计算机组成原理模拟试题参考答案

1. 单选题

CCDCA DBCCA CCBBA

1. 填空题

1. 答：控制器，存储器，输入设备，输出设备

2. 答：（1）8位 （2）23位

3. 答：32,32,0

4.答：主存群号，标记（或Cache行号），块内地址，或者主存组号，组内块号，块内地址

5.答：**统一编址方式（内存映射方式），独立编址方式（使用专门的I/O指令方式）**

**6. 答：Mem，Wr**

1. 名词解释

1. 基准程序（benchmarks）

答：专门用来进行性能评价的一组程序，不同的机器运行相同的基准程序可比较它们的运行时间。

2. 对阶（浮点加减运算）

答：浮点数加减运算的对阶是使两数的阶码相等(小数点实际位置对齐，尾数对应位权值相同)。

3. **CISC**

答：即复杂指令集计算机，将复杂指令加入到指令系统中，以提高计算机的处理效率。

4. 向量中断

答：直接依靠硬件来获得中断服务程序的入口地址的这种中断称为向量中断。

5. 结构冒险

答：同一个部件同时被不同指令所用，即使用硬件资源时发生了冲突。

1. 简答题
2. IEEE754单精度浮点数标准中非规格化数是如何定义的？尾数的隐藏位是多少？阶码的真值是多少？

答：非规格化数的阶码为全0，尾数为非0值。尾数的隐藏位为0，阶码的真值为-126.

2. 下列MIPS指令中分别包含哪些寻址方式？指令的功能是什么？

（1）beq $s1，$s2，addr

（2）lui $t1,100

（3）j 100

（4）lw $s3，20($t1)

答：（1）寄存器寻址，PC相对寻址。指令的功能是比较寄存器$s1和$s2，如果相等则转移。

（2）寄存器寻址，立即数寻址。将寄存器t1的高16位设置为100，低16位为0.

（3）伪直接寻址（或页面寻址）。跳转指令，跳转到PC指定的指令处，PC的形成是，高4位保持不变，将100左移两位（或乘以4）送入PC的低28位。

（4）寄存器寻址，基址寻址。从存储器中取一个字单元内容送入s3寄存器中，存储器地址为：将常数20加上寄存器t1的内容。

3. DMA中常用的数据传送方式有哪三种？简述它们的传送方法。

答：有CPU停止法(成组传送)、周期挪用(窃取)法(单字传送)和交替分时访问法。

CPU停止法：DMA传输时，CPU脱离总线，停止访问主存，直到DMA传完一块数据。

周期挪用法：DMA传输时，CPU让出一个总线事务周期，由DMA控制器控制总线来访问主存，传送完一个数据后立即释放总线。

交替分时访问法：将每个存储周期分成两个时间片，一个给CPU，一个给DMA，这样在每个存储周期内，CPU和DMA都可访问存储器。

4. 答：

中断响应是指主机发现外部中断请求，中止现行程序的执行，到调出中断服务程序这一过程。中断响应是又CPU硬件完成的，它完成工作包括关中断、保护断点和程序状态、识别中断源。

中断处理是指执行相应中断服务程序的过程。中断处理是由软件完成的，它就是执行相应的中断服务程序，不同的中断有不同的中断服务程序，大致包括准备阶段、具体的中断处理阶段和恢复阶段。

1. 计算与分析题

1. 一个高级语言编写的程序被两个不同的编译器编译生成两种不同的指令序列X1和X2，在时钟频率为2GHz的机器上运行，目标指令序列中用到的指令类型有A、B、C和D四类。四类指令在机器上的CPI和两个指令序列所用的各类指令条数如下表所示。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 指令类型 | A | B | C | D |
| 各类指令的CPI | 1 | 3 | 4 | 2 |
| X1的指令条数 | 5 | 3 | 2 | 2 |
| X2的指令条数 | 4 | 5 | 2 | 3 |

