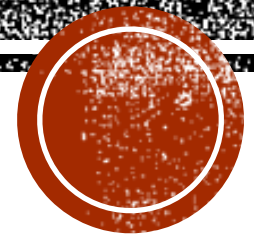
CACHES



Tony present

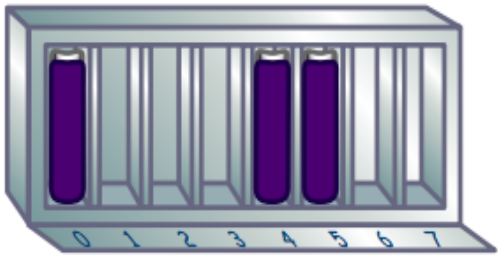
THINGS WE NEED TO CONSIDER!

1. Address assignment. Tag, index, byte offset.
2. Size of total cache.
3. To examine the algorithm.
4. Cache friendly programing.
5. Average memory access time. (AMAT) for multi caches

- Mostly consider 3 times => Types of cache = 3

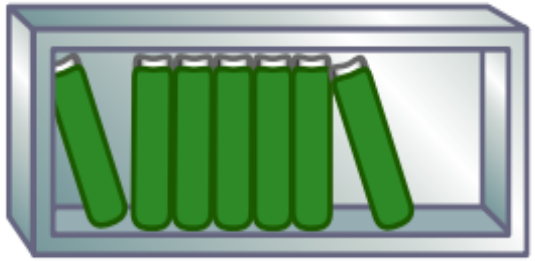
TYPES OF CACHES, THIS TOPIC SUCKS

- Direct Mapping cache



- N-Way Set Associative

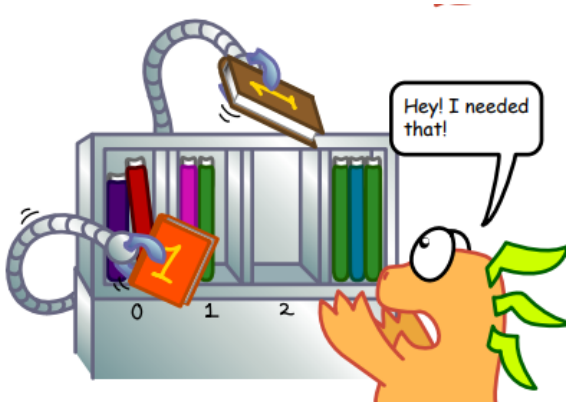
- Fully Associative



