



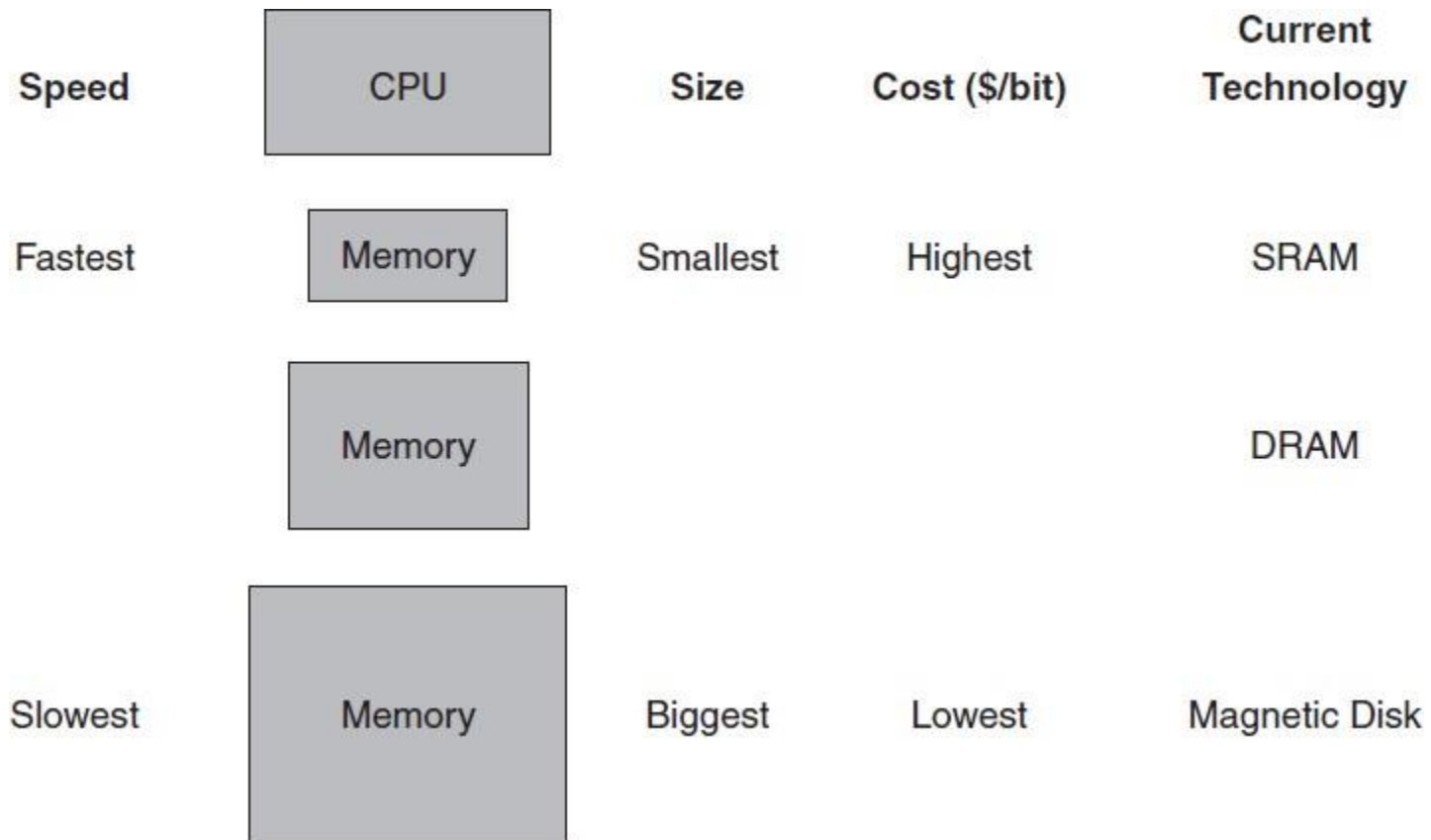
Lecture – 19



Memory Hierarchy

- A memory hierarchy consists of multiple levels of memory with different speeds and sizes.
- The faster memories are more expensive per bit than the slower memories and thus smaller.
- Three technologies used in building memory hierarchies:
 1. DRAM (Main memory)
 2. SRAM (Cache)
 3. Magnetic disk (Hard disk)

