

10/27: Chapter 9: Caches (Continued)

Review:

Fast

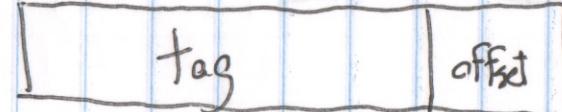
Big

Pick one!

(C, B, S)

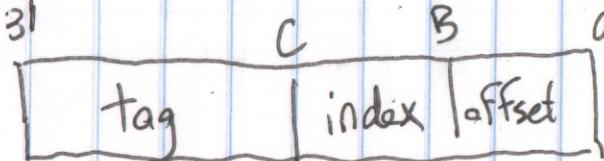
FA: 31

MAR



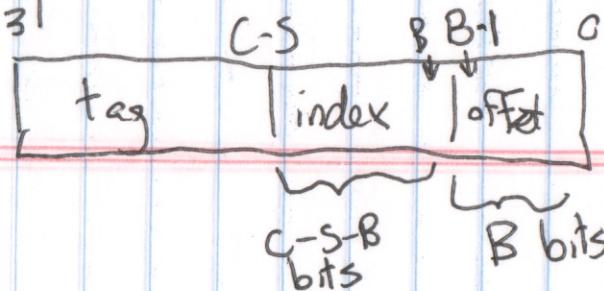
(S=C-B)

DM: 31



(S=0)

SA: 31

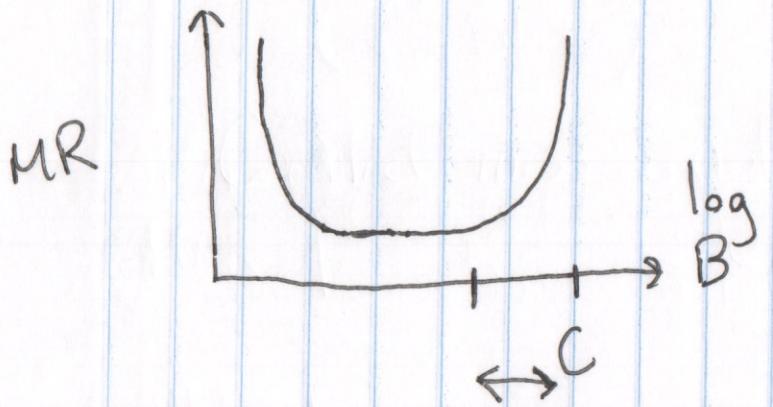
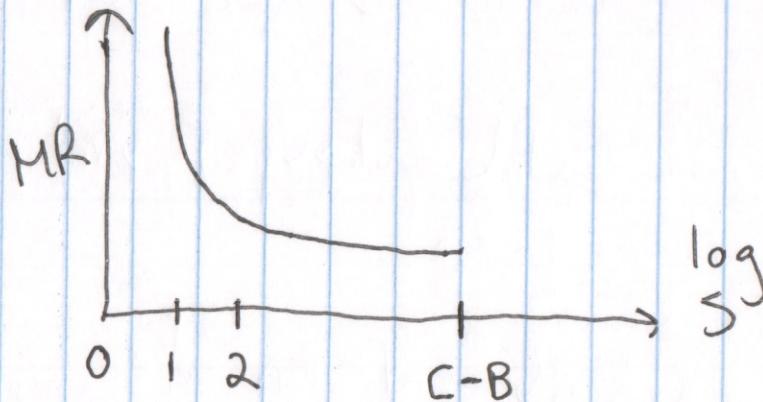
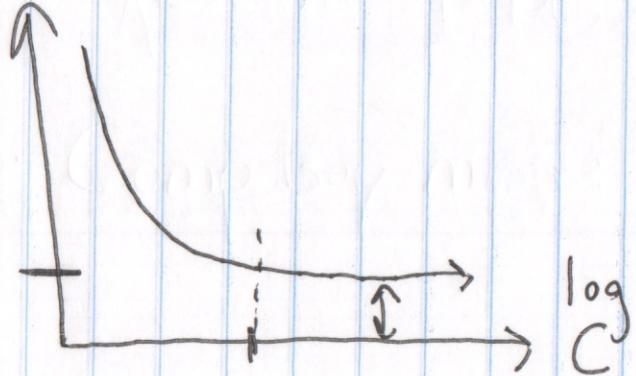


2^C bytes of data storage

2^B bytes per block

2^S ways (blocks set)

MR
compulsory miss rate



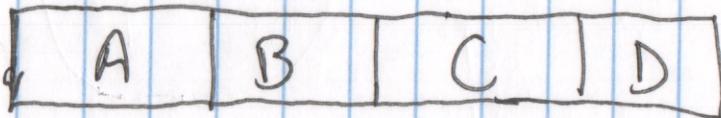
$\leftrightarrow C$
Cache pollution: bringing in unnecessary data

Cache Optimizations

• Block Replacement

4-way SA \neq

Trace: A, B, C, D, D, D, B, C, A, E
↑ ↑ ↑ ↑ h h h h m

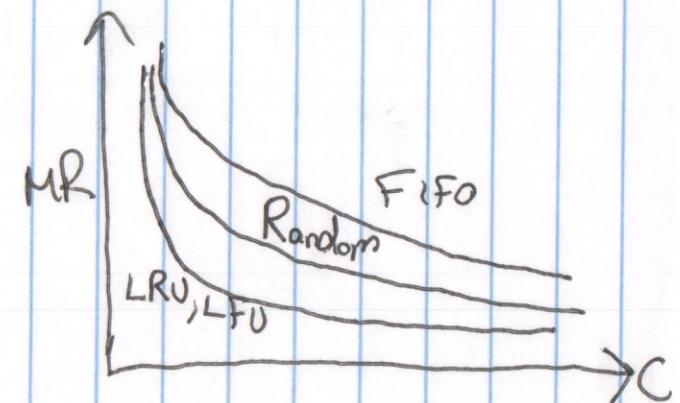


{ • Least Recently Used (LRU) : D

• First in First Out (FIFO) : A

• Least Frequently Used (LFU) : A, B, or C

• Random: any!



Write policies

SW X

- 1) When to write (to next level)?
- 2) Allocate on miss?

→ Write back (WB): only write to next level when block is evicted

- requires dirty bit

eviction:

if block.dirty : write back

else: no need

Write through (WT): ~~update~~ update lower levels of mem on

any write

+ consistency at all times at all levels

- memory traffic ↑

2) Allocate on miss?

- Write allocate (WA): bring into \$ on miss

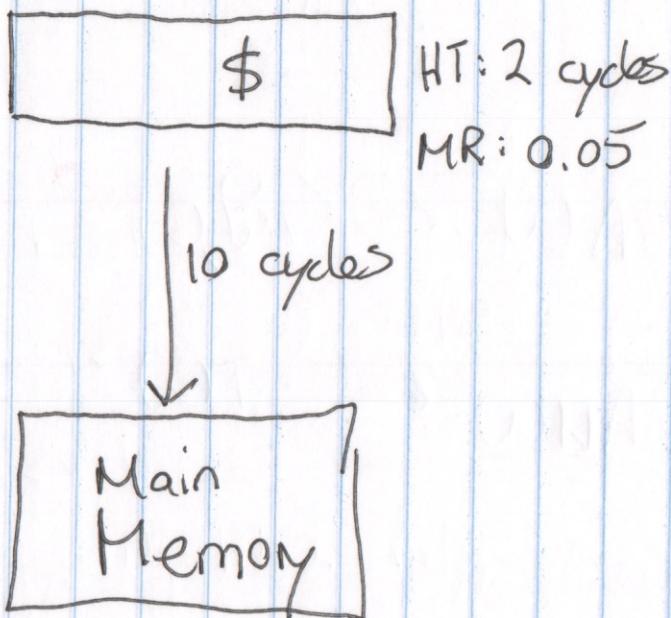
- Write no allocate (WNA): don't

WB → WA (project)

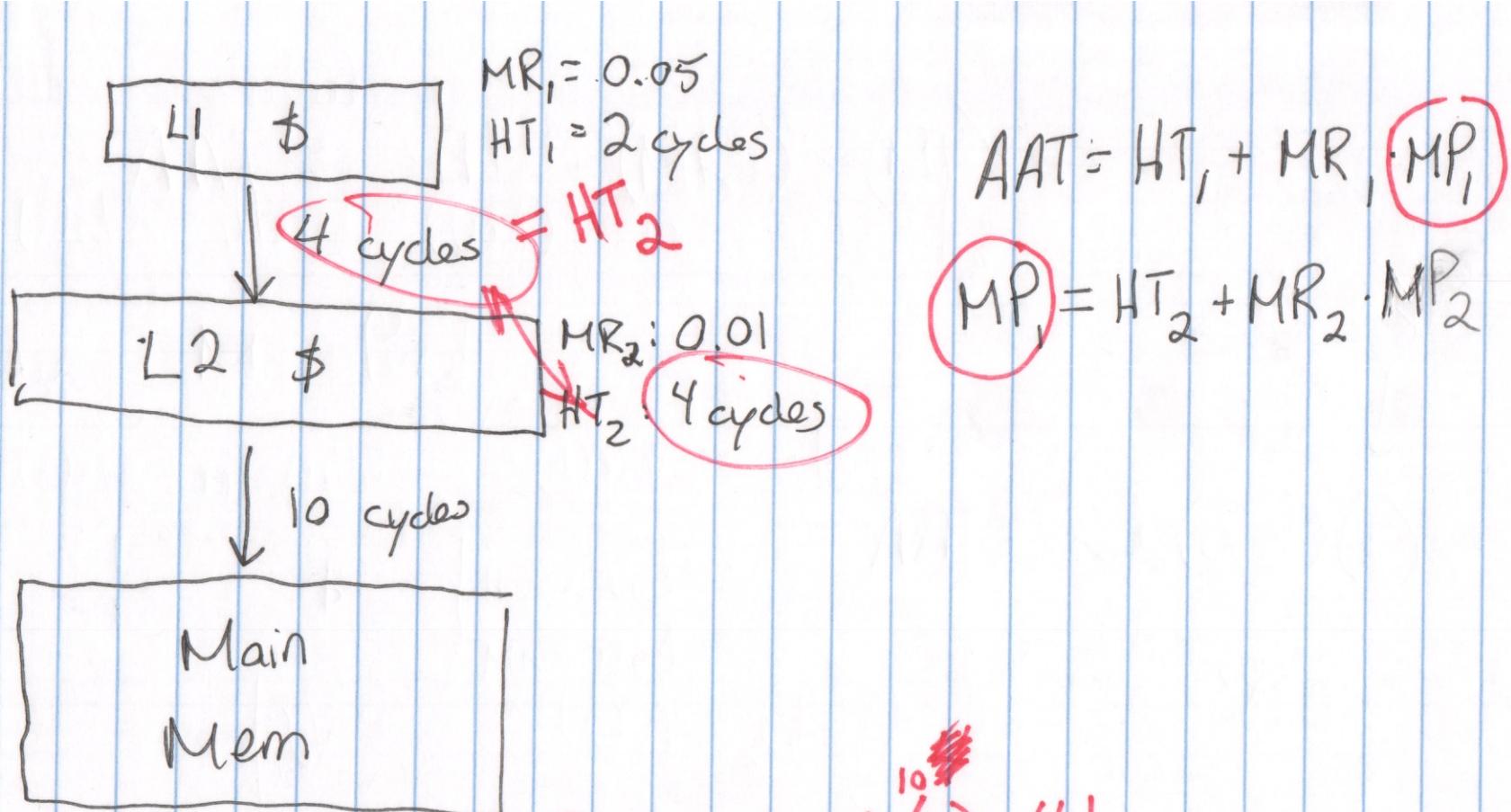
WT → WNA

$$AAT = HT + (MR \cdot) \times (MP)$$

Reducing Miss Penalty:



$$\begin{aligned} AAT &= 2 + (0.05)(10) \\ &= 2.5 \text{ cycles} \end{aligned}$$



$$AAT = HT_1 + MR_1 \cdot MP_1$$

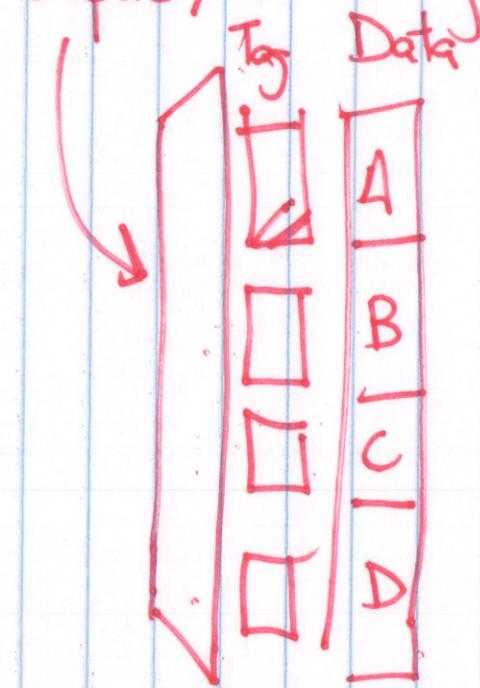
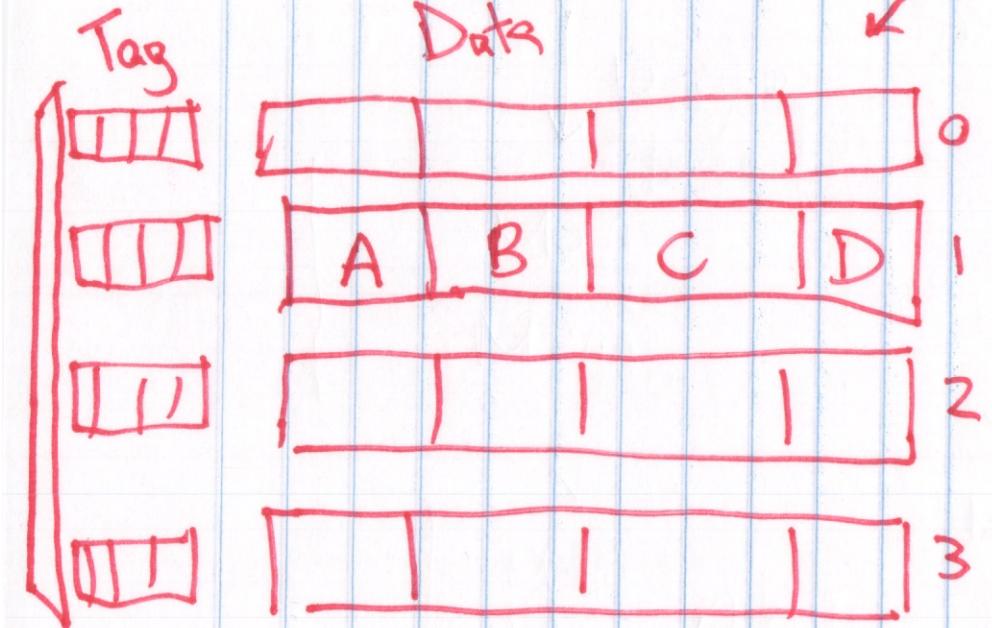
$$MP_1 = HT_2 + MR_2 \cdot MP_2$$

$$MP_1 = 4 + (0.01) \cancel{(10)} = 4.1$$

$$AAT = 2 + 0.05 \times 4.1 = 2.21 \text{ cycles}$$

The 3 "Cs"

E, A, B, C, D, D, D, B, C, A, E
 Compulsory misses hits
 4-way SA ↑
 Conflict (if it maps to index 1)
 Capacity (if not big enough)



FA

if (miss in SA) AND (would have missed in FA): Capacity miss

if (miss in SA) AND (would have hit in FA): Conflict miss