

Week 13



假设一个使用虚拟内存和 L1 缓存的存储系统具有以下特征：

- 内存系统按字节寻址，访存请求每次仅传递一个字节给处理器。
- 虚拟地址长度 14 比特，物理地址长度 12 比特。
- 页大小 64 字节，使用单级页表。
- TLB 拥有 16 个条目，四路组相联。
- L1 缓存物理寻址，块大小 4 字节，共 16 个组，直接映射。

现在 CPU 发起了一次对虚拟地址 0x05a4 的单字节内存加载请求，回答以下问题。

- 若请求发起时，TLB 的部分内容如下表所示。则 TLB 是否发生命中？如果命中，此次内存访问的物理地址是多少？

组号	标签	物理页号	有效位	标签	物理页号	有效位
0	0x0B	—	0	0x1F	—	0
	0x07	0x0D	1	0x02	0x2F	1
1	0x01	0x05	1	0x05	0x0D	1
	0x14	—	0	0x2A	0x16	1
2	0x03	—	0	0x05	0x1C	1
	0x0B	0x07	1	0x00	0x1B	1
3	0x26	0x34	1	0x02	—	0
	0x19	0x2F	1	0x38	—	0

- 该系统的页表有多少个条目？

- 如果 TLB 命中，则使用 1) 得到的物理地址，否则使用物理地址 0x1e4。如果 L1 缓存的内容如下表所示，则此次访存请求是否命中缓存？如果命中，访存结果是多少？

组号	标签	有效位	块偏移			
			0x0	0x1	0x2	0x3
0	0x1F	0	—	—	—	—
1	0x05	1	0x02	0x09	0xCB	0xA3
2	0x1C	1	0x09	0x55	0x01	0x08
3	0x0D	0	—	—	—	—
4	0x1B	1	0x9B	0xEE	0xE2	0x86
5	0x2F	1	0x00	0x00	0x01	0x00
6	0x07	0	—	—	—	—
7	0x05	1	0x6F	0x23	0xAB	0xD0
8	0x16	0	—	—	—	—
9	0x1C	1	0x63	0x2F	0x1B	0x00
10	0x1C	1	0x28	0x34	0x01	0xC4
11	0x16	1	0x29	0xC8	0x56	0x99
12	0x34	0	—	—	—	—
13	0x34	0	—	—	—	—
14	0x0D	0	—	—	—	—
15	0x07	1	0xE8	0x59	0x04	0x45

(1)  $0x05a4 = 0b00010110100100$

Tag = 0x05. 组号为 2  $\Rightarrow$  命中

物理页号 0x1C = 0b011100

物理地址 0b011100100100 = 0x724

(2) 虚拟页一页对应一个条目  $\Rightarrow$  有  $2^{14} \div 64 = 2^8 = 256$  个条目

(3)  $0x724 = 0b011100100100$

Tag 0b011100 = 0x1C 组号 0b1001 = 0d9

块偏移 00

$\Rightarrow$  查得：命中。内容为 0x63.

18

一段程序循环往复地按顺序访问 A、B、C、D 四个地址上的数据。考虑一个拥有 2 条目的全相联缓存，回答以下问题。

- 使用 LRU 替换策略时，填写下表。当程序长时间运行时，缓存的命中率为多少？

访问地址	A	B	C	D	A	B	C	D
way 0	—	A	A	C	C	A	A	C
way 1	—	—	B	B	D	D	B	B
命中?	N	N	N	N	N	N	N	N

- 提出一种缓存替换策略，使得上述程序可以在该缓存中拥有最大的命中率，并计算该命中率。

(1) 命中率 0%.

(2) 替换策略：只替换空缓存，运行情况如下：

访问地址	A	B	C	D	A	B	C	D
way 0	—	A	A	A	A	A	A	A
way 1	—	—	B	B	B	B	B	B
命中?	N	N	N	N	Y	Y	N	Y

命中率为  $\lim_{N \rightarrow \infty} \frac{2(N-1)}{4N} = 50\%$ .

19. (1) LTag 唯一, 保证同组内通过 LTag 判断是否命中的准确性, 节省了再访问 HTag 区的时间; 若不唯一, 相同 LTag 的地址之间可能会出现误命中等错误。

(2) ① 替换的地址 LTag 要在同组内唯一

② 考虑低位命中为预测性的, 替换时要考虑 HTag 对比的性能开销

(3)  $16KB = 2^{14} \text{ Byte}$        $8KB$  回路组  $\Rightarrow 2^{14} \div 4 = 2^{12}$  组

$\Rightarrow 14 - 11 = 3$  位

块大小 1 Byte?

20. ① 监听一致性的优点是实现简单, 不需要额外目录结构, 只要每个 core 都监听总线上的 data 来判断是否修改; 缺点是扩展性差, cores 越多 bus 负担越大, 徒增功耗和带宽消耗。

② 目录一致性优点是扩展性好, 不要大的广播网络; 但缺点是实现复杂, 需额外目录和控制单元, 而这些部分出错也会损失性能。

③ 缓存一致性实现代价有: 占用通信网络带宽, 占用存储空间存储一致性数据, 存在通信延迟, 消耗功耗等