

计组书面作业第三单元

1. 某磁盘的转速为 10000 转/分，平均寻道时间是 6 ms，磁盘传输速率是 20 MB/s，磁盘控制器延迟为 0.2 ms，读取一个 4 KB 的扇区所需的平均时间约为 ()。

A. 9 ms B. 9.4 ms C. 12 ms D. 12.4 ms

$$6ms + 1/10000 * 60 * 1000/2ms + 4K/20M * 1000ms + 0.2ms = 6 + 3 + 0.2 + 0.2 = 9.4s$$

B

▼ 讲义例子

硬磁盘访问时间

- ▶ 磁盘访问时间 = 寻道时间 + 旋转延迟 + 传输时间 + 磁盘控制器延迟
- ▶ 举例：
 - ▶ 平均寻道时间 = 12ms;
 - ▶ 旋转速度 = 5400rpm
 - ▶ 磁盘控制器延迟： 2ms
 - ▶ 传输速度 = 5MB/s
 - ▶ 扇区大小 = 512 bytes
 - ▶ 读取一页（8KB）需要多长时间？
- ▶ 旋转延迟：平均旋转延迟应为磁盘旋转半周的时间。
- ▶ 旋转1周 = $1/5400$ minutes
= 11.1ms => ½ 周： 5.6 ms
- ▶ 读1个扇区时间 = 12ms + 5.6ms + .5K/5MB + 2ms
= 12 + 5.6 + .1ms + 2ms
= 19.7 ms
- ▶ 读1页的时间=
= 12 ms + 5.6ms + 8K/5MBpersec + 2ms
= 12ms + 5.6ms + 1.6ms + 2ms
= 21.2 ms

39

2. 下列关于虚拟内存, 描述正确的有 ()。
- I. 增加 TLB 可以提高虚拟内存访问性能
 - II. 虚拟内存可以提高存储访问性能
 - III. 多级页表可以节省内存页表占用空间
- A. 仅 I B. I 和 III C. II 和 III D. 全部都正确

B

3. 简答并给出理由: 若全相联映射和组相联映射使用 LRU 替换算法, 全相联映射的缓存命中率一定不低于组相联映射和直接映射吗?

在一些极端情况下, 组相联和直接映射命中率可能更高, 比如我们可以构造这样一个例子, 缓存行个数为4, 对于全相联映射, 访问 A, B, C, D, E, A 的命中率为0, 对于组相联映射, 假设这是一个 2 路 2 set 的组相联缓存, B, C, D, E 被分配到同一个 set 下, A 被分配到另一个 set 下, 则访问会命中一次。同理, 对于直接映射, 假设 D, E 在同一个 set 下, 其他的地址在不同的 set 下, 那么第二次访问A也会命中

但除了这样的极端例子外, 一般来说不低于, 相比组相联和直接映射, 全相联缓存更加灵活, 因为它可以将一个主存块放置在任何一个缓存行中, 而不需要受到固定的组或索引的限制。相比之下, 组相联映射和直接映射都具有更多的限制, 可能导致某些索引下的内存块始终为零而有些索引下的内存块更经常地被替换, 从而降低缓冲命中率

4. 简答并给出理由: 相比于不做 RAID 的方式, RAID1 的读写性能提升接近两倍吗?

RAID1的读性能可以提高两倍, 但写性能由于每次都要同时写两次, 所花时间可以理解为两次写入的较大者, 所以写的性能并不会提高

5. (给出计算过程) 某计算机系统的内存系统中, 已知 Cache 命中访问时间为 20ns, Cache 缺失访问时间为 100ns。CPU 执行一段程序时, CPU 访问内存系统共 5000 次, 其中缺失次数为 450 次。那么 CPU 访问内存的平均访问时间是_____。

平均访问时间 = 命中率 \times 命中访问时间 + 缺失率 \times 缺失访问时间 = $(1 - 450/5000) \times 20 + 450/5000 \times 100 = 27.2ns$

6. (给出计算过程) 为了存储 512GB 的数据, 使用 RAID0 的方式, 所占用的磁盘空间大小为____, 使用 RAID6 的方式 (4+2), 所占用的磁盘空间大小为 ____。

RAID0: RAID0没有校验位, 因此磁盘占用空间大小就是512GB

RAID6: 实际和校验存储4:2, 所以是 $512GB \times 6/4 = 768GB$

7. 现假设一计算机系统:

- 采用 12 位的虚拟地址;
- 采用 12 位的物理地址;
- 页面大小为 256 字节 (256 B);
- TLB 是 2 路组相连, 共有 8 个表项;
- Cache 是 2 路组相连, 共有 8 个大小为 4B 的 Cache Lines。

