**2022期末回忆卷**

选择题  
●给机器码翻译成指令  
●给32位大立即数地址取出其中内容（lui+lw）  
●浮点数加法过程：0.12345\*10^2 和 0.12345\*10^(-2) 对齐后 10 的幂次是多少  
●四个选项选对的：  
○PTE 数=物理页数  
○增加 associativity 可以减少 capa miss  
○async bus 的概念  
○内存结构中耗时最多的是最上层  
●IEEE 754：1.355-2.105 用单精度浮点数表示  
●单周期的时候时钟周期完不成哪个工作(读并写内存)  
●TLB 有 tlb 的时候先看谁：tlb pgtbl cache phymem  
●RAID 是干啥的  
●cache 增大 block size 可以减少哪种 miss  
●write through 指什么  
●6 个 block，每个 block 4words，求 840 对应的 block 号  
  
选择

1. 给了指令机器码，解释是什么指令

bne x0, x0, -20

3.标准化的浮点数的最小值

5. WAW RAW WAR判断

8. 提升block size 可以减小什么？

cache miss hit time compulsory miss

9. 中断异常相关

10. polling，interrupt，DMA对CPU效率的影响从小到大排序

大题

1.流水线CPU，计算CPI，

包含指令lw sw add branch 外加另一种（不记得了）

有IMEM penalty DMEM penalty lw stall和branch not taken

还给了时钟频率和MEM penalty = 75ns

2. 流水线cpu数据通路

（1）sw指令数据通路信号的判断

（2）没有hazard优化的PipCPU，给了七条指令的一段程序，需要通过插入NOP避免hazard，让我们写出插入之后的程序

（3）补充数据通路，用bypassing避免arithmetic指令的RAW竞争

（4）给出新数据通路中每条指令操作的rs1，rs2数据的来源

（5）求改完之后的时钟周期数

（6）通过scheduling来进一步优化这段程序，减小clock cycle

3. 汇编

（1）用最少的指令实现 x20>x11 | x20 < 0的跳转

（2）用branch跳转范围不够，怎么扩大范围

（3）把C程序翻译成汇编程序

4. cache

（1）有cache和TLB，计算物理地址，虚页地址内部各段的bit数量；计算cache，tlb内部分段的bit数

cache：4KB， blocksize128B，二路组相联

TLB：entry512个，二路组相联

虚页大小8KB

物理地址32bit

虚页地址54bit

（2）采用LRU策略，按顺序以以下地址访问cache：0，100，200，300，1024，2048，4096，250，100。问哪几次访问hit，哪几次miss，cache中数据被替换掉几次，hit rate是多少

（3）接上问，问最终在cache里的valid数据有哪些  
  
(1) 把这6个数排序：在原码、补码、符号表示、IEEE754 下的 0xF0000000，在原码、补码下的 0xFFFFFFFF  
(2) 不用别的寄存器，交换 x10 和 x11  
(3) 书上IndexOutOfBound 那个例子，判断数组越界

2 写汇编

void main(char \*s,int \*n){

char c,ch;

c = '3';

ch = '5';

\*n = replace(\*s,c,ch);

}

int replace(char \*u,char c,char ch){

int i = 0;

while(u[i]!=0){

if(u[i]==c){

u[i]=ch;

break;

}

i++;

}

return i;

}

3 cache(不确定完全一致)  
(1) 32B 的 cache，一个 entry 8B，直接映射，write through+write around，给出如下访问：0 16 48 8 56 16 8 56 32 0 60，问每一次的index, tag, 是否hit  
(注意 write around)  
(2) 128B 的 cache，一个 entry 16B，2路组关联 LRU，write back，给出访问：64 32 64 0 112 64 128 48 240 0，问的跟上面一样，外加最后的dirty block number

