

17. 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	组号	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	0x1F	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 i) 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

- 虚拟地址为 0x05a4, 且有 14 位, 即 00010110100100。

提取出以下几个字段: TLBI: 0x05, 占高 6 位, 用于对 TLB 中项标签相匹配。

TLBI: 0x02, 占 7~8 位, 用于 TLB 中组索引。

VPN: 0x1B, 占高 3 位, 页号。

VPO: 0x24, 占低 6 位, 页内偏移。

故 TLB 发生命中, 转换后的物理页号是 0x1C, 即 011100, 则本次内存访问的物理地址为 011100100100。

- 物理地址有 12 位, 即内存大小为 4KB, 而页的大小为 64B, 故需要 2<sup>6</sup> 个页, 采用单级页表, 故页表中有 64 个页表项, 即 64 个页表条目。

- 访问内存的物理地址为 011100100100, 其中高 6 位用于标签匹配, 中间 4 位用于组索引, 低 2 位用于选择组中的具体块。

011100 即 0x1C      1001 即第 9 组      00

查表可以发现 L1 缓存命中, 读出块偏移为 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。

1) 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

2) 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。

2) 只有缓存连续2次命中才发生快换替换。当发生快换时替换的规则与LRU相同。命中率可以达到50%。

19. 一些处理器引入了“微标签”(microtag)的技术来降低组相联缓存标签匹配过程的时序压力。该技术将地址的标签部分进一步拆分为高位标签 (HTag) 和低位标签 (LTag)，在判断缓存命中与否时，控制器仅取出低位标签进行比较，将匹配的缓存块预测为一次命中并把数据前馈给处理器。在随后的剩余周期内，高位标签被取出并进一步用于判断该预测最终是否构成真正的命中。回答以下问题：

- 低位标签在同一缓存组内通常被要求是唯一的，试说明原因。
- 基于对 1) 的讨论，简要说明该技术的引入对于通常的缓存数据重定向有什么影响。
- 考虑到虚拟页偏移和物理页偏移是一致的，为了提高访存性能，系统可以进一步要求地址的低位标签和组索引完全位于页偏移字段内，这样低位标签的匹配过程就完全不需要经过地址翻译而可以直接进行，后续的高位标签则使用页表翻译后的结果判断是否构成真实命中。基于上述过程，对于 16KB 页大小的内存系统，一个 8KB 大小的四路组相联缓存至多可以拥有几比特的低位标签？

① 因为引入了 microtag 技术，在缓存控制器会将低位标签命中数据块直接前馈给处理器，减少了高位标签匹配的时间以提升缓存性能，但实现的前提是组内低位标签是唯一的，否则若组内多个块低位标签都匹配，必须进行高位标签的比较，避免导致额外的时序开销。

② 当缓存采取直接映射的策略时，若某组块仍在进行高位标签的匹配，而后续操作又需要对该组进行快替换，那就会导致冲突。

③ 页内偏移占 12 位，其中组索引和块内偏移占 9 位，故最多 3 比特的低位标签。

20. 监听一致性和目录一致性各有什么优缺点？简述缓存一致性的实现代价体现在哪些方面？

总线监听比基于目录的一致性机制更快，但对总线带宽的要求高，且可扩展性有限，频繁地监听缓存会导致与处理器访问竞争，从而增加缓存访问的时间和功耗。

目录一致性机制虽然速度较慢，但可扩展性更好，对带宽要求也没有监听一致性协议高。

实现代价：需要额外维护缓存有行状态信息，状态转换的检测和处理，缓存一致性信息的传递等，会消耗一定的处理器时间和带宽资源，且由于多级缓存和多处理器系统的复杂性，会导致系统调试和验证变得更加困难。