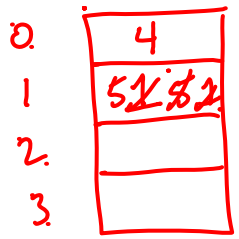
Where to place a block in the cache



Index = Address % 4

Addr:	1	4	1	5	1	5	.....
Index:	1	0	1	1	1	1	
	M	M	H	M	M	M	

Direct Mapped = a data item can reside in only 1 location

Associativity

↓

allows data to reside in multiple locations

- Reduce number of misses that occur because of conflicts

Example

2-way set associative

0	4	
1	~~4~~ 9	5
2	2	6
3		

Index = Addr % 4

- if both empty, pick one way to place data

Addresses:

	1	4	5	2	1	4	5	6	1	4	5	2	9
Index:	1	0	1	2	1	0	1	2	1	0	1	2	1
	M	M	M	M	H	H	H	M	H	H	H	H	

Hit Rate = 7/12

Replacement policy

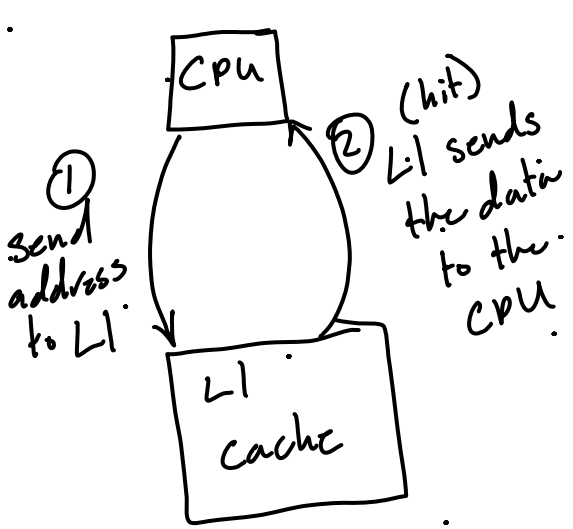
- LRU: (Least Recently Used) Replace the element used longest ago.
- Random: Randomly select a way to replace.