3CS MISSES

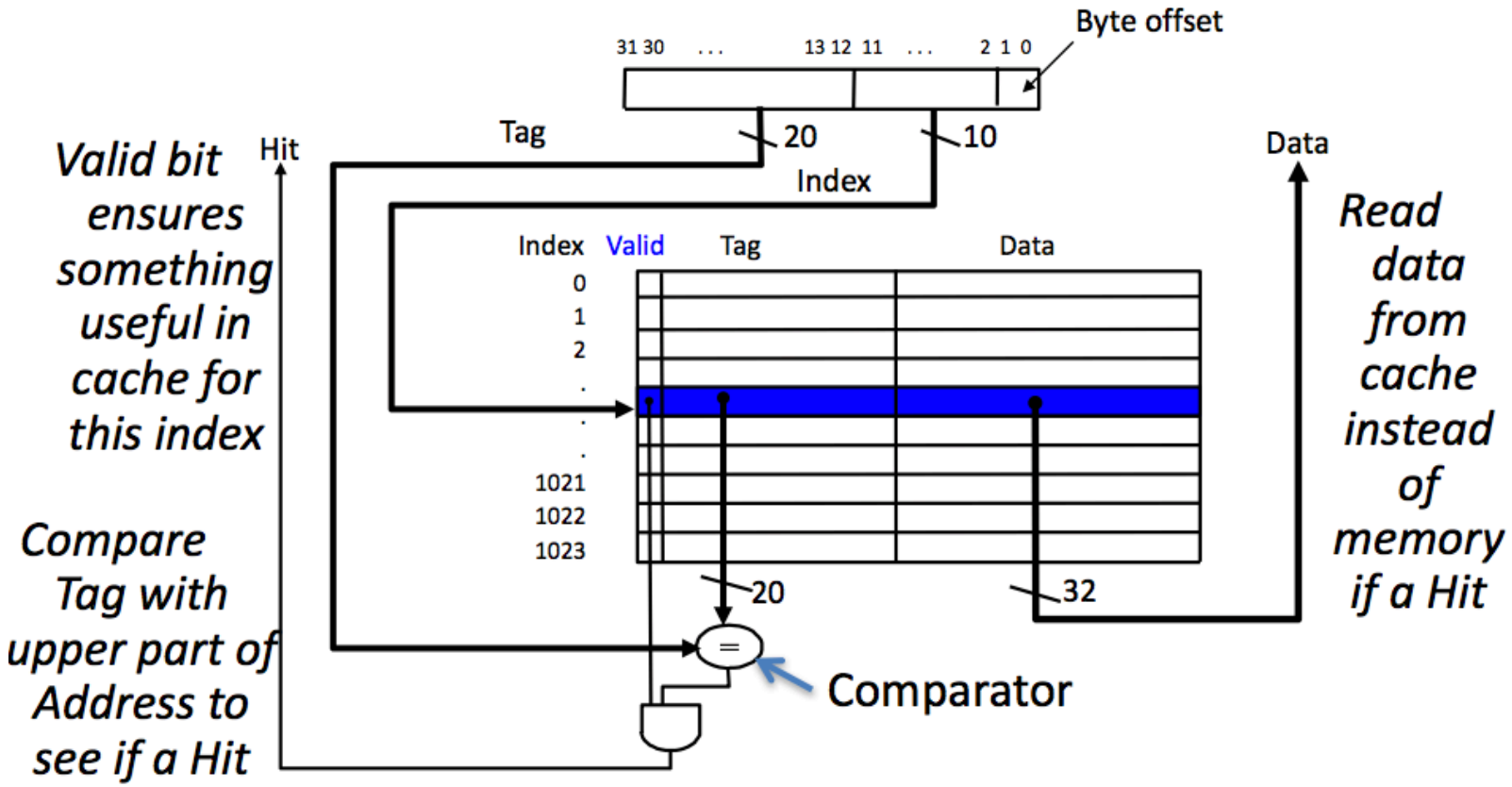
- Compulsory

- Conflict



- Capacity

(EASY!) DIRECT MAPPING CACHE CONT.



(EASY!) DIRECT MAPPING CACHE CONT.

- Determine address assignment.

1. Find # block => index width
2. Find # bytes per block => offset width
3. Tag width = address width – index width – offset width

- Note! You should consider the bit of your machine ! The word is 4byte in 32-bit machine and 8byte in 64-bit machine

- Total bits in a cache:
- Index * (valid + tag + data) -> write through… ? LRU ? dirty bit ? write back ?