问：X1和X2各有多少条指令？所含的时钟周期数各为多少？CPI各为多少？执行时间各为多少？小数保留到小数点后1位。

解：X1有12条指令，所含的时钟周期数为5×1+3×3+2×4+2×2=26，CPI为 26/12=2.2,执行时间为26/2G = 13ns。

X2有14条指令，所含的时钟周期数为4×1+5×3+2×4+3×2=33，CPI为 33/14 =2.4,执行时间为33/2G = 16.5ns。

2. 将十进制数-135.59375转换成IEEE754 的32位标准浮点数格式，分别写出其二进制数表示和16进制数表示。

解：

135.59375=(10000111.10011)2

规格化尾数=1.000011110011，阶码e=7，E=7+127=134=（10000110）2

尾数符号S=1，尾数M=000011110011

32位二进制格式：1 10000110 00001111001 100000000000

16进制格式：C3079800

3. 若指令“ORI rt，rs，immediate”的编码格式如下：

|  |  |  |  |
| --- | --- | --- | --- |
| [31:26] | [25:21] | [20:16] | [15:0] |
| OP | rs | rt | immediate |

其中，OP为001101B，rs为8，rt为15，immediate为200。

将该指令分别按照小端方式和大端方式存储在2000号开始的四个字节存储单元中。要求用十六进制数表示，结果直接填入表中。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 单元地址 | 2000 | 2001 | 2001 | 2003 |
| 小端方式 |  |  |  |  |
| 大端方式 |  |  |  |  |

解：指令码的二进制表示：001101 01000 01111 00000000 11001000

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 单元地址 | 2000 | 2001 | 2001 | 2003 |
| 小端方式 | 35 | 0F | C8 | 00 |
| 大端方式 | 35 | 0F | 00 | C8 |

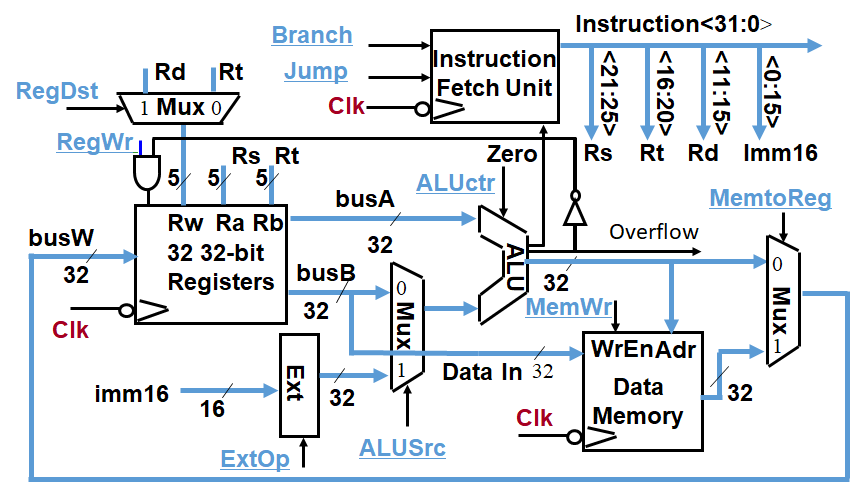
4. 设有下图所示单周期MIPS数据通路，试分别指出下列指令在该数据通路中执行时，各控制信号的取值是什么？说明：有效为1;无效为0; ALUctr可为add,sub,addu,subu,or,and;无影响为x.

（1）andi $s1,$s2,100

（2）sub $8,$4,$5

（2）lw $t1，20($s2)

（3）beq $s1,$t1,200



|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **指令** | **RegDst** | **ALUSrc** | **ALUctr** | **MemWr** | **ExtOP** | **RegWr** | **MemtoReg** | **Branch** | **Jump** |
| **andi** |  |  |  |  |  |  |  |  |  |
| **sub** |  |  |  |  |  |  |  |  |  |
| **lu** |  |  |  |  |  |  |  |  |  |
| **beq** |  |  |  |  |  |  |  |  |  |