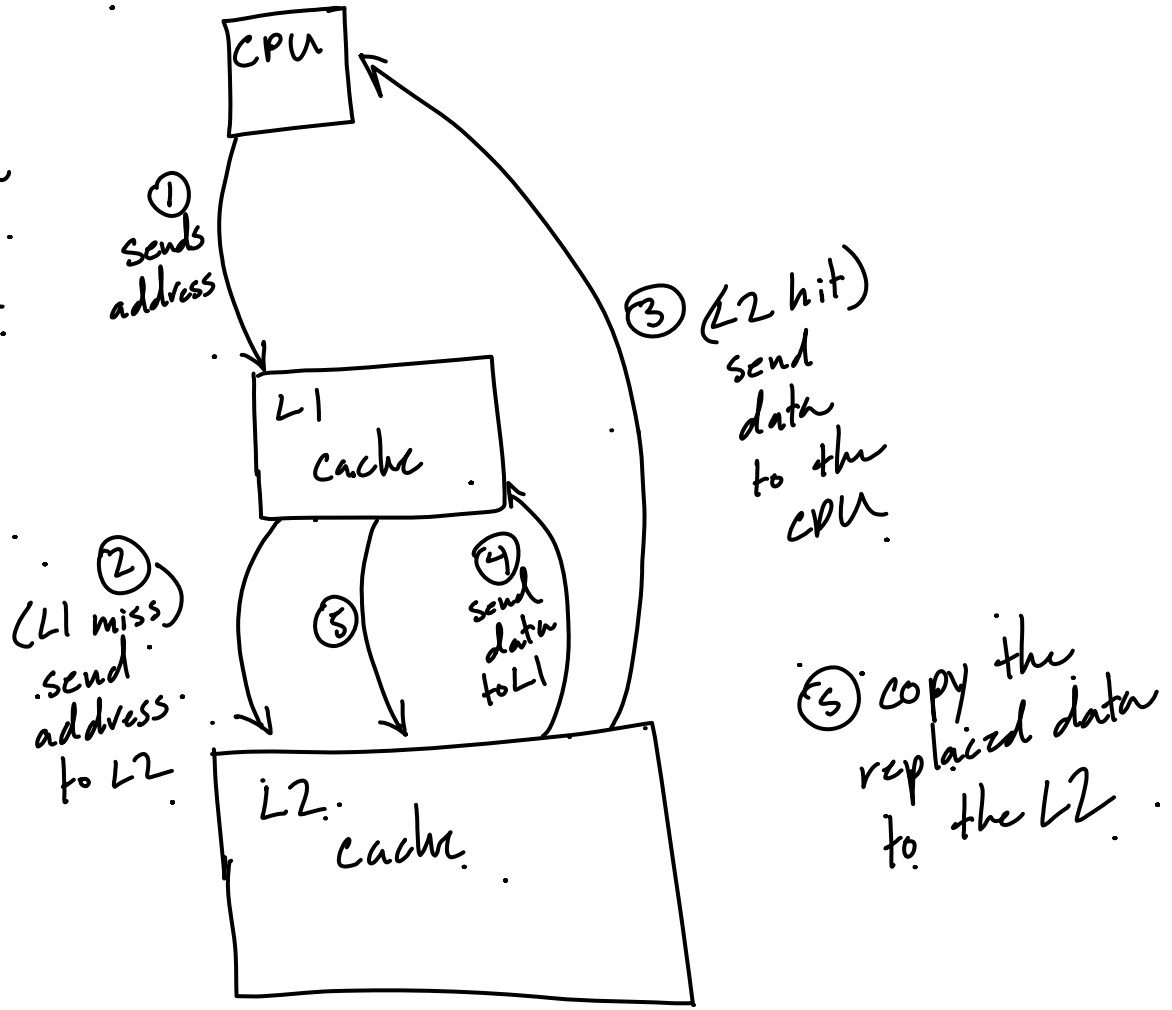
Example

For a particular program running on a pipelined processor, the data cache hit rate is 95%. The miss penalty is 50 cycles. If 30% of the instructions access memory, what is the average CPI of the program?

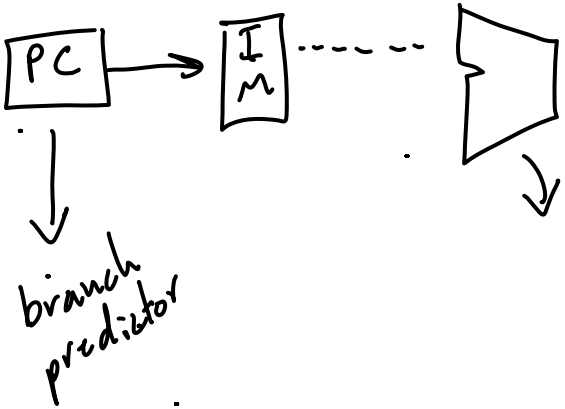
What is the speedup, if the cache configuration is modified so that the hit rate becomes 97% ?

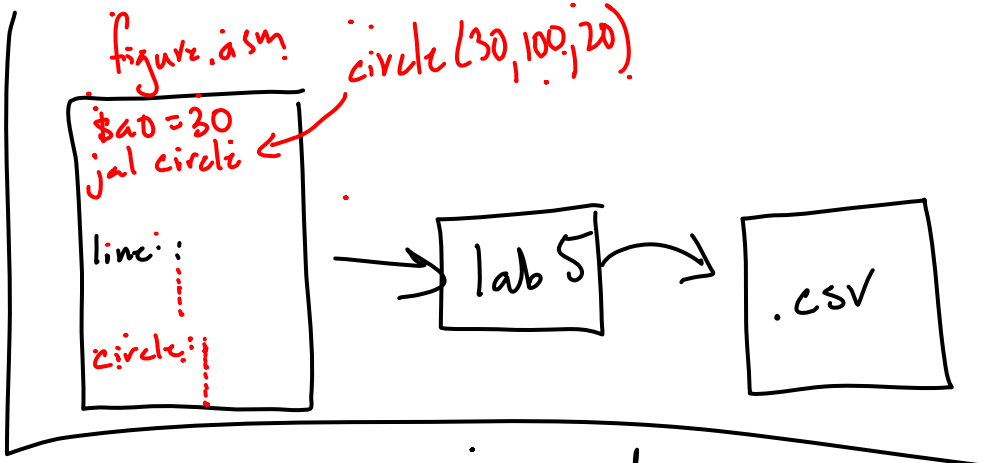
Multilevel caching





Lab 5



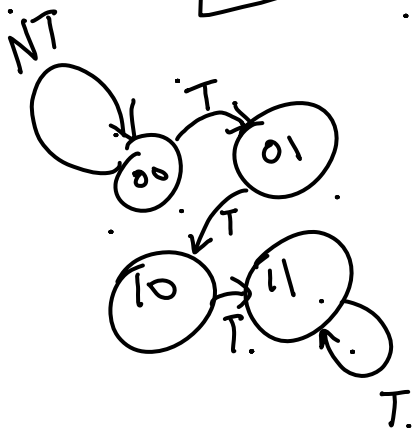


GHR

0	1

← shift in

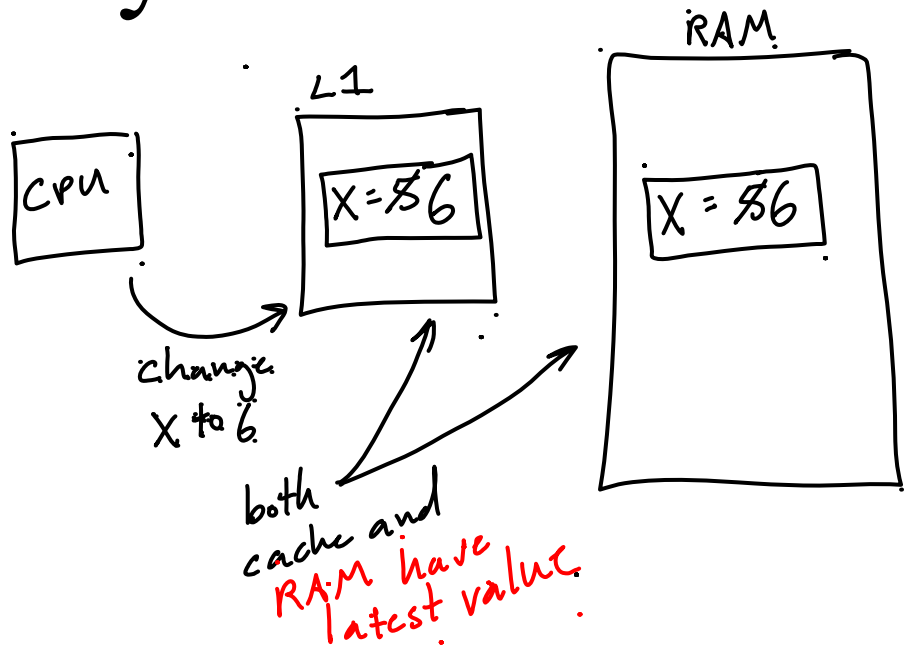
→ 0	0	1
1	0	1
→ 2	0	1
3	0	0



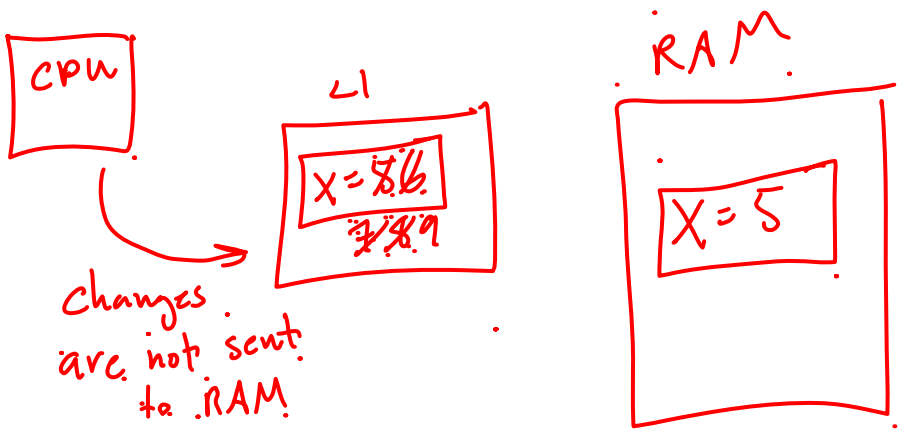
actual outcome	predicted outcome	
T	NT	incorrect
T	NT	incorrect
NT	NT	correct
T	NT	incorrect

Writing to memory with caches

- Write-through –
  cache and RAM have updated values



- Write-back –
  - writes are not sent to RAM
  - RAM is updated when the cached value is replaced



1-way

Show the hits and misses for a direct-mapped cache with a block length of 4-words and a total size of 16 words. The cache is word addressable (there is no byte-offset).

	M	M	M	H	M	M
Addresses:	1	4	8	5	20	17
Bl. Off.:	1	0	0	1	0	1
Index:	0	1	2	1	1	0

Hit Rate = 1/6

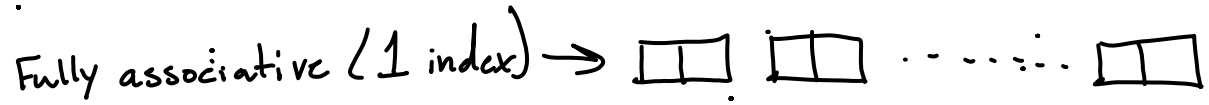
Index	0	1	2	3
0	~~0~~ 16	~~1~~ 17	~~2~~ 18	~~3~~ 19
1	~~4~~ 20	~~5~~ 21	~~6~~ 22	~~7~~ 23
2	8	9	10	11
3				

Bl. Off.

Bl. Off. = Addr. % 4

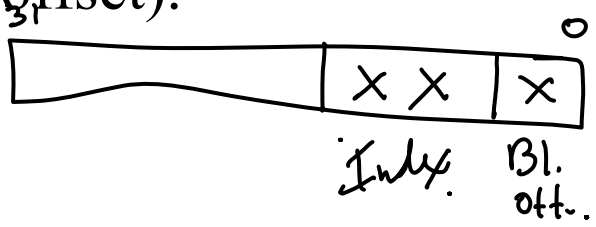
Index = $\left(\frac{\overset{20}{\overset{\cancel{16}}{17}}}{4}\right)$ % 4

- no class Wed.
- Lab 5 extra credit (handin by 11/24 Mon.) — lab5_ec
  - top 1/3 or 5 submissions (whichever is higher)
    - 3% extra credit on final and pizza



Show the hits and misses for a 2-way associative cache with a block length of 2-words and a total size of 16 words. The cache is word addressable (there is no byte-offset).

	M	M	M	H	M	M
Addresses:	1	4	8	5	20	17
Bl. off.:	1	0	0	1	0	1
Index:	0	2	0	2	2	0



Bl. off. = Addr % 2

Index = $\left(\frac{\text{Addr}}{2}\right)$ % 4