

# 第十四周作业. 17-20 上次16题忘做了, 这里补上

(1) 512 Byte cache. 28路 line size: 16 byte index: 4 bit LRU 策略.

input: 0 ~ 0010|0000|0000 offset: 4 bit  
0010|0000|0000 ~ 0100|0000|0000 input[0][~]  
input[1][~]

① 从 0 ~ 0001|0000|0000

0010|0000|0000 ~ 0011|0000|0000 命中 96/128. 缓存块全部填充  
i 从 0 变为 63.

② 从 0001|0000|0000 ~ 0010|0000|0000

0011|0000|0000 ~ 0100|0000|0000 命中 96/128. i 从 64 ~ 127

总命中率 75%

(2) 增加总缓存大小. 即增加 line 数. index 位增加. 假设增加到 6 bit.

则: 0 ~ 1|000000|0000

对于 0 和 1|000000|0000 开头的 input 地址.

命中率依然是 75%. 保存在不同组中

假设 index 5 位.

开始

0 ~ 10|00000|0000 < 01|00000|0000 开始

i = 0 ~ 128 全部 cold miss. 75%.

index 更高位时. 命中率也不会增加, 这是程序的顺序取数的模式决定的

(3) 增加 line size. 可以改善命中率. line size = 16 byte  $\rightarrow$  32 byte (8 个数据)

0 和 1|0000|0000 开头.

i 从 0 变到 128. 命中  $\frac{7}{8}$

缺失率 12.5%



17. 虚拟地址: 14bit 物理: 12bit



Page size: 64 Byte. 单级页表.

TLB: 16 entries, 4路组相联 ...

L1: cache line: 4 Byte 16 sets. 直接映射:

(1)  $0x05a4 = 00010110100100$

TLB:  $10110 \rightarrow 10110$ , tag=101  $\rightarrow 0x05$   
四路组相联

物理页号  $0x10 \rightarrow 10000$ , 页内偏移  $100100$ .

物理地址:  $0100001100100$ . 命中

(2) 物理地址空间:  $2^{12} = 4KB \rightarrow 64$  个页

页表条数: 64 条

(3) 使用物理地址  $0x424$

$0100001100100$  标签  $10000 \rightarrow 0x10$  未命中缓存

18. 2条目的全相联缓存.

A B C D

(1). 访问地址

A B C D A B C D

0 - A A C C A A C

1 - - B B D D B B

hit N N N N N N N N

缓存命中率为0.

(2) 每个周期内 A 只出现一次. 假设对 A 要命中. 那么该周期 A 的缓存一定是来自上一周期; 而访问 A, B, C, D 过程中缓存一定要不断更新的. 提出下面策略.



① 有空位时填补空位

② 当前缓存块中, 前一级若有命中, 替换掉命中的

③ 前一次未命中, 替换掉前次替换的, 下面画波浪线的表示要替换的

A	B	C	D	A	B	C	D	A	B	C	D	A	B	C	D
-	A	A	A	<u>A</u>	<u>A</u>	<u>B</u>	C	<u>C</u>	<u>C</u>	<u>C</u>	<u>C</u>	<u>D</u>	A	A	A
-	-	<u>B</u>	<u>E</u>	D	D	D	<u>D</u>	<u>D</u>	A	B	B	B	<u>B</u>	<u>B</u>	C
N	N	N	N	Y	N	N	Y	N	N	Y	N	N	Y	N	Y

命中率  $\frac{1}{3}$

19. (1). 地址 Tag  $\rightarrow$  HTag LTag

一个组内 Tag index offset, index 是相同的.

假设为 4 路组相联, 则对于特定的 index, 比较 4 路量时与 LTag 对应位是否相同. 在同一组内, 一般缓存命中只有 1 块, 否则会出现逻辑错误. 因此 LTag 在同一缓存组内应当是唯一的.

(2) 缓存替换策略必须考虑到换入的 cache line 的 LTag 不能与其他缓存块相同. 缓存替换难度更高, 可能有相同 LTag 的缓存不能同组.

(3) 16KB 页大小  $\rightarrow$  offset 有 14 bits HTag

8KB cache + 4 路  $\rightarrow$

总容量:  $2^m \times 4 \times 2^n = 8K = 2^{13}$

位数:  $x + m + n = 14$ .

故:  $x = 3$

若高于 3 bits 位, 则地址的 LTag 和 index 位不足以仅放在页偏移字段内



## 20. 监听一致性:

- ① 优: 基于总线, 容易实现和部署; 处理响应快.
- ② 缺: 处理器数量大时, 总线数据流量大, 导致性能瓶颈.  
扩展性差.

## 目录一致性:

- ① 优: 性能高(总线流量小), 可扩展性好, 易配置优化.
- ② 缺: 实现复杂, 存储开销大, 延迟高.

## 实现缓存一致性的代价:

- ① 缓存一致性为避免逻辑错误, 需要额外的<sup>等待</sup>时间和存储开销.
- ② 系统复杂性、管理维护难度更高.