Page Table					
VPN	PPN	Valid	VPN	PPN	Valid
0	1	1	8		
1			9	5	1
2	5	0	A		
3	1	0	B	4	1
4			C		
5			D		
6	B	0	E	3	1
7	C	1	F		

TLB			
Index	Tag	PPN	Valid
0	3	0	1
1			
	3	2	1
2			
	2	F	1
3			
	1	F	0

- (1) 下面展示了一个虚拟地址的格式, 请标出以下这些域: VPO (virtual page offset)、

VPN (virtual page number)、TLBI (TLB index)、TLBT (TLB tag)。(如果这些域不存在, 则无需标明)。

11	10	9	8	7	6	5	4	3	2	1	0

- (2) 下面展示了一个物理地址的格式, 请标出以下这些域: PPO (Physical Page Offset)、PPN (Physical Page Number)、CO (Offset within the Cache Line)、CI (Cache Index)、CT (Cache Tag)。(如果这些域不存在, 则无需标明)。

11	10	9	8	7	6	5	4	3	2	1	0

- (3) 图中展示了某一时刻 TLB 和页表的部分状态, **图中所有数字均为 16 进制**, 已知对于此时的 TLB 和页表:

- 访问如下的虚拟地址时, TLB 命中: 0x48A, 0x1EA, 0xE0F, 0xB5A, 其中 0x48A 对应的物理地址为 0xA8A, 0x1EA 对应的物理地址为 0xEEA;
- 访问虚拟地址 0xF5A 时, TLB 缺失, 查找页表得到对应的物理地址为 0xB5A;
- 访问虚拟地址 0x88A 和 0x56F 时, 发生缺页。

请利用上述信息以及图中部分已知的 TLB 表项和页表项, 将图中所有空缺的部分补充完整。



清华大学
Tsinghua University

①

11	10	9	8	7	6	5	4	3	2	1	0
VPN						VPO					
TLBT				TLBI							

②

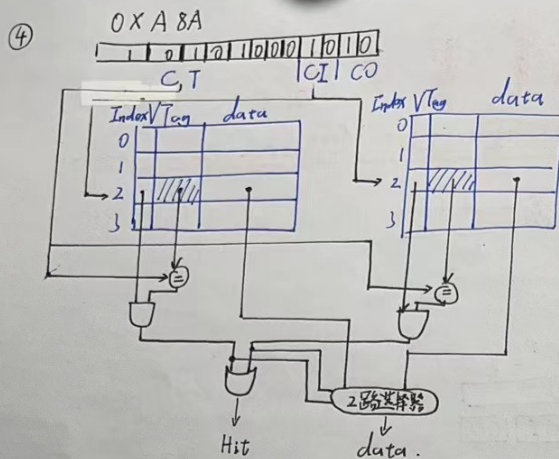
11	10	9	8	7	6	5	4	3	2	1	0
PPN						PPO					
CT				CI CO							

③

Page Table						TLB			
VPN	PPN	Valid	VPN	PPN	Valid	Index	Tag	PPN	Valid
0	1	1	8		0	0	3	0	1
1	E	1	9	5	1	0	1	A	1
2	5	0	A	F	1	1	0	E	1
3	1	0	B	4	1	1	3	2	1
4	A	1	C	0	1	2	3	3	1
5		0	D	2	1	2	2	F	1
6	B	0	E	3	1	3	2	4	1
7	C	1	F	B	1	1	1	F	0

1 2 10

- (4) 已知访问物理地址 0xA8A 时发生了 Cache Hit，请画出 Cache 的结构示意图并标出该次 Cache Hit 在示意图中的位置，并以此访问为例简述查找 Cache 的过程。



CT = 01010100 CI = 10 CO = 10

首先会根据 CI 去查两路中 Index 为 2 的 cache line, 然后和 CT 与蓝色阴影线一相等, 从而使 Hit 信号为 1, 并输出其中的 cache line, 然后再根据 CO 查对应的字节.