SET ASSOCIATIVITY

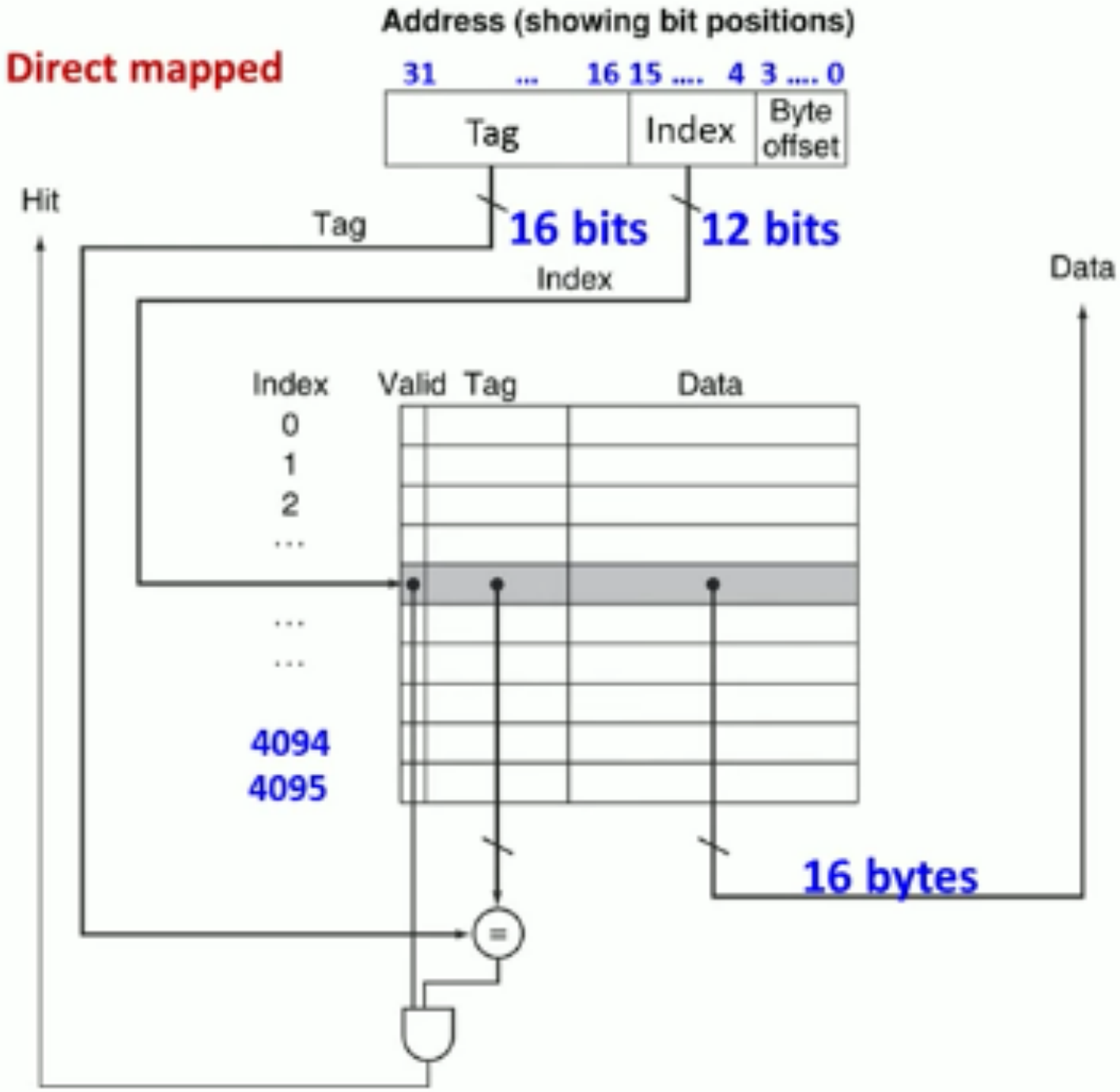
- Cache of 4096 blocks, 4-word block, 32-bit address.
- Tell me # sets, tag width for direct mapped, 2-way associative, 4 way associative, fully associative types.

☹ I don't wanna that!

SET ASSOCI

- Cache of 4096 blocks, 4-word l
- Tell me # sets, tag width for di
  associative, fully associative typ
- easy for direct mapping,

Size of the byte offset field = 4 bi
Size of the index field = 12 bits
Size of the tag field = 32-12-4 = 16
The total number of tag bits = 16



SET ASSOCIATIVITY

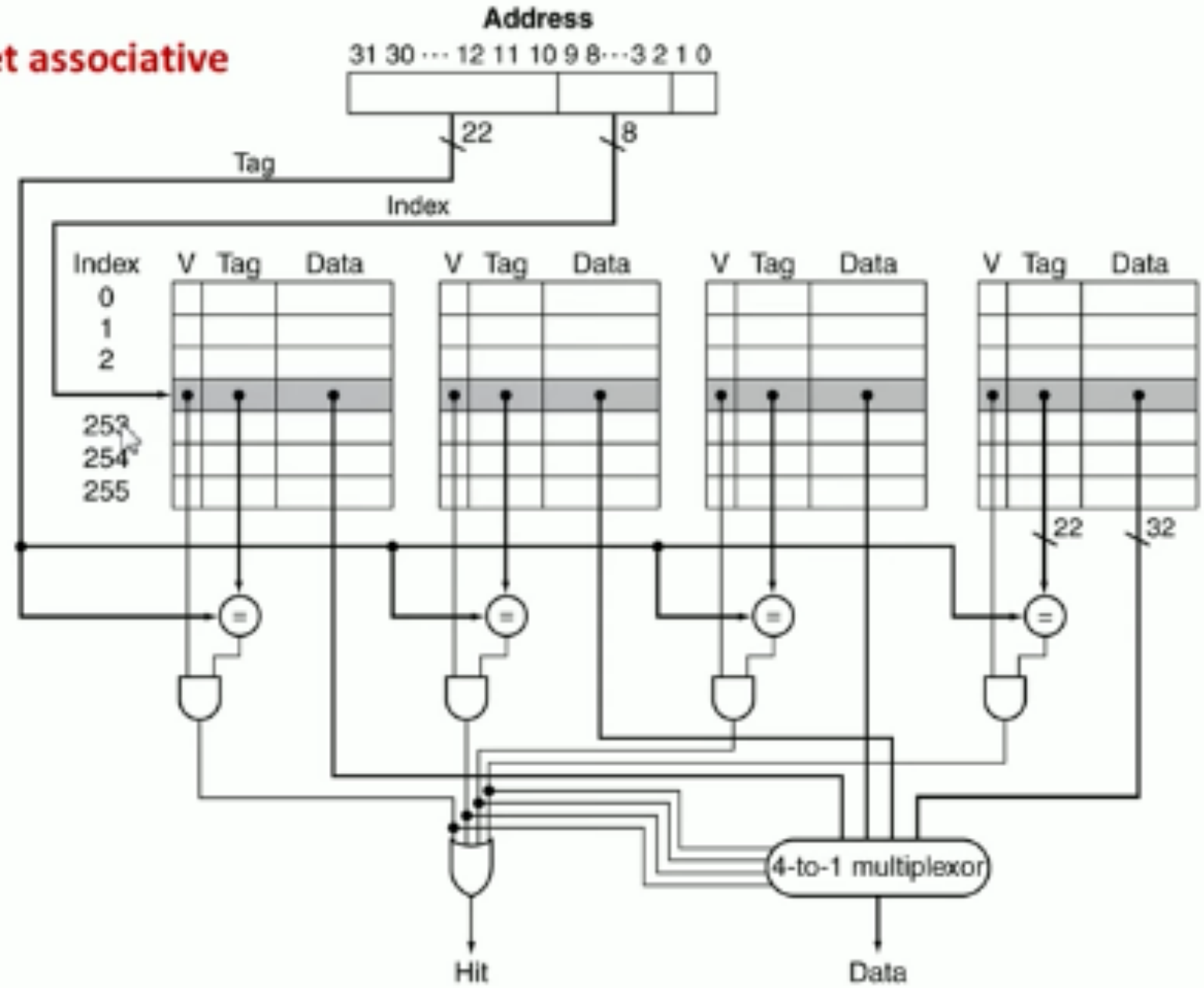
- Cache of 4096 blocks, 4-word block, 32-bit address.
- Tell me # sets, tag width for direct mapped, 2-way associative, 4 way associative, fully associative types.
- 2 way associative.

Size of the byte offset field = 4 bits
Size of the index field = 11 bits
Size of the tag field = 32-11-4 = 17 bits
The total number of tag bits = 17 x 4096 = 69632 bits

SET AS

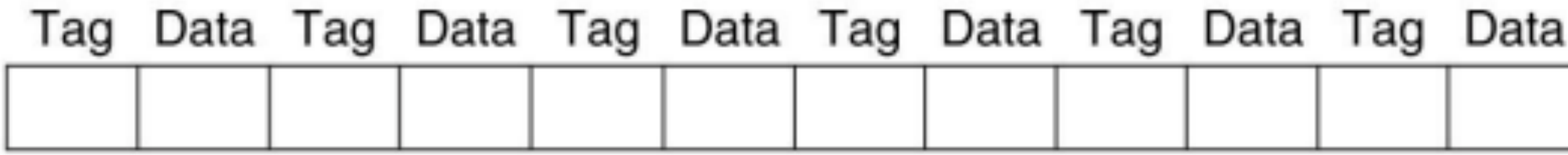
- Cache of 4096 bl
- Tell me # sets, ta
  associative, fully a
- 4 way associative

Size of the byte off
Size of the index fie
Size of the tag field
The total number o



SET ASSOCIATIVITY

- Cache of 4096 blocks, 4-word block, 32-bit address.
- Tell me # sets, tag width for direct mapped, 2-way associative, 4 way associative, fully associative types.
- fully associative



SET ASSOCIATIVITY

- Cache of 4096 blocks, 4-word block, 32-bit address.
- Tell me # sets, tag width for direct mapped, 2-way associative, 4 way associative, fully associative types.
- fully associative
- NO INDEXes ! YEAH !!!! We finished it !

AVERAGE MEMORY ACCESS TIME (AMAT)

- AMAT=hit time+miss rate×miss penalty

a) L1$ hits in 1 cycle (local miss rate 25%)
b) L2$ hits in 10 cycles (local miss rate 40%)
c) L3$ hits in 50 cycles (global miss rate 6%)
d) Main memory hits in 100 cycles (always hits)

Answer:

The AMAT is 1 + 25%×(10 + 40%×(50)) + 6%×(100) = 14.5 cycles.

Q&A

Thank you for attendance.