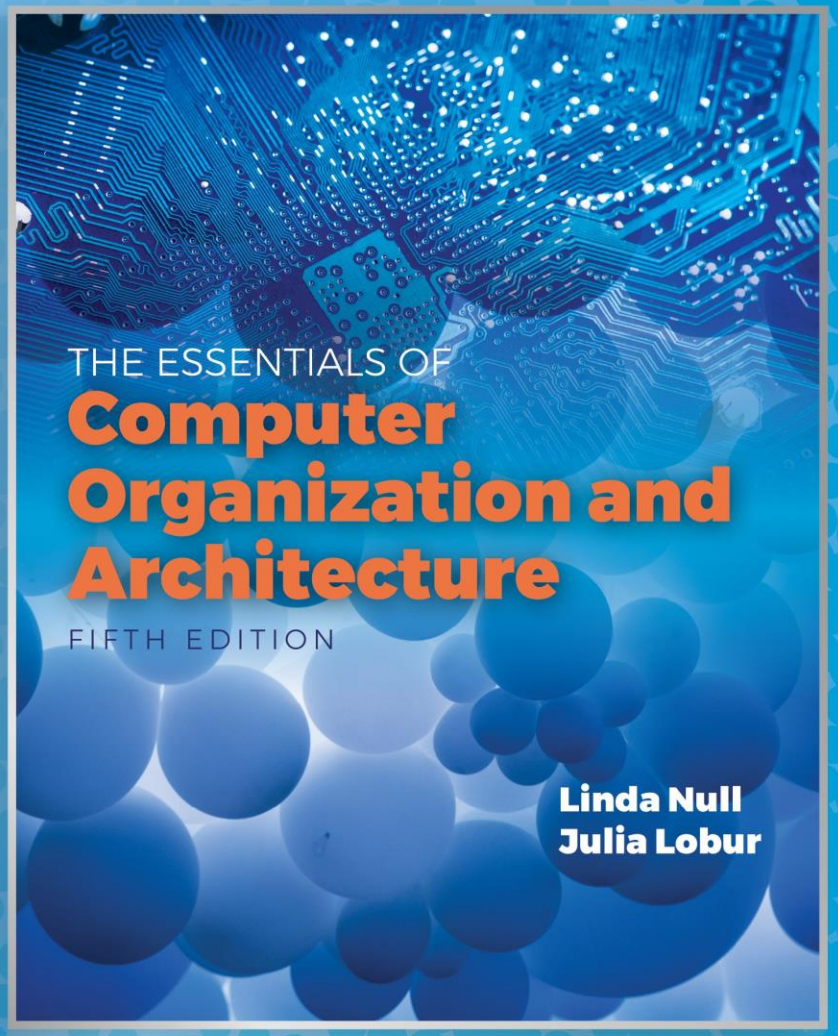
This is the fifth lecture of Chapter 6

Chapter 6

Memory (E)

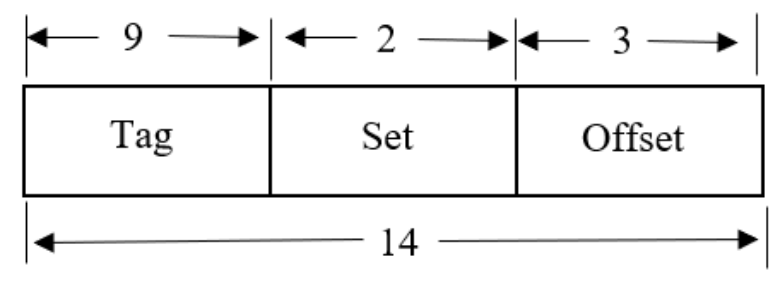


Quick review of last lecture

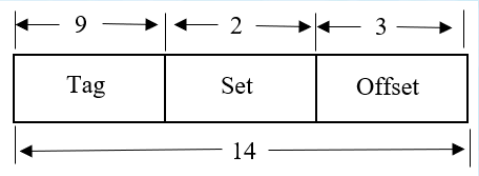
- Cache Replacement Policy
  - LRU, FIFO, Random
- Effective access time (EAT)
  - $\mathbf{EAT = \mathit{H} \times Access_{\mathit{C}} + (1 - \mathit{H}) \times Access_{\mathit{MM}}}$
- Cache Read and Write Policy
  - Load through, Write through, Write back (dirty block), Write Allocate, Write No-Allocate
- Unified Cache vs Harvard Cache
- Victim Cache and Trace Cache