答：

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **指令** | **RegDst** | **ALUSrc** | **ALUctr** | **MemWr** | **ExtOP** | **RegWr** | **MemtoReg** | **Branch** | **Jump** |
| **andi** | **0** | **1** | **And** | **0** | **0** | **1** | **0** | **0** | **0** |
| **sub** | **1** | **0** | **Sub** | **0** | **X** | **1** | **0** | **0** | **0** |
| **lu** | **0** | **1** | **Addu** | **0** | **1** | **1** | **1** | **0** | **0** |
| **beq** | **X** | **0** | **Subu** | **0** | **X** | **0** | **x** | **1** | **0** |

5. 假定采用“取指、译码/取数、执行、访存、写回”五级流水线方式，对于以下指令序列（1）哪些指令对发生数据冒险？(2)哪些可以采用转发技术解决？（3）哪些不能用“转发”技术解决？需要采用什么方法解决？

add $s3, $s1, $s0

add $t2, $s0, $s3

lw $t1, 100($s3)

add $s1, $t1, $s3

答：

（1）发生数据冒险的指令对有：第1条与第2条，第1条与第3条，第1条与第4条，第3条与第4条。

（2）第1条与第2条，第1条与第3条。

（3）第1条与第4条采用将寄存器写口和读口分别控制在前、后半个时钟周期内操作；第3条与第4条采用lw指令后插入空操作指令。

6. 某计算机存储器按字节编址，虚拟地址空间为16MB，主存（物理）地址空间大小为1MB，页面大小为4KB。Cache数据区容量为4KB，主存与Cache交换数据单位为256B，采用2路组相联映射方式。系统运行到某一时刻，页表的部分内容和TLB的内容如下表所示，表中的页框号及标记字段为16进制数。

页表 TLB

|  |  |  |
| --- | --- | --- |
| **装入位** | **...** | **页框号** |
| **1** | **...** | **06** |
| **1** | **...** | **04** |
| **1** | **...** | **15** |
| **1** | **...** | **02** |
| **0** | **...** | **-** |
| **1** | **...** | **2B** |
| **0** | **...** | **-** |
| **1** | **...** | **32** |
| **...** | **...** | **...** |

|  |  |  |  |
| --- | --- | --- | --- |
| **组号** | **有效位** | **标记** | **页框号** |
| **0** | **0** | **-** | **-** |
| **1** | **001** | **15** |
| **0** | **-** | **-** |
| **1** | **012** | **1F** |
| **1** | **1** | **013** | **2D** |
| **0** | **-** | **-** |
| **1** | **008** | **7E** |
| **0** | **-** | **-** |

（1）虚拟地址共有几位？哪几位表示虚页号？物理地址共有几位？哪几位表示页框号？

（2）cache每一行中的标记有多少位？总容量是多少？（每行1位有效位V）

（3）虚地址001C60H和024BACH所在的页面是否在主存中？若在，它们对应的物理地址是多少?这两个单元映射到cache的哪一组？

**答：**

**（1）虚地址空间16MB=224，所以，虚地址共有24位。**

**页面大小为4KB=212B，页内地址12位，故虚页号有24-12=12位地址，即虚地址的高12位。**

**因为主存空间为1MB=220B，所以物理地址共20位。**

**物理地址与虚地址的页内地址一样同为12位，故页框号为20-12=8位，即物理地址的高8位为页框号。**

**（2）cache总行数=4KB/256B=16，组数为16/2=8, 物理地址中间3位为组号，高20-3-8=9位为装入cache行的标记部分，即cache行的标记部分为9位。**

**Cache总容量=16\*（1+9）bit+4KB=32928bit=4116B=4.0195KB （三个答案都可以）**

**（3）由于TLB映射方式为4路组相联，共有2组，故虚页号中的最低1位(bit)为组号，高11位(bit)做标记，即虚页号12位被分为：11位标记+1位组号。**

**对于虚地址001C60H，其虚页号为001，高11位为0，最低位=1，查TLB的1组中有效位=1的所有行，没有标记为0的行，故TLB缺失，需查页表。页表按虚页号排序，001号虚页为第二行，其有效位为1，页框号(物理页号)为04，所示该地址单元已在主存中，物理地址=04C60H。该地址映射到cache时，低8位为块内地址，中间3位为组号100，即映射到第4组。**

