CACHE MEMORY

→ CPU fast + fetching instruction & data from memory → CPU does

↳ + Registers → CPU fast but content of registers is low.

↳ + Cache → CPU fast

slower than registers
store more than ↑

faster than main memory
have less content.

→ CACHE HIT : Look into cache and perceive that whatever required is present inside it.

Hit latency : Time taken to find whether required element is present in cache.

Cache Miss : If required data / instr^n is not present in cache.

Miss latency : Due to cache miss, main memory is used to get an element. Time taken in this is miss latency.

→ Page hit and Page fault refer to whether required data check is present in main memory or not. This is done

↳ by Page Tables.

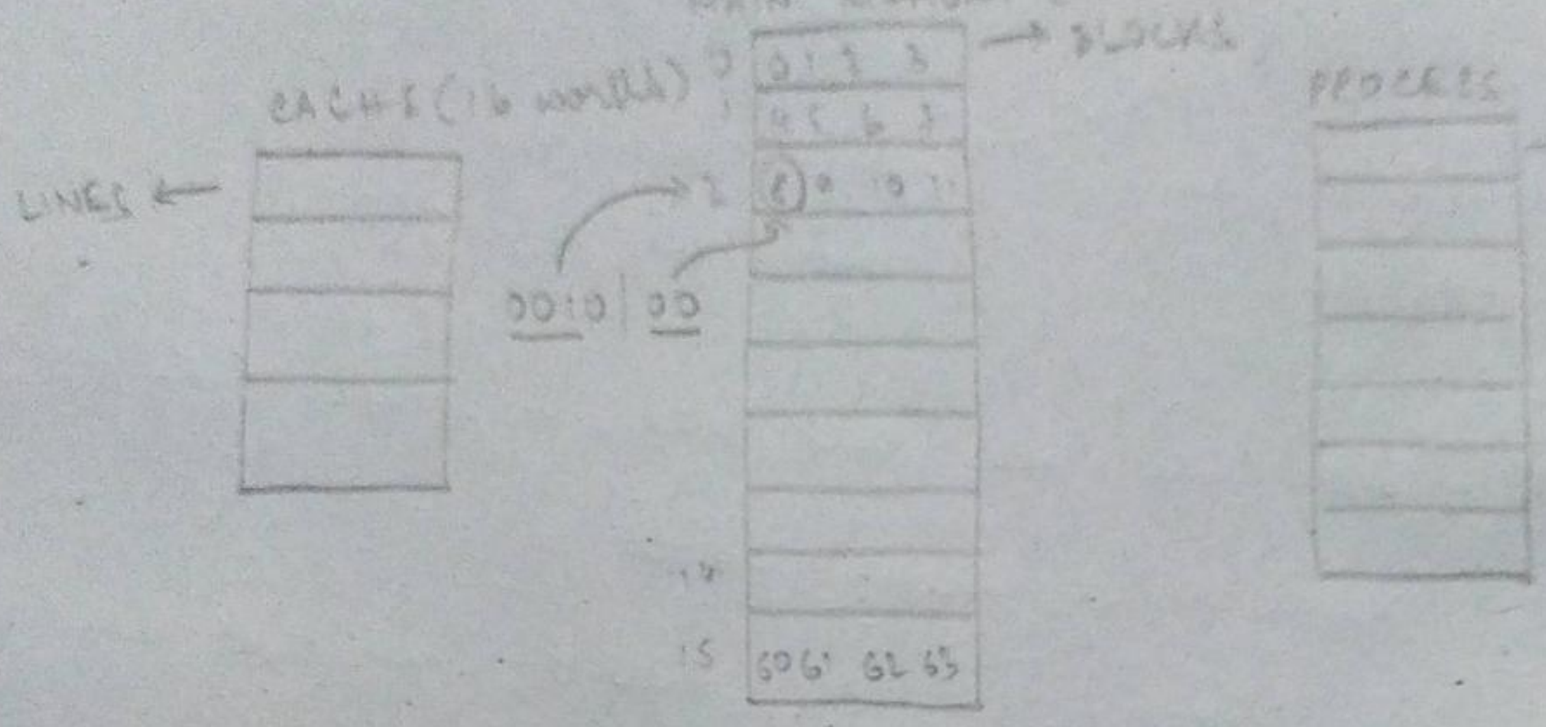
For cache, Tag directory is equivalent to Page Tables.

* what will be stored in cache is decided using

locality of reference (spatial & temporal)

DIRECT MAPPING

→ dividing the main memory into frames is conceptual.



Size of block ≡ Size of Lines

Block size = 4 words

block in MM = 16 , # 8 lines in cache = 4

MM contain 64 words, so, 6 bits are necessary to denote it.

Example: 0010 | 00

M.S.B denote block number i.e, which block to look at

L.S.B denote Block offset i.e, inside block, the index.

TAGS LINE #

00 - 0 -	0	4 8 12
01 - 1 -	1	5 9 13
10 - 2 -	2	6 10 14
11 - 3 -	3	7 11 15

→ Line number 0 also maps to block number 0, 4, 8 & 12.

This is because, L.S.B two bits of 0, 4, 8, 12 is 00, named as 0.

00 | 10 | 00

Tag #Line Block offset.

The CPU will first check the line number, then it will match the tag. If tag matches then it will go for Cache Hit.

PROBLEMS

①

MM size	Cache size	Block size	Tag bits	Tag directory size
128 KB	16 kB	256 B	3	3×2^6
32 GB	32 kB	1 kB	20	20×2^5
64 MB	512 kB	1 kB	7	7×2^9
16 GB	16 MB	4 kB	10	10×2^{12}
64 MB	64 KB	NA	10	NA
64 MB	512 KB	NA	7	NA

Assuming that memory is byte addressable.

128 KB, byte addressable, physical address contain 17 bits

128 KB, word addressable, physical address contain 15 bits.
↳ 32K W ↳ 4 byte

* For MM size 128 KB, cache size 16 kB & block size 256 B

PA = 17 bits

bits in
lines = 6 ($2^6 = \frac{2^{14}}{2^8} = \frac{16\,kB}{256\,B}$)

Block offset = 8 ($2^8 = 256$)

Tag bits = 17 − 8 − 6 = 3.

Tag dir size = # Tag bits × # lines = 3×2^6