4 虚拟内存  
页表大小，以及给定情况问 TLB 能不能容纳 8MiB 内存

5 流水线  
(1)先问了 sub 的时候除了 ALUOp 以外各个信号的值  
给了一段代码，应该是  
sub x5,x7,x11 ld x13, 0(x5) ld x7,0(x2) add x13,x5,x13 sd x13, 0(x5)  
问：  
(2) 如果没有 forward 和 hazard detection，在代码插 nop 使其能正确  
(3) 如果有 forward 没有 hazard detection，干(2)的事  
(4) 如果有 forward，写前7个时钟周期 ForwardA 和 ForwardB 的信号值

**2021-2022**

选择

（1）给机器码判断汇编指令

（2）如何正确偏移 考lui lw ld（主要考ld的imm最多12位...吧我猜）

（3）增大blocksize能减少哪一种miss

（5）简答题

1 不用别的寄存器交换两个寄存器的值 用最少的指令判断if(a≥b且a<0)的条件

2 C语言转汇编 把字符串中的某一个字符转换成另一个字符的操作

3 cache 给地址判断miss/hit （1）直接映射 （2）两路组相联

4 pagetable&TLB （1）计算pagetable的物理内存 （2）pagesize为4KB，如果TLB只有64个entry，能不能支持一个至少访问8MB内存的程序。如果不能，pagesize需要多大。

5 流水线 （1）单周期写控制信号 （2）流水线没有前递和Hazard加空指令 （3）有前递但没有Hazard加空指令 （4）每一个时钟周期的ForwardA和ForwardB

（1）很基础的一条指令写出寄存器的值 （2）3条指令写出寄存器的值 （3）16进制转8进制（（4）十进制小数转换成单精度浮点数

1.5 没记错的话应该是给两个CPU运行的频率GHz和CPI（clock per instruction）计算比较性能

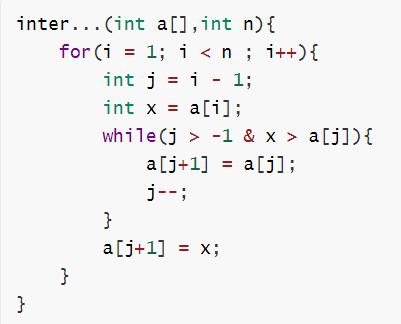
（6）2个指令翻译成机器码，1个反过来，考了两个跳转jal+SB类型指令，还有一个R类型 **第二题** 1.2路组相连cache （1）一些有关cache的计算，就是block size，还有表的大小这种； （2）给你一系列字节地址让你写出相应的访问中命中率以及替换的块数； （3）让你写出最后的时候里面的cache的情况 2.虚拟内存的 （1）计算page table的大小差不多（有点忘了） （2）IO和CPU交互的三种方式，就只用知道是啥就行，它让你简单讲讲怎么工作的

第二大题关于 page table 的具体描述是这样的： 假设虚拟地址为 43 位，物理内存为 16GiB ，page 大小是 4KiB ，page table entry 的 size 是 32bit ，问 page table 需要多大的物理内存

**第三题** 两道指令的题 1.没有处理data遇险的情况下流水线中指令的执行结果（就是考考你对流水线几个阶段的认识）不仅没有考虑数据竞争，还没有考虑控制竞争（就是读取beq指令之后先向下执行了两条指令，在beq的Ex阶段判断出要跳转之后，再进行beq的跳转操作）

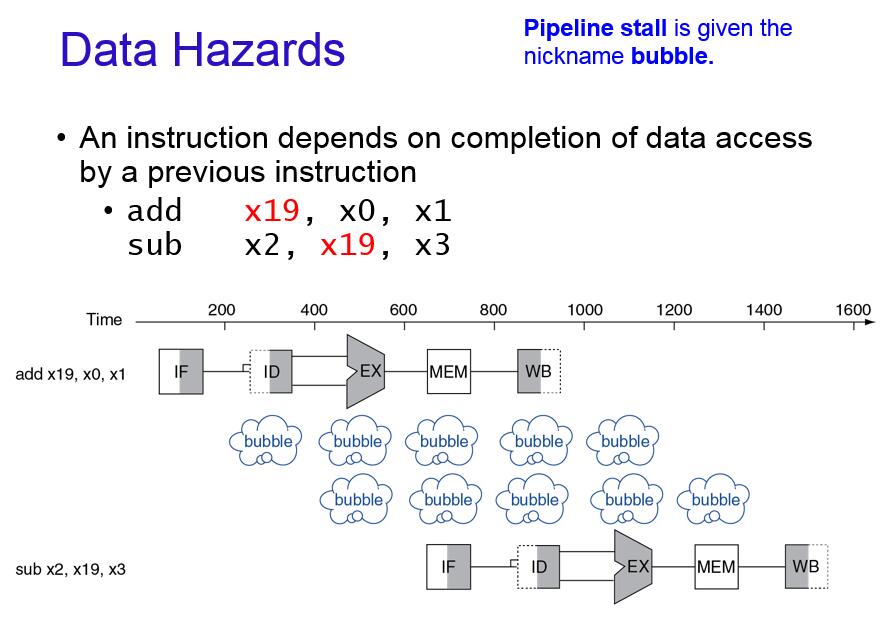
**以下是引用18楼：用户27rabbit在2021-07-04 17:50:32的发言：[>>查看原帖<<](https://www.cc98.org/topic/5114223/2" \l "8" \t "_blank)** 第三大题的第一个题，需要注意的是，我们并不是在 EX stage 决定 beq 是否跳转。跳转地址的计算是在 MEM 阶段完成的，所以会有一条指令被执行两次，第一次是流水线顺势执行下去了，第二次是 beq 跳转导致的。

上面这个题看起来还挺有意思的！ 2.给你一段C代码让你写对应的汇编



**第四题** 流水线 好像还是关于遇险的题目

（1）画出数据遇险可能发生的时候采取的措施，是让画图，就多周期每条指令的执行情况（要有bubble的那种） 图就是这种的。



（2）写出在第4周期各个模块的执行的执行（就是在同一时间每个部分处理的指令）//就是上面的图的一列 （3）让你画出数据遇险在图中的解决方法 （4）一个计算性能的题目： 第四题第四问具体是这样的： 加入 forwarding 后，周期从 200 变成了 210 ，但是 nop 的数量从 1.5 变成了 1.04 ，问提高了多少

**2019-2020春夏 计算机组成 考点回忆**

1. 填空 25%

-32.6 的 float 表示

1. Memory Hierachy 25%
   * direct-mapped cache
     + 参数计算
     + 一串访问序列
       - how many blocks replaced?
       - 列出最后 cache 中 valid 的块
   * virtual memory: 给 vAddr长度 pAddr长度 页粒度
     + page table 大小
     + TLB configured as 2-way associative; 256 entries. Draw a figure to illustrate address mapping from virtual to physical
2. Program 30%
   * 一段自修改的代码，跟踪寄存器的值. 思考
     + 如何改 addi 的 imm16
     + 如何改 bne 的 offset
   * 题源 PPT ch2
     + void sort(int v[], int n) // 冒泡排序
     + int compare(int a, int b) // a >= b
     + void swap(int v[], int k) // 交换 v[k] 和 v[k+1]
     + 随便写的话不难，然而开头阴阳怪气地来了一句"Be sure to handle frame pointer and stack pointer properly"，并且
     + 第二小题要求追踪几次函数调用的堆栈。比较迷惑，出题者意图应该是想实现类似stdcall的调用规范
3. Multicycle CPU implementation
   * (several kind of exceptions, interrupts) Identify states where these exceptions can be identified
   * 一段关于中断与异常扩展的描述，但在一般的EPC, Cause寄存器外，又整了个 "Error Trap" 和 "ErrorPC"
     + 扩充数据通路和控制信号来支持"Error Trap"
     + eret FSM