Computer Architecture

Fall, 2025
Week 13
2025.11.24

組別：_____ **簽名：**_____

[group 1]

1. Please list the pros and cons of SRAM and DRAM. And where are they used for.

Ans:

SRAM

Pros: Fast, no need to refresh (data can last forever with power)

Cons: Expensive, low density (larger area), high power consumption.

Used for caches.

DRAM

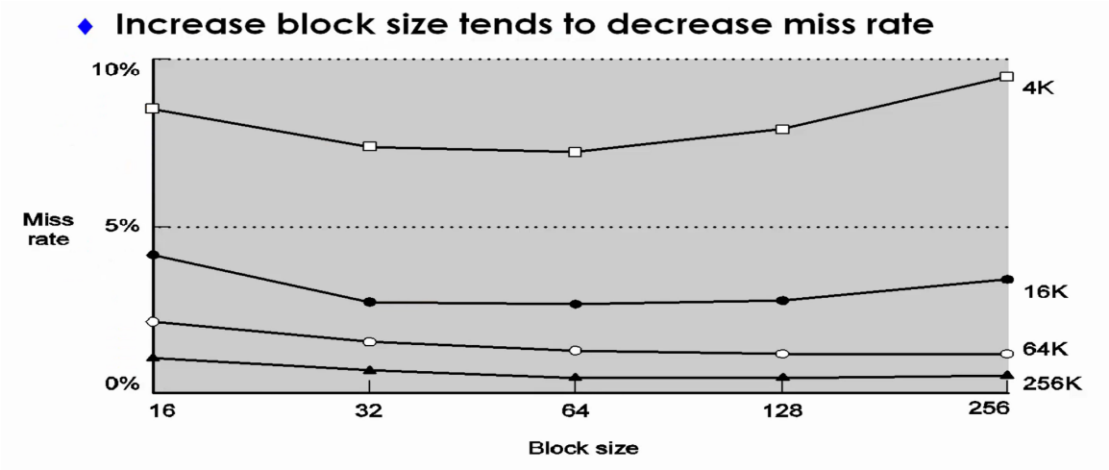
Pros: Cheap, high density (smaller area), low power consumption.

Cons: slow, need to be refresh.

Used for main memory.

[group 2]

2. Why doesn't a much larger block size increase the miss rate when the cache size increases?



Ans.

1. Even if the block size increases, the cache can still store a lot of blocks since the increased cache capacity. Therefore, it significantly reduced capacity misses
2. The disadvantage of a larger cache, pollution, is absorbed by a "large capacity". Since the large cache adds more useless information is not a problem.
3. Spatial locality remains effective. The large blocks can still prefetch nearby data, reducing compulsory misses.

[group 12]

3. Please arrange the following components in order according to their distance from the processor, from farthest to nearest:

   a. Disk

   b. Tape

   c. Register

d. Memory

e. Cache

Ans:

Tape, Disk, Memory, Cache, Register.

[group 3]

4. True / False

   Q1. Using a larger cache block size usually helps exploit spatial locality.

   Q2. In a direct-mapped cache, one memory block can be placed in multiple cache locations.

   Q3. A higher cache hit rate generally improves CPU performance.

   Q4. A valid bit of 0 means the cache line does not contain valid data.

   Q5. With a write-through policy, a write hit updates both the cache and the main memory.

   Q6. Temporal locality means that once a piece of data is accessed, it is unlikely to be accessed again soon.

Ans:

1. T, Spatial locality means nearby addresses are likely accessed soon, so bigger blocks bring more nearby data at once.
2. F, A direct-mapped cache allows **exactly one** possible cache location for each memory block.

3. T, A hit is fast; a miss is slow. More hits → fewer expensive memory accesses → better performance.
4. T, when valid bit = 0, the cache entry is empty or unusable.
5. T, write-through means every write is immediately copied to memory.
6. F, Temporal locality means the *opposite* — recently used data is likely to be used again.

[group 4]

5. For a direct-mapped cache design with a 32-bit address, based on the following bits of address, what is the ratio between total bits required for such a cache implementation over the data storage bits?

tag	index	byte offset
31-10	9-4	3-0

Ans:

1 + [(22+1) / (8 * 16)] = 1.1796 ~ 1.18

22 : tag

1 : valid

16 : bytes per block

$$\frac{\text{data bits} + \text{tag bits} + \text{valid bit}}{\text{data bits}} = 1 + \frac{\text{overhead bits}}{\text{data bits}}$$

[group 5]

6. In the context of a Write Hit, explain the difference between a Write-Through policy and a Write-Back policy. What is the primary performance drawback of the Write-Through method, and what hardware mechanism is typically introduced to mitigate this specific drawback?

Ans:

Difference in Policy:

Write-Through: When a write hit occurs, the information is updated in both the cache and the main memory simultaneously to ensure data consistency.

Write-Back: When a write hit occurs, only the block in the cache is updated. The main memory is not updated immediately. The system tracks modified blocks (often using a dirty bit), and the data is only written back to the main memory when that specific dirty block is replaced.

Drawback of Write-Through:

The primary disadvantage is that it increases traffic to the memory and makes write operations take longer because the CPU may have to wait for the slower main memory to complete the write.

Mitigation Mechanism:

To mitigate the stalling caused by Write-Through caches, a Write Buffer is employed. This buffer acts as a temporary queue that holds data destined for memory. It allows the CPU to deposit the data and resume execution immediately, rather than waiting for the slow write to main memory to complete. The CPU only needs to stall if this buffer becomes full.

[group 9]

7. True or False

    1) In a direct-mapped cache, each memory block can be placed in only one specific cache block.
    2) If the block size of a cache increases, the number of index bits also increases.
    3) A cache miss always occurs when the valid bit of a cache block is 0.
    4) A conflict miss can occur even if the cache size is large enough to hold the entire working set.
    5) The tag field of a cache address is used to determine the block offset within a cache block.

Ans:

1) True.
2) False (Explanation: Larger block size → *fewer* blocks → *fewer* index bits.)

3) True.

4) True (Direct-mapped caches can force two active blocks to compete for the same index.)

5) False (Tag identifies *which memory block* is stored; offset determines the byte/word inside the block.)

[group 10]

8. Consider the following code:

```
int a[1024];
for(int i=0; i<1024; i+=4){
    s+=a[i];
}
```

We have 4KB directed-mapped cache with 64 blocks and the array starts at 0x0.

1) Which type of locality does the loop use (temporal, spatial or both) ?

2) Calculate the hit rate.

Ans:

1) Spatial

2) 75%

[group 9]

9. Define hit time, hit rate, miss penalty, and miss rate in a memory hierarchy.

Ans：

- Hit rate: fraction of memory access found in the upper level

- Hit time: time to access the upper level (RAM access time + Time to determine hit/miss)
- Miss Rate = 1 - (Hit Rate)
- Miss Penalty: time to access a block in the lower level + time to deliver the block to the processor (latency + transmit time)

[group 14]

10. What's the use of the valid bit and dirty bit within the cache?

Ans:

valid bit:

Indicates whether the cache block contains valid data from memory.

dirty bit:

Indicates whether the cache block has been modified. If the block is dirty, it must be written back to the next level of memory when it is evicted.

[group 7]

11. Determine whether the following statements regarding Memory Hierarchy are **True or False.**

    1) The "Principle of Locality" suggests that a program accesses all parts of the memory address space evenly and randomly at any given time.
    2) SRAM is typically used for Cache because it is faster, whereas DRAM is used for Main Memory because it is denser (more bits per area) and cheaper.
    3) In a memory hierarchy, data can be copied directly from Disk to Cache, skipping Main Memory entirely.

4) "Miss Penalty" refers to the time it takes to replace a block in the upper level plus the time to deliver it to the processor.

Ans:

1) False, The Principle of Locality states that programs tend to access a relatively small portion of the address space at any instant of time (the 90/10 rule: 90% of time is spent in 10% of the code). This includes both Temporal Locality (recently accessed items will be accessed again) and Spatial Locality (items near each other will be accessed together).
2) True, SRAM is faster but more expensive and less dense (fewer bits per chip area), making it suitable for cache. DRAM is slower but much cheaper and denser, making it ideal for larger main memory.
3) False, In memory hierarchy, data is copied only between adjacent levels. Data must flow: Disk ↔ Main Memory ↔ Cache ↔ Processor. Each level serves as a "buffer" for the next higher level. You cannot skip levels.
4) True, Miss Penalty = time to replace a block in the upper level + time to deliver the block to the processor. This is the extra time required compared to a hit.