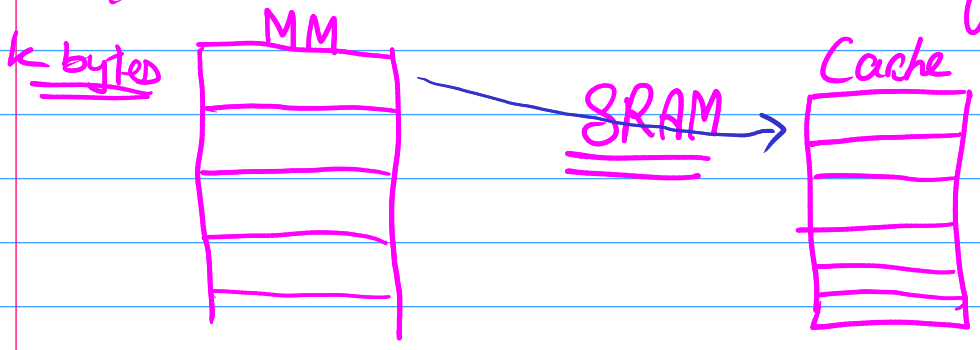
Cache organization

Technique to map a location in the main memory to a location in the cache memory



What happens if a particular data or instruction is not found in the cache?

Tag | Valid | D | Data

Cache hit
Cache miss

CPU time = no. of CC × CCT

Mem stall cycles = no. of misses × miss penalty

$= IC \times \frac{\text{misses}}{\text{Instruction}} \times \text{miss penalty}$

$= IC \times \frac{\text{mem access}}{\text{instr}} \times \frac{\text{miss}}{\text{mem acc}} \times \text{miss penalty}$

Ex CPI = 1, when all memory accesses hit in the cache. If 30% loads and stores. Miss penalty is ~~Total cycles~~ 100 cycles & miss rate = 5%
How much faster the computer will be if all instructions were cached

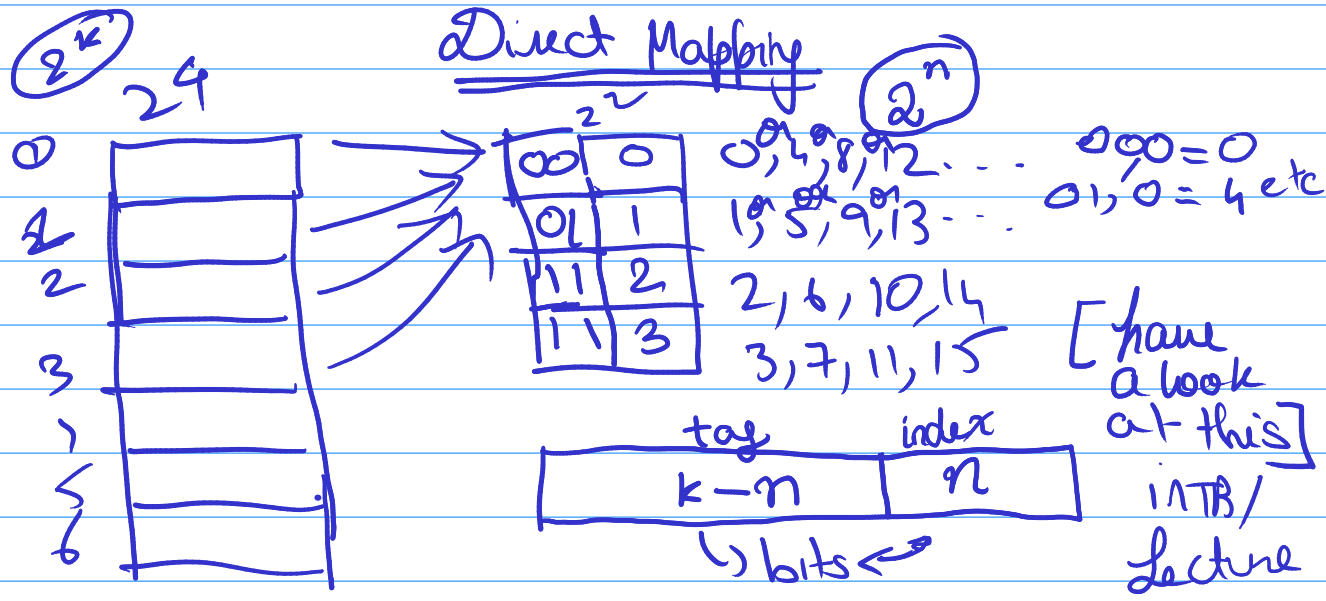
CPU time = Instn x CPI x Clock cycle time
[if all cached]

= IC x 1 x CCT

Cache IC x $\frac{0.3}{1}$ x 0.05 x 100 x CCT

= 1.5 IC x CCT

Overall CPI = 2.5 IC



[have a look at this] INTB/ Lecture

Problems: (1) Collision → replacement strategy

(2) Valid bit

(Another process or program occupying same location)

(3) SW ⇒ Store to cache & main

Location Update

- update @ end ?
- update everytime ?
  - no dependency of another program → Write back / Write once
  - → when 2 programs have shared resource → Write-through

(3) Location update solved using dirty bit

Init state

Index	V	Tag	Data	D
000	N			
001	N			
⋮	⋮			

(1) Address 10110 (miss) referred

Index	V	Tag	Data
000	N		
⋮			
110	Y	10	Mem 10110

(2) Address referred 11010 (*miss*):

Index	V	Tag	Data
000	N		
001	N		
010	Y	11	Mem(11010)
011	N		
100	N		
101	N		
110	Y	10	Mem(10110)
111	N		

replace

(4) Address referred 10010 (*miss*):

Index	V	Tag	Data
000	N		
001	N		
010	Y	10	Mem(10010)
011	N		
100	N		
101	N		
110	Y	10	Mem(10110)
111	N		

(3) Address referred 10110 (*hit*):

Index	V	Tag	Data
000	N		
001	N		
010	Y	11	Mem(11010)
011	N		
100	N		
101	N		
110	Y	10	Mem(10110)
111	N		

to CPU

dirty bit allows the OS (or hardware) to check if the cache data is written in the main memory or not.