**对于虚地址024BACH，虚页号为024H，其高11位为000 0001 0010B=012H，查找到TLB中有“标记”=012H的行，其“有效位”=1，实页号=1FH，故该虚地址单元在主存的物理地址=1FBACH。该物理地址中间3位就是映射的组号为011，即第3组。**

7. 设某计算机有5级中断，中断响应优先级为1>2>3>4>5，而中断处理优先级为1>4>5>2>3。要求：

（1）设计各种中断处理程序的中断屏蔽位(假设1为屏蔽，0为开放)；

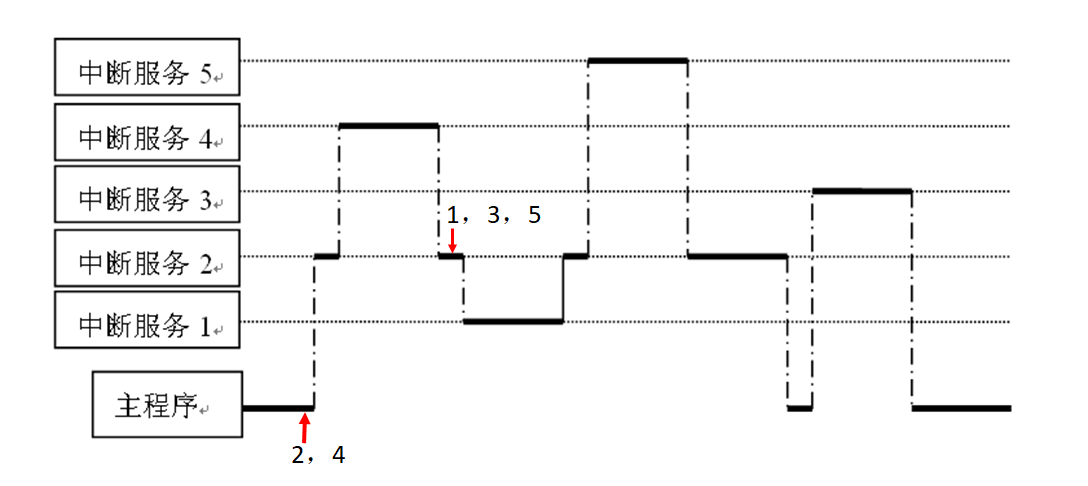
（2）若在运行主程序时，同时出现第2、4级中断请求，而在处理第2级中断过程中，又同时出现1、3、5级中断请求，试画出此程序运行过程示意图。

解：**各级中断处理程序的中断屏蔽字：**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **中断程序级别** | **中断屏蔽字** | | | | |
| **1级** | **2级** | **3级** | **4级** | **5级** |
| **第1级** | **1** | **1** | **1** | **1** | **1** |
| **第2级** | **0** | **1** | **1** | **0** | **0** |
| **第3级** | **0** | **0** | **1** | **0** | **0** |
| **第4级** | **0** | **1** | **1** | **1** | **1** |
| **第5级** | **0** | **1** | **1** | **0** | **1** |

程序运行过程又分成两种情况：

（1）假设第2级中断服务程序中**开中断**后第1条指令执行时未出现1、3、5级中断请求，即开始2级中断服务程序时排队队列中只有4级中断。4级中断处理结束后才发生1、3、5级中断请求。



（2）假设在第2级中断服务程序开中断前就出现1、3、5级中断请求，即2级中断服务程序中开中断后1、3、4和5级有中断请求。根据2级中断的屏蔽字，3级被屏蔽，优先级响应队列中有1、4、5，1级得到响应，在1级中断中屏蔽了所有的中断，1级中断处理完后回到2级，响应4级。在4级中根据其屏蔽字3级和5级被屏蔽，4级处理完后回到2级，响应5级，在5级中根据其屏蔽字3级被屏蔽，5级处理完后回到2级。2级处理完回到主程序，最后响应3级，3级处理完后回到主程序。

