的目标寄存器中存放的32位计算结果不需要进行额外的符号扩展就可以进行后续的64位计算.

因为RV64I指令集采用了扩展指令，使得所有的32位操作都能被自动进行符号扩展.

并存储在64位寄存器中，无论是addw还是addiw，其结果存放在目标寄存器中时已进行了符号扩展.

小补充：RV32中，add指令的操作数为32位，在RV64中，addw的操作数为64位，其中低32位表示操作数。

ADD：有符号加法指令；ADDW：低32位有符号加法指令；ADDI：有符号立即数加法指令。ADDIW：低32位有符号立即数加法指令。

5. 什么是RISV的标准指令集中存在的HINT指令空间？它有什么用？

解：它是一组用于给处理器提供信息的指令。HINT指令不是必须的指令，但可以提高处理器的性能和能效。由一个16位的opcod（操作码）和一个4位的funct3（功能码）字段组成。

作用：向处理器提示信息，以便优化处理器的性能和能效。包括提示信息可以包括：

① 告诉处理器何时可以进行指令重排列。

② 提醒处理器当处理器等待某事件发生时，可以进行其他任务，以避免浪费其计算资源。

③ 告诉处理器何时可以休眠，以减少功耗。

总：告诉其如何更好地利用计算资源。

第：是一组用于提供提示和建议的指令。告诉CPU如何最优化执行代码。

6. 解：a1,a2为-3

DIV指令执行有符号除法操作，被除数和除数的符号必须相同，如果被除数和除数中有一个为负数，则结果为负数；如果都为正数，则结果为正数。

21 a3. 为1

当两数正负号一样时，结果为正；

当有一个除数时，结果与被除数符号保持一致。

11. 写出以下指令的使用的寻址模式。

例子：1) jal ra, 0x88

JAL指令使用的是PC相对地址模式，偏移量为0x88。

2) jalr x0, ra, 0

JALR指令使用的是基址加偏移量寻址模式，即ra寄存器的值加上偏移量0。

3) addi a0, a1, 4

ADDI指令使用的是立即数模式，即a1寄存器的值加上立即数4。

(4) mul a0 a1 a2

MUL使用的是寄存器模式，即a1寄存器和a2寄存器的值相乘

5) ld a4, 16(sp)

LD是基址加偏移量模式，即栈指针寄存器SP的值加上偏移量16。