

# Homework07

## 1(改编自COD\_CH,P343,5.16)

如5.7节所述，虚拟内存使用页表来跟踪虚拟地址到物理地址的映射。本题显示了在访问地址时必须如何更新页表。以下数据构成了在系统上看到的虚拟字节地址流。假设有4KiB页，一个4表项全相联的TLB，使用严格的LRU替换策略。如果必须从磁盘中取回页，请增加下一次能取的最大页码：

十进制	4669	2227	13916	34587	48870	12608	49225
十六进制	0x123d	0x08b3	0x365c	0x871b	0xbec6	0x3140	0xc049

TLB

有效位	标签	物理页号	上次访问时间间隔
1	0xb	12	4
1	0x7	4	1
1	0x3	6	3
0	0x4	9	7

页表

索引	有效位	物理页号/在磁盘中
0	1	5
1	0	在磁盘中
2	0	在磁盘中
3	1	6
4	1	9
5	1	11
6	0	在磁盘中
7	1	4
8	0	在磁盘中
9	0	在磁盘中
a	1	3
b	1	12

(1)对于上述每一次访问，列出

- 本次访问在TLB会命中还是失效
- 本次访问在页表中会命中还是失效

- 本次访问是否会造成缺页错误
- TLB的更新状态

可以参考表1，后续三、四问可以在此基础上加入tag和index等字段

表上有些空没有可以不填(例如TLB命中，page和page Fault可以空着)

- (2)重复(1)，但这次使用16KiB页而不是4KiB页。拥有更大页大小的优势是什么？有什么缺点？
- (3)重复(1)，但这次使用4KiB页和一个两路组相联TLB。(两路组相联TLB四个表项，前两个一组，索引为0，后两个一组，索引为1)
- (4)重复(1)，但这次使用4KiB页和一个直接映射TLB。(直接映射TLB四个表项从上到下的索引依次为0，1，2，3)
- (5)讨论为什么CPU必须使用TLB才能实现高性能。如果没有TLB，如何处理虚拟内存访问？

注：页表索引是不分tag，index的，本身就是虚页号，区别于TLB

(1)

1.2.3.4 代表级别，如最近，标为4  
4/2 代表最近使用且共用2次。

Address	Virtual Page	TLB H/M	Page H/M	Page Fault	TLB			
					Valid	LRU order	tag	Physical Page
0x123d	1	M	H	Y	1	1/0	b	12
					1	3/0	7	4
					1	2/0	3	6
					1	4/1	1	13
0x08b3	0	M	H	N	1	4/1	0	5
					1	2/0	7	4
					1	1/0	3	6
					1	3/1	1	13
0x365c	3	H	/	/	1	3/1	0	5
					1	1/0	7	4
					1	4/1	3	6
					1	2/1	1	13
0x871b	8	M	H	Y	1	2/1	0	5
					1	4/1	8	14
					1	3/1	3	6
					1	1/1	1	13
0xb6e6	b	M	H	N	1	1/1	0	5
					1	3/1	8	14
					1	2/1	3	6
					1	4/2	b	12
0x3140	3	H	/	/	1	1/1	0	5
					1	2/1	8	14
					1	4/2	3	6
					1	3/2	b	12
0xc049	c	M	M	Y	1	4/2	c	15
					1	1/1	8	14
					1	3/1	3	6
					1	2/1	b	12

(2)

表1

Address	Virtual Page	TLB H/M	Page H/M	Page Fault	TLB			
					Valid	LRU order	tag	Physical Page
0x123d	0	M	H	N	1	1/0	6	12
					1	3/0	7	4
					1	2/0	3	6
					1	4/1	0	5
0x08b3	0	H	/	/	1	1/0	6	12
					1	3/0	7	4
					1	2/0	3	6
					1	4/2	0	5
0x365c	0	H	/	/	1	1/0	6	12
					1	3/0	7	4
					1	2/0	3	6
					1	4/3	0	5
0x871b	2	M	H	Y	1	4/1	2	13
					1	2/0	7	4
					1	1/0	3	6
					1	3/3	0	5
0xb6e6	2	H	/	/	1	4/2	2	13
					1	2/0	7	4
					1	1/0	3	6
					1	3/3	0	5
0x3140	0	H	/	/	1	3/2	2	13
					1	2/0	7	4
					1	1/0	3	6
					1	4/4	0	5
0xc049	3	H	/	/	1	2/2	2	13
					1	1/0	7	4
					1	4/1	3	6
					1	3/4	0	5

好处: 提高TLB命中率, 加快存取速度.

坏处: 降低物理内存利用率, 更易产生碎片.

0	6	12	4
	7	4	1
1	3	6	3
	4	9	7

(3).

表1

Address	Virtual Page	TLB H/M	Page H/M	Page Fault	TLB			
					Valid	LRU order	tag	Physical Page
0x123d	1	M	H	Y	1	1/0	6	12
					1	3/0	7	4
					1	2/0	3	6
					1	4/1	0	13
0x08b3	0	M	H	N	1	4/1	0	5
					1	2/0	7	4
					1	1/0	3	6
					1	3/1	0	13
0x365c	3	M	H	N	1	3/1	0	5
					1	4/1	1	6
					1	1/0	3	6
					1	2/1	0	13
0x871b	8	M	H	Y	1	2/1	0	5
					1	3/1	1	6
					1	4/1	4	14
					1	1/1	0	13
0xb6e6	6	M	H	N	1	1/1	0	5
					1	2/1	1	6
					1	3/1	4	14
					1	4/2	5	12
0x3140	3	H	/	/	1	1/1	0	5
					1	4/2	1	6
					1	2/1	4	14
					1	3/2	5	12
0xc049	C	M	M	Y	1	4/2	6	15
					1	3/2	1	6
					1	1/1	4	14
					1	2/2	5	12

(4)

表1

Address	Virtual Page	TLB H/M	Page H/M	Page Fault	TLB			
					Valid	LRU order	tag	Physical Page
0x123d	1	M	H	Y	1	2	6	12
					1	4	0	13
					1	3	3	6
					0	1	4	9
0x08b3	0	M	H	N	1	4	0	5
					1	3	0	13
					1	2	3	6
					0	1	4	9
0x365c	3	M	H	N	1	3	0	5
					1	2	0	13
					1	1	3	6
					1	4	0	6
0x871b	8	M	H	Y	1	4	2	14
					1	2	0	13
					1	1	3	6
					1	3	0	6
0xb6e6	6	M	H	N	1	3	2	14
					1	2	0	13
					1	1	3	6
					1	4	2	12
0x3140	3	M	H	N	1	3	2	14
					1	2	0	13
					1	1	3	6
					1	4	0	6
0xc049	C	M	M	Y	1	4	3	15
					1	2	0	13
					1	1	3	6
					1	3	0	6

(5) <sup>没有</sup>如果TLB两话几乎每次虚拟内存访问都需要经过一次页表再经过一次磁盘访问,累积起来两话性能将下降.

## 2(改编自COD\_CH,P345,5.24)

本题研究具有写缓冲区的处理器的cache控制器的控制单元。使用图5-39的有限状态自动机作为设计有限状态自动机的起点。假设cache控制器用于5.9.3节描述的简单直接映射cache(图5-39)，但你需要再添加一个容量为1个块的写缓冲区。

回忆一下，写缓冲区的目的是作为临时存储器，这样处理器就不必等待脏块失效的两次内存访问。它不是在取新块之前写回脏块，而是缓冲脏块并立即开始读取新块。然后，在处理器工作时再将脏块写入主存。

- (1) 如果处理器在从写缓冲区将块写回主存时发出一个命中cache的请求，会发生什么？
- (2) 如果处理器在从写缓冲区将块写回主存时发出一个cache失效的请求，会发生什么？
- (3) 设计一个有限状态自动机以启用写缓冲区