The Basic Structure of Memory Hierarchy

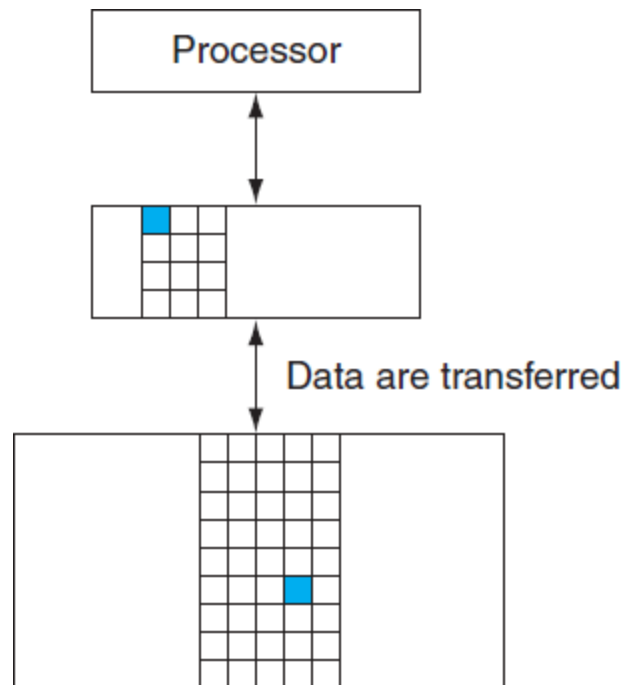


Memory Hierarchy

Memory Technology	Typical Access Time	\$ Per GB in 2008
SRAM	0.5 – 2.5 ns	2000 – 5000
DRAM	50 – 70 ns	20 – 75
Magnetic Disk	5,000,000 – 20, 000,000 ns	0.20 – 2

- The goal is to provide the user with as much memory as is available in the cheapest technology, while providing access at the speed offered by the fastest memory.

Two level Hierarchy



Terminology

- **Block:**

The minimum amount of information that can be either present or not present in the two level hierarchy.

- **Hit:**

Data requested by the processor appears in some block in the upper level.

- **Miss:**

Data requested by the processor is not present in the upper level.

- **Hit rate / Hit ratio:**

The fraction of memory accesses found in the upper level.

- **Miss rate:**

The fraction of memory accesses not found in the upper level. (1-Hit rate)

Terminology

- **Hit Time:**

The time needed to access a level of the memory hierarchy, including the time required to determine whether the access is a hit or a miss.