Example: Cache Hit Rate

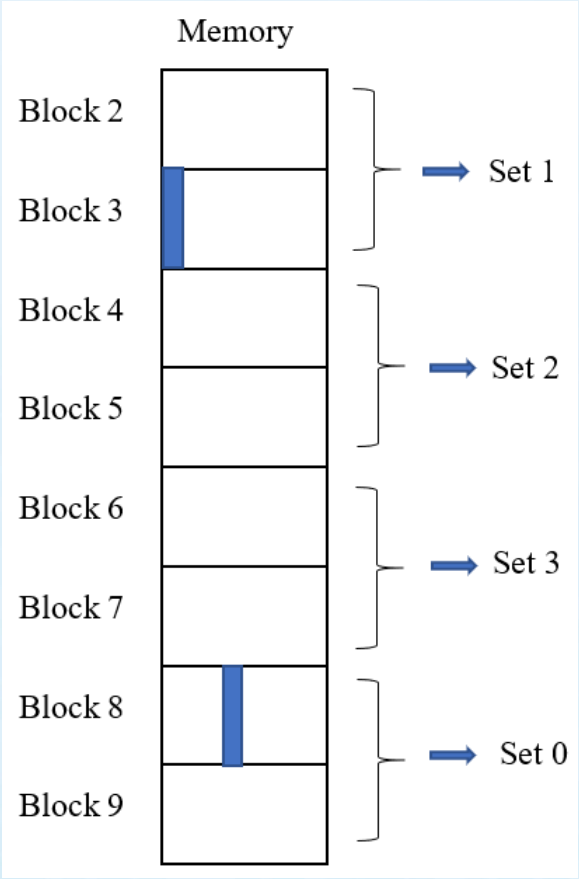
- A 2-way set-associative cache consists of four sets. Main memory contains 2K blocks of eight bytes each and byte addressing is used.
  a) Show the main memory address format that allows us to map addresses from main memory to cache. Be sure to include the fields as well as their sizes.
  b) Compute the hit ratio for a program that loops 10 times from addresses 0x18 to 0x43 in main memory.
- First, address format



Example: Cont.



b) Compute the hit ratio for a program that loops 10 times from addresses 0x18 to 0x43 in main memory

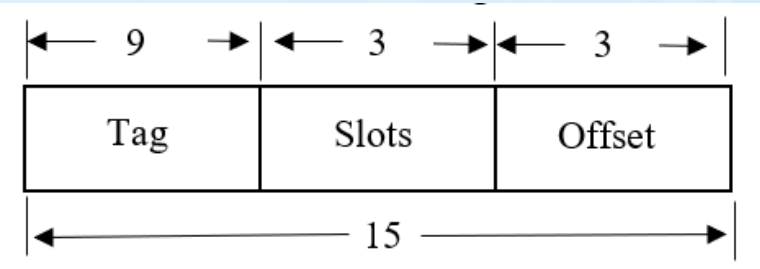


1. Starting block
2. Ending block
3. Check to see if replacement is needed
4. Total # of memory accesses
5. # of misses
6. # of hits
7. Hit rate …

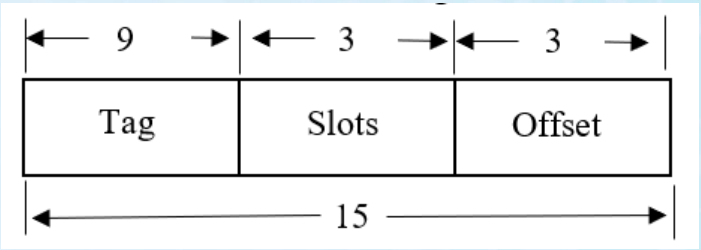
Example: Cache Hit Rate and EAT

- A direct-mapped cache consists of 8 blocks. A byte-addressable main memory contains 4K blocks of 8 bytes each.
  - Initially, the cache is empty

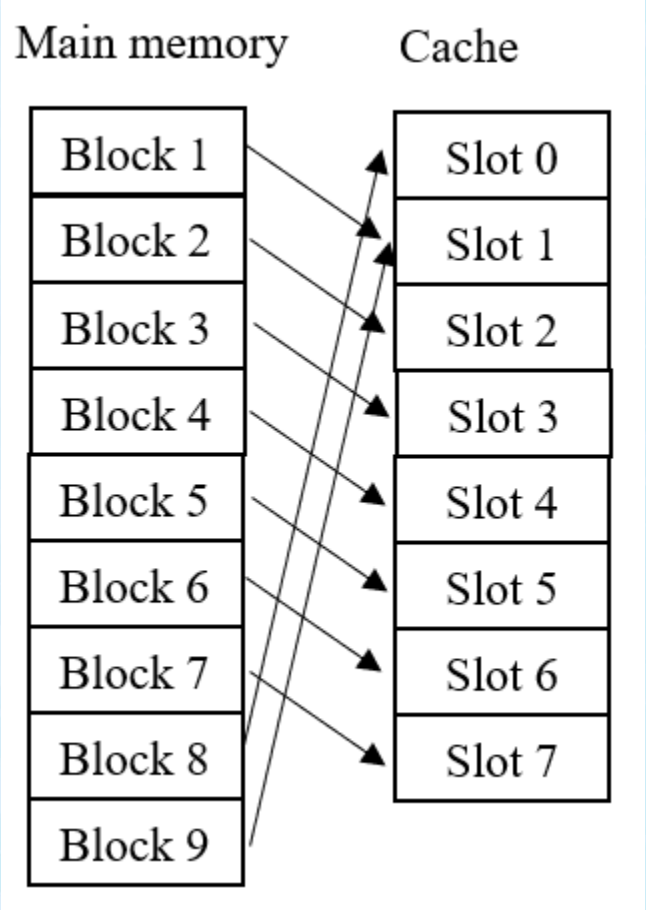
- So, the main memory address format



- Compute the hit ratio for a program that loops 5 times from locations 8 to 75_{10} in memory



1. Starting block	1	Misses
2. Ending block	9	5
3. Check to see if replacement is needed	Yes	1
4. Total # of accesses	340	1
5. # of misses	17	1
6. # of hits	323	1
7. Hit rate …	95%	5



Example cont.

- Access time for the cache is 22 ns and the time required to fill a cache slot from main memory is 300 ns.
- If a block is missing from cache, the entire block is brought into the cache and the access is restarted
  - Not load-through

- So, EAT is

  .95(22ns) + .05(300ns + 22ns) = 20.9 + 16.1 = 27 ns