

1

| | V | Tag | Data | V | TAG | Data |
|-------|---|----------------------|------|---|----------------------|------|
| SET 0 | 1 | 101101 M[180-181] | | 1 | 001011 M[44-45] | |
| SET 1 | 1 | 000011 M[14-15] | | 1 | 101111 M[190-191] | |

组索引位 $s=1$, 偏移位 $b=1$

3->00000011: set 1; tag11=000000 miss 替换第一行空 (先从内存中取出块存入)

180->10110100: set 0; tag01=101101 miss 替换第一行空

43->00101011: set 1; tag11 不匹配 miss 替换第二行空 tag12=001010

2->00000010: set 1; tag11 匹配 取出

191->10111111: set 1; 无匹配 替换第二行 (最后一次访问较久远) tag12=101111

88->01011000 set 0; 无 替换第二行空 tag02=010110

190->10111110 set 1; tag12 匹配 取出

14->00001110 set 1; 无 替换第一行 (最后一次访问较久远) tag11=000011

181->10110101 set 0; tag1 匹配 取出

44->00101100 set 0; 无 替换第二行(最后一次访问较久远) tag02=001011

结果如表

共命中 3 次, 分别为地址 2、190、181

2、

C1: $E=1; b=1; s=2;$

C2: $E=1; b=2; s=1;$

| Address | Binary address | C1 hit/miss | C2 hit/Miss |
|---------|----------------|-------------|-------------|
| 3 | 000000 11 | M | M |
| 180 | 101101 00 | M | M |
| 43 | 001010 11 | M | M |
| 2 | 000000 10 | M | M |
| 191 | 101111 11 | M | M |
| 88 | 010110 00 | M | M |
| 190 | 101111 10 | H | H |
| 14 | 000011 10 | M | M |
| 181 | 101101 01 | H | M |
| 44 | 001011 00 | M | M |

如表, C1 不命中的少, 周期少, C1 更好

C1: miss rate = $8/10 = 80\%$, total cycles = $8*25 + 10*3 = 230$ (访问了 10 次)

C2: miss rate = $9/10 = 90\%$, total cycles = $9*25 + 10*5 = 275$ (访问了 10 次)

3、

$C=2^{15}$ bytes

$B=2^3$ bytes $b=3$

$E=1$

$S=2^{12}$ $s=12$

$t=32-s-b=17$

总: $32*1024+(1+17)*8=41984$ bytes

| |
|----|
| 得分 |
| |

第五题 (16 分)

现有一个直接映射 (Direct-mapped) 的高速缓存 (cache)，其总容量为 16 字节，每个高速缓存块 (cache block) 为 4 字节 ($B=4\text{Byte/block}$)。Cache 的初始状态为空。

(1) 经过如下地址访问序列后，请填写 cache 第 2 组 (即 Set 2，注：从 0 开始编号) 的状态。数据块用 10 进制表示，举例：M[4-7] 表示地址 4-7 的数据。地址序列全部为读访存，每次读取 2 个字节，地址单位为字节。

(提示：一次访存可能访问多个高速缓存块。)

R7 时, R8 会在 set 3 填上 0010 Block

| | | tag | Set | offset | |
|---|---|-----------------------|-----|--------|-----|
| R | 0 | [000000] ₂ | | | M |
| R | 1 | [000001] ₂ | | | H |
| R | 7 | [000111] ₂ | | | M M |
| R | 4 | [000100] ₂ | | | H |
| R | 0 | [000000] ₂ | | | H |

| v | Tag | Block |
|---|-----|---------|
| 1 | 00 | M[8-11] |

Set 2

本次访存序列，共发生 3 次高速缓存缺失 (miss)？

(提示：一次访存可能引发多次高速缓存缺失。)

(2) 接上题，高速缓存重新清空后，执行如下访问序列，同时包含读访存 (R) 和写访存 (W) 操作，每次读/写 2 个字节，高速缓存采用 write back + no-write-allocate 的管理策略，其他配置不变。请填写访存后 cache 第 1 组 (Set 1) 的状态。

| | | tag | Set | offset | |
|---|----|-----------------------|-----|--------|--------------------------------|
| R | 0 | [000000] ₂ | | | Read miss |
| R | 1 | [000001] ₂ | | | Read hit |
| W | 7 | [000111] ₂ | | | Write miss (no-write-allocate) |
| R | 12 | [001100] ₂ | | | Write miss (no-write-allocate) |
| R | 9 | [001001] ₂ | | | Read miss |

| v | Tag | Block |
|---|-----|-------|
| 0 | — | — |

Set 1

本次访存序列，共发生 1 次高速缓存命中 (hit)？

(3) 为了提升高速缓存的效率，设计师提出两种修改方案

- 将高速缓存块增大到 8 字节 ($B=4\text{Byte/block}$)，其他配置不变，每次访存命中 (hit) 的时间开销为 4 个时钟周期；
- 将高速缓存变为全相连 (fully-associative, 即 $S=1$)，其他配置不变，每次访存命中 (hit) 的时间开销为 2 个时钟周期；

对于如下访问序列，分别采用两种修改方案的访存时间总开销相等，请问每次高速缓存缺失 (miss) 的时间开销是 10 时钟周期？

记 miss 开销 m 个时钟周期

| | | tag | a) | b) |
|---|---|-------------------------|-----|-----|
| R | 0 | [000000] ₂ , | M | M |
| R | 1 | [000001] ₂ , | H | H |
| R | 7 | [000111] ₂ , | H M | M M |
| R | 8 | [001000] ₂ , | H | H |
| R | 9 | [001001] ₂ | H | H |

$$4 \times 4 + 2m = 3 \times 2 + 3m$$

$$m = 10$$

提示：修改高速缓存配置不会影响高速缓存缺失的时间开销