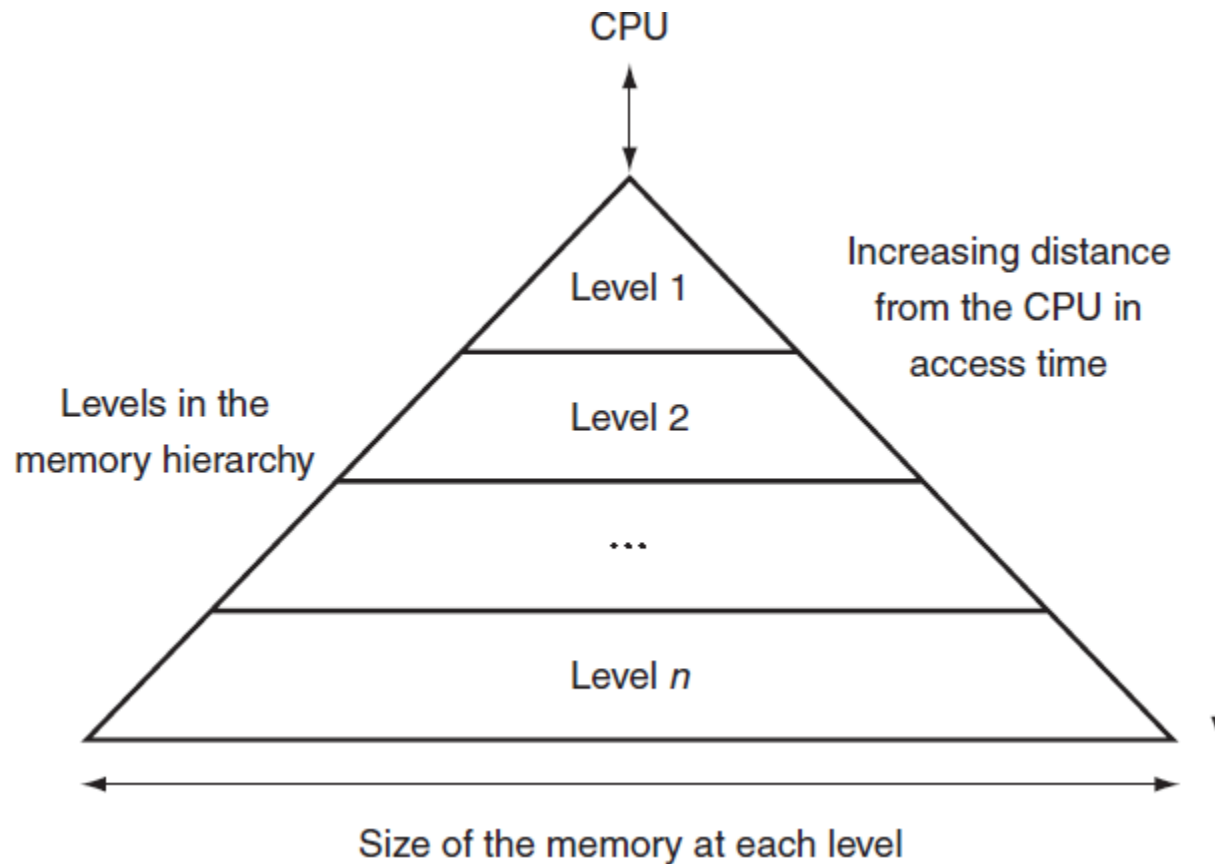
- **Miss Penalty:**

The time to replace a block in the upper level with the corresponding block from the lower level, plus the time to deliver this block to the processor.

Principle of Locality

- It states that program access a relatively small portion of their address space at any instant of time.
- Temporal Locality:
If an item is referenced, it will tend to be referenced again soon.
- Spatial Locality:
If an item is referenced, items whose addresses are close by will tend to be referenced soon.

Memory Hierarchy



Cache

- It refers to the level of memory hierarchy between the processor and main memory.