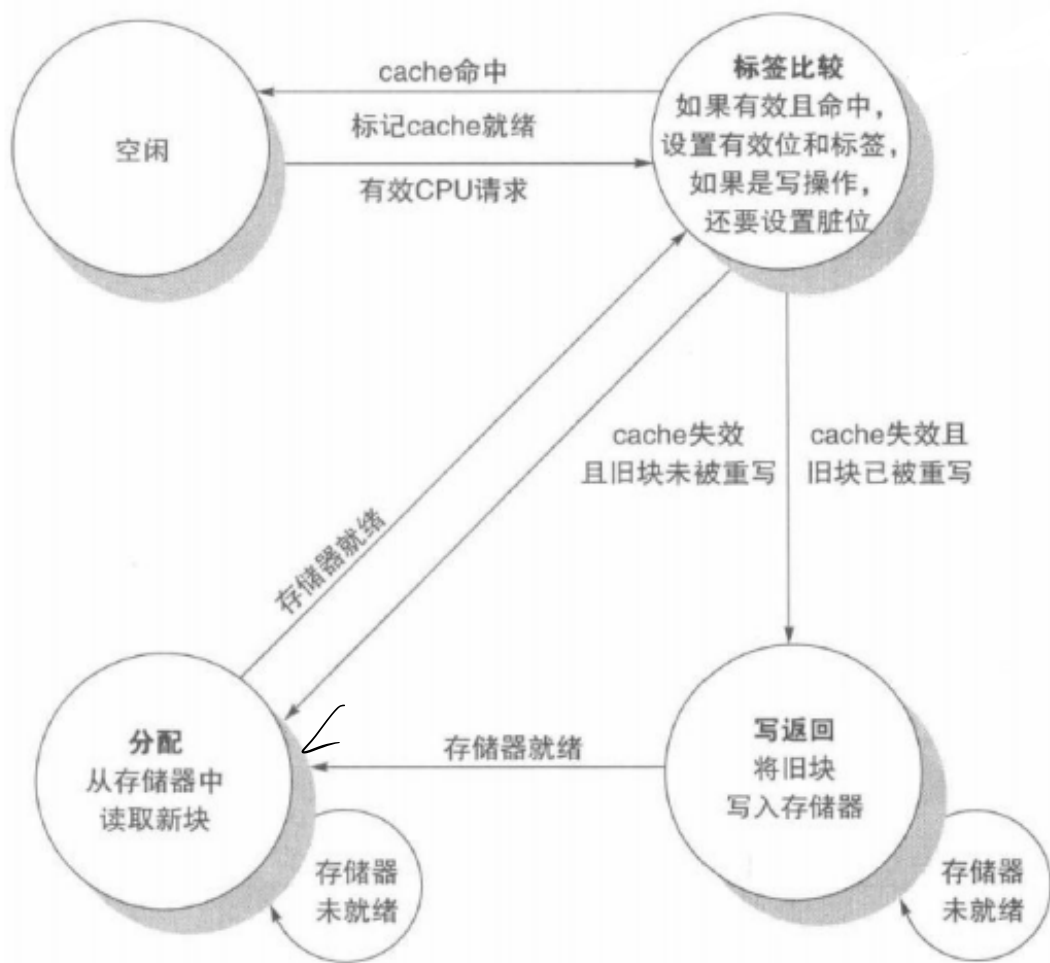
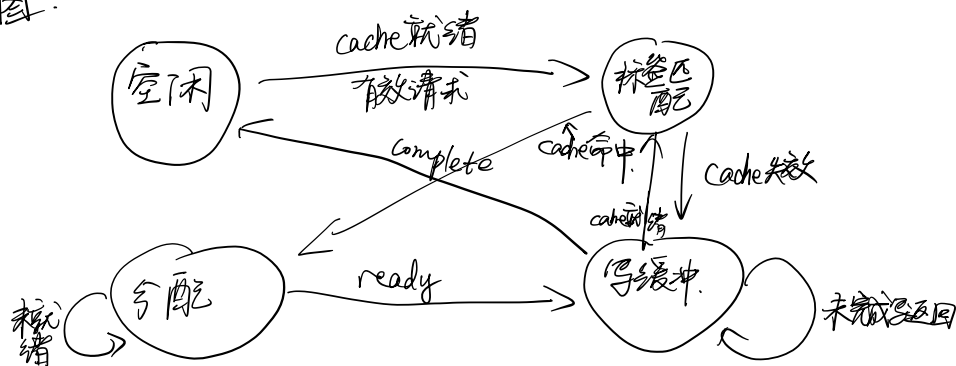


图5.39

- (1) cache仍能满足这个请求，即使正在发生写回；
- (2) cache需要等待直到写回完成。
- (3) 如图。



### 3(改编自唐书, P151, 4.17、4.18)

(1) 写出1100、1101、1110、1111对应的海明码(按偶配原则, 高位在右边, 低位在左边)

(2) 已知接受到的海明码(按偶配原则配置, 高位在右边, 低位在左边)为1100100、1100111、1100000、1100001, 检查上述代码是否出错? 第几位出错?

注: 如果觉得高位写在右边不方便, 可以说明一下写在左边也是可以的, 但是要注意所有位置的顺序

(1) 设校验码为  $A_1, A_2, A_4$ . (2) 设  $S_1, S_2, S_3$  用作校验.  $\overline{S_3 S_2 S_1}$  代表出错位.

$$\begin{cases} A_1 = A_3 \oplus A_5 \oplus A_7 \\ A_2 = A_3 \oplus A_6 \oplus A_7 \\ A_4 = A_5 \oplus A_6 \oplus A_7 \end{cases}$$

$$\begin{cases} S_1 = A_1 \oplus A_3 \oplus A_5 \oplus A_7 \\ S_2 = A_2 \oplus A_3 \oplus A_6 \oplus A_7 \\ S_3 = A_4 \oplus A_5 \oplus A_6 \oplus A_7 \end{cases}$$

$$1100 \rightarrow 0111100$$

$$1101 \rightarrow 1010101$$

$$1110 \rightarrow 0010110$$

$$1111 \rightarrow 1111111$$

4

$$1100111$$

$$\begin{cases} S_1 = 1 \\ S_2 = 1 \\ S_3 = 1 \end{cases}$$

出错,  $A_7$  取反

$$1100000$$

$$\begin{cases} S_1 = 1 \\ S_2 = 1 \\ S_3 = 0 \end{cases}$$

出错,  $A_6$  取反

$$1100001$$

$$\begin{cases} S_1 = 0 \\ S_2 = 0 \\ S_3 = 1 \end{cases}$$

出错,  $A_4$  取反.

某磁盘转速为10000转/分, 平均寻道时间是6ms, 磁盘传输速率是20MB/s, 磁盘控制器延迟为0.2ms, 读取一个4KB的扇区所需平均时间约为多少?

$$\text{平均旋转延迟} = 10000 \text{ 转/分} \Rightarrow 6 \text{ ms/转} \Rightarrow 3 \text{ ms/转}$$

$$\text{传输时间: } 4 \text{ KB} / 20 \text{ MB/s} = 0.2 \text{ ms}$$

$$\text{平均时间: } 6 + 3 + 0.2 + 0.2 = 9.4 \text{ ms}$$