- (5) 如果某虚拟地址访存发生页面缺失 (page fault), 请简述页面缺失处理流程 (需包含 CPU 与磁盘的交互过程与 I/O 方式, 建议不超过 100 字)。

当发生页面缺失时, 处理器会触发一个中断信号。操作系统会接管控制权, 执行页面缺失处理流程。操作系统会通过磁盘 I/O 操作将页面从磁盘读取到内存中的空闲页面中。具体来说, CPU 通过总线发送读取请求给磁盘控制器, 磁盘控制器根据请求定位到正确的磁盘块, 并将数据传输给内存。一旦页面从磁盘读取到内存中, 操作系统会更新页表, 将页面重新映射到正确的物理地址。

8. 某同学买到了两款处理器芯片, 这两款芯片均只有一级处理器缓存。其中一款芯片 A 可以查询到芯片规范 (Specification), 另一个款芯片 B 查不到芯片规范。该同学想了解一下这两款芯片的缓存结构, 以改善程序性能。

内存地址采用从 0 开始字节编址的线性地址空间。

- (1) 该同学查询到芯片 A 的缓存大小 (数据部分) 为 32KB, 路数为 8, 缓存行大小为 64B。对于一个 32 位内存地址, 请标记其中的标记位 (Cache Tag, CT)、索引位 (Cache Index, CI)、块内偏移 (Cache Offset, CO) 部分, 并给出计算理由。

缓存行大小为 64B, 因此 CO 字段需要 6 个比特维护, 由于数据部分 32KB, 而缓存路数为 8, 因此 set 数为 $32KB / (8 \times 64B) = 64$, 因此 CI 字段需要 6 个比特维护, 其余部分为 CT

31-12	11-6	5-0
CT	CI	CO

- (2) 该同学查询到芯片 B 所在的系列芯片的相关参数的可能取值范围, 计划通过程序访问性能统计 (缓存命中率) 来推测芯片 B 的参数。

芯片 B 的可能取值范围如下:

- 缓存行大小可能的取值是: 16B、32B、64B 和 128B;
- 缓存大小 (数据部分) 可能的取值是: 4KB、8KB;
- 路数可能的取值是: 1、2、4、8。
- 缓存替换策略采用 FIFO 策略。

该同学设计的访问内存地址为:

15,	8193,	16384,	63,	4096,	8194,	64,	16385,
32768,	0,	129,	1024,	3072,	8192,	260,	513,
0,	4,	8,	4096,	64,	128,	128,	64

在对上述地址连续的 24 次访存中，采集到的前 8 次访问、中间 8 次访问、最后 8 次访问的缓存命中次数分别为 3 次、0 次、8 次。

请问该芯片的缓存行大小、路数多少、缓存大小，并简要写出理由。

缓存行大小为64B，路数为4，缓存大小为8KB，首先可以确认第一行缓存命中的是63，8194，16385，说明63和15在一个缓冲行内，64和15不在一个缓冲行内，因此缓冲行大小为64B，下一步，考虑到第二行对0的访问没有命中，说明是从第一行63以后的访问触发了缓存替换，将访问15时写入缓冲的缓冲行替换走，根据取值范围可以判断，CI的取值范围在 $2^3 \sim 2^3 2^7$ 之间，结合路数只能是2的整数次幂，可以推断出路数为4（刚好访问了相同的set5次导致的替换），最后时确定缓存大小，由于最后一行全部命中，说明访问第一次访问 4096 写入的缓冲没有被替换走，而如果缓存大小为4KB的话，CI需要4个比特位，那么从第一行访问 4096 开始访问了相同set的不同tag达到了四次（32768，0，129，1024，8192），这会导致 4096 对应的缓冲行被替换走，说明CI应该不止4个比特位，也就是说缓存大小为8KB

综上，缓存行大小为64B，路数为4，缓存大小为8KB

- (3) 该同学正在调试一个应用程序的性能，该程序包含三个双精度浮点数二维数组 M, N, Q 的操作，原始程序如下：

```

for (i=0; i<n; i++) {
    for (j=0; j<n; j++) {
        sum = 0.0;
        for (k=0; k<n; k++)
            sum += M[i][k] * N[k][j];
        Q[i][j] = sum;
    }
}

```

假设 n 足够大，请写出缓存效率最优的算法（不考虑矩阵切块），并分析其缓存命中率（缓存中仅保存 M、N、Q，循环变量和 sum 均在寄存器中）。

```

for(i=0;i<n;i++){
    for(j=0;j<n;j++){
        Q[i][j] = 0;
    }
}

```

```

}
for(i=0;i<n;i++){
    for(k=0;k<n;k++){
        for(j=0;j<n;j++){
            Q[i][j]+=M[i][k] * N[k][j];
        }
    }
}

```

优化策略是增加了对矩阵Q的初始化后，改变内层循环的顺序，使得最内层的循环可以在连续的内存上取值，从而最大程度使用缓存的数据

数组中每个数是双精度（8B），因此缓冲命中率可以粗略估计为 $1 - 8B/64B = 7/8$