X_4
X_1
X_{n-2}
X_{n-1}
X_2
X_3

a. Before the reference to X_n

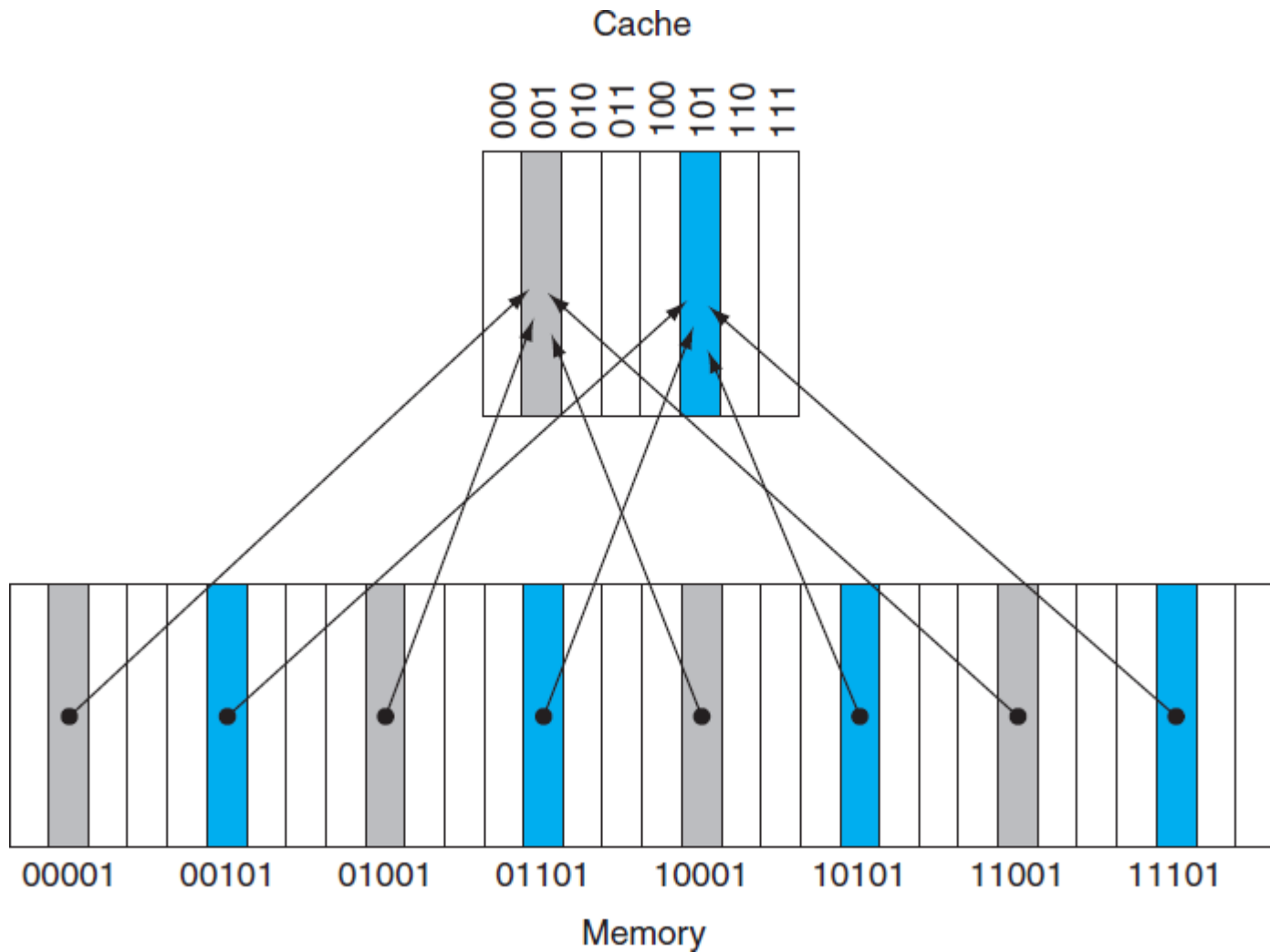
X_4
X_1
X_{n-2}
X_{n-1}
X_2
X_n
X_3

b. After the reference to X_n

Direct Mapped Cache

- A cache structure in which each memory location is mapped to exactly one location in the cache.
- Assign cache location based on the address of the word in the memory.
- **Mapping:**
(Block address) modulo (Number of cache blocks in the cache).
- Can accessed directly with the lower order bits.
- Each cache location can contain the contents of a number of different memory locations.

A Direct Mapped Cache





Tag and Valid Bit

- A field contains the address information required to identify whether a word in the cache corresponds to the requested word.
- It indicates that the associated block contains valid data.

Accessing A Cache

Decimal address of reference	Binary address of reference	Hit or miss in cache	Assigned cache block (where found or placed)
22	10110_{two}	miss (7.6b)	$(10\mathbf{110}_{\text{two}} \bmod 8) = \mathbf{110}_{\text{two}}$
26	11010_{two}	miss (7.6c)	$(11\mathbf{010}_{\text{two}} \bmod 8) = \mathbf{010}_{\text{two}}$
22	10110_{two}	hit	$(10\mathbf{110}_{\text{two}} \bmod 8) = \mathbf{110}_{\text{two}}$
26	11010_{two}	hit	$(11\mathbf{010}_{\text{two}} \bmod 8) = \mathbf{010}_{\text{two}}$
16	10000_{two}	miss (7.6d)	$(10\mathbf{000}_{\text{two}} \bmod 8) = \mathbf{000}_{\text{two}}$
3	00011_{two}	miss (7.6e)	$(00\mathbf{011}_{\text{two}} \bmod 8) = \mathbf{011}_{\text{two}}$
16	10000_{two}	hit	$(10\mathbf{000}_{\text{two}} \bmod 8) = \mathbf{000}_{\text{two}}$
18	10010_{two}	miss (7.6f)	$(10\mathbf{010}_{\text{two}} \bmod 8) = \mathbf{010}_{\text{two}}$

Accessing A Cache

Index	V	Tag	Data
000	N		
001	N		
010	N		
011	N		
100	N		
101	N		
110	N		
111	N		

a. The initial state of the cache after power-on

Index	V	Tag	Data
000	N		
001	N		
010	N		
011	N		
100	N		
101	N		
110	Y	10 _{two}	Memory(10110 _{two})
111	N		

b. After handling a miss of address (10110_{two})

Index	V	Tag	Data
000	N		
001	N		
010	Y	11 _{two}	Memory (11010 _{two})
011	N		
100	N		
101	N		
110	Y	10 _{two}	Memory (10110 _{two})
111	N		

c. After handling a miss of address (11010_{two})

Index	V	Tag	Data
000	Y	10 _{two}	Memory (10000 _{two})
001	N		
010	Y	11 _{two}	Memory (11010 _{two})
011	N		
100	N		
101	N		
110	Y	10 _{two}	Memory (10110 _{two})
111	N		

d. After handling a miss of address (10000_{two})

