

# 计算机系统基础期末易错点和注意点记录

2019年1月4日 12:43

1. 比较难分析cache命中率的时候，用命中总次数除以总次数，一定没错！
2. 计算过程务必带上单位！以防因为单位漏换算等出错！
3. 计算数组有关cache命中率的时候，请先变成主存块分析（利用同余等思想）
4. 注意审题！
  - a. 直接相联？全相联？组相联？映射方式千万不能看错了！
  - b. LRU？随机替换？
  - c. 全写法？回写法？
  - d. 字节编址？
  - e. cache每一行=有效位+(脏位)+标记位+数据位
  - f. 页表每一项=有效位+修改位+使用位+存取方式位+物理页号位
  - g. TLB快表还要在页表总位数的基础上加上标记位
  - h. 内存条传输宽度！**传输宽度**/位平面数>1决定有片选信号位（传输宽度这个数据有用滴！如果>1表明一次要读取多个芯片，需要片选信号，且此时交叉编址）
5. 连续编址：片选信号在高位；交叉编址：片选信号在低位
6. 磁盘有4个盘片：意味着有8面记录信息！
7. 首地址按4字节对齐：意味着重定位以后首地址按4字节对齐就行了！
8. 磁盘的平均等待时间=磁盘旋转一周所需事件/2！
9. 弱符号放在.bss节当中，也即.data节当中没有给它分配空间，如册P152（1）（2）
10. 重定位之前值的计算？看看绑定地址和实际给的地址之间的偏差，如buf[1]重定位就应该是0x00000004，因为和重定位的&buf差了4字节
11. 只有全局符号才分！强！弱！
12. TLB、页表、cache一定要记得有效位为1？有标记位记得标记位相同？先比较这两！
13. 注意“指令I的执行过程中会不会发生XX缺失”要考虑指令是否缺失和数据是否缺失！
14. printf("Hello,World!\n");语句当中的字符串在hello.o文件中的.rodata节，在可执行目标文件的只读代码段中
15. n路组相联，如果采用LRU替换算法的话，需要另外的**log(2)n**位LRU位
16. typedef、字符串常量都不算“符号”！符号只有全局、外部、本地三种
17. 注意符号的“定义”和“声明”的区别！
18. 看清题目！问的是多少位，不是多少Byte！血亏
19. 计算题要写清过程，不要因为没有过程再被白白扣分了.....5555
20. 无符号数零拓展（z）、有符号数带符号拓展（s）
21. 过程调用前后有重要的语句？pushl、popl
22. 8位补码定点整数1001 0101左移1位的值为（**溢出**）
23. test指令不改变通用寄存器内容，只改变EFLAGS寄存器内容

24. 只能用32位寄存器作为地址所在寄存器
25. 小端、大端法存储
26. Linux: 只有2、4两种字节对齐方式; Windows: 如果有long、double类型则8字节对齐
27. 字符串末尾有一个\0! ! ASCII码是00H
28. 填代码必要时可以加上显式类型转换以实现无歧义 of 的无符号/带符号数右移
29. ELF可执行目标文件中的程序头表用来描述节与虚拟地址空间中的存储段的映射关系!
30. `int *bufp1=&buf[1]`的对bufp1的重定位在.rel.data节中 (因为它是全局变量, 并且这里是对bufp1的定义和初始化); 但是`bufp1=&buf[1]`对bufp1的重定位在.rel.text节中, (因为这里是对bufp1的引用, 用一个movl指令实现, 在代码的.text节当中)
31. 即使是引用全局变量, 如果这个全局变量已经初始化, 那么它也应该在当前.o文件的.data节当中
32. 分析缺页的时候务必分别考虑指令和数据会不会缺失!
33. 中断向量地址是指中断服务程序入口地址的地址
34. 硬件实现数据交换——DMA
35. 直接控制硬件: 设备驱动程序和中断服务程序
36. 不确定就翻书!
37. 加油!
38. 你最棒!