

访问地址为0x724

- 4.17 (1) 页大小 64 字节 \Rightarrow 页内偏移 6 位, 则 0x05a4 的标签为 0x05, 索引 0x02, 命中
 (2) 页内偏移 6 位, 虚拟地址 14 位, 则虚拟页号 8 位, 页素有 $2^8 = 256$ 个条目
 (3) 使用物理地址 0x724, 块内偏移 2 位, 索引 4 位, 则标签为 0x1C, 索引为 0x9
 块内偏移为 0x0, 由于组号为 0x9 的标签为 0x1C, 则缓存命中, 结果为 0xb3

4.18 (1)	访问地址	A	B	C	D	A	B	C	D	(1) 程序长时间运行时命中率为 0
	way 0	-	A	A	C	C	A	A	C	(2) 每次均保留未来更快被访问的地址, 替换另一个, 程序长时间运行
	way 1	-	-	B	B	D	D	B	B	
	命中?	N	N	N	N	N	N	N	N	
2)	访问地址	A	B	C	D	A	B	C	D	时平均缺失 2 次后命中 1 次, 即命中
	way 0	-	A	A	A	A	A	B	C	率为 33%
	way 1	-	-	B	C	D	D	D	D	

- 4.19 (1) 若同一缓存组内存在多个相同的低位标签 (高位标签一定不相同, 因为同一缓存组具有相同的组索引), 则在初次判断命中时可能冲突, 即发生多个命中结果; 为了确立应取其中哪个结果, 需要引入新的算法和逻辑, 加大开销降低效率
 (2) 对于通常的缓存替换策略如 LFU, LRU 等, 均是通过给缓存块贴时间或频次标签等进行的, 当低位标签不唯一时, 替换策略算法便无法确定多个命中结果中应该更新哪一个的标签。这可能会导致比较时间变长, 或缓存行状态维护发生偏差
 (3) 页大小 16 KB \Rightarrow 页内偏移 14 位,

8KB 四路组相联, 每路大小为 2KB \Rightarrow 索引和块内偏移共 11 位
 \Rightarrow 至多可以拥有 3 比特位低位标签

- 4.20 监听-致性。优点: 实现简单, 每个缓存内部维护状态, 不需要额外的目录结构实现
 ② 响应速度快, 当一个缓存中的共享数据被修改时可迅速通过总线进行广播, 而不需跨越缓存层级到达目录

缺点：①性能开销较大，需要各个核心的缓存频繁进行监听，影响处理器自身性能 ②可能出现短时间内两个核心共同修改共享数据情形，监听无法对此进行及时处理

目录-致性：优点：①只需在存在共享数据需求时进行处理，通信开销小 ②目录对并发性问题有更灵活自主的处理方式，策略等，可以支持较高的并发度

缺点：①额外需要编写目录的逻辑，存放目录结构，对缓存容量进行挤占 ②处理并发性问题时需要跨层级缓存，即时性较低

从中可以看出缓存-致性需要不时处理监听，目录等，会影响处理器自身的执行，也会要求更多的缓存空间，硬件逻辑等