Accessing A Cache

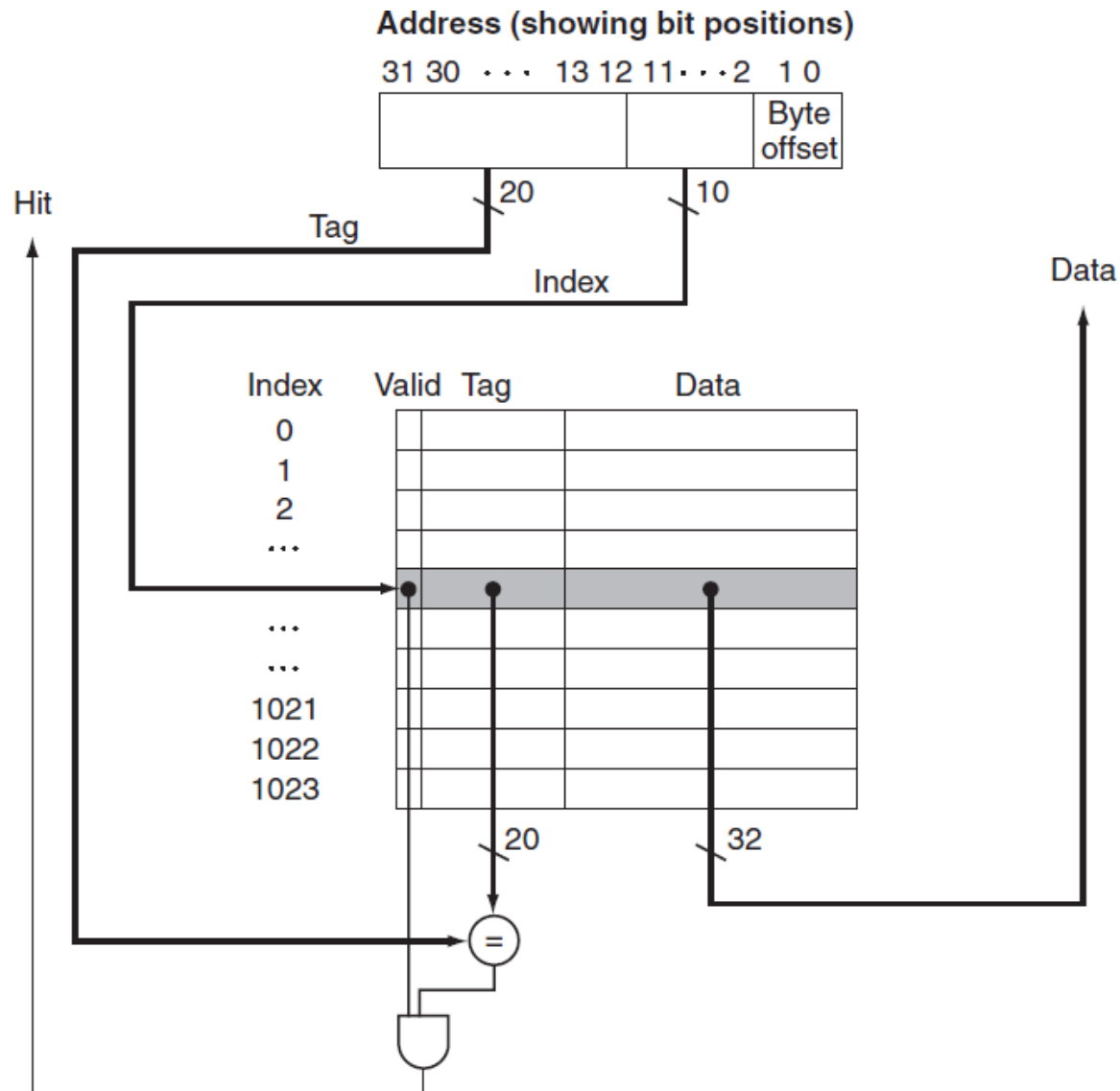
Index	V	Tag	Data
000	Y	10 _{two}	Memory (10000 _{two})
001	N		
010	Y	11 _{two}	Memory (11010 _{two})
011	Y	00 _{two}	Memory (00011 _{two})
100	N		
101	N		
110	Y	10 _{two}	Memory (10110 _{two})
111	N		

e. After handling a miss of address (00011_{two})

Index	V	Tag	Data
000	Y	10 _{two}	Memory (10000 _{two})
001	N		
010	Y	10 _{two}	Memory (10010 _{two})
011	Y	00 _{two}	Memory (00011 _{two})
100	N		
101	N		
110	Y	10 _{two}	Memory (10110 _{two})
111	N		

f. After handling a miss of address (10010_{two})

Referencing a Cache Block



Cache Size

- The total number of bits needed for a cache is a function of the cache size and the address size.
- Let address = 32 bits
Cache size = 2^n blocks with 2^m words.
Tag size = $32 - (n+m+2)$

Bits in a Cache

- How many total bits are required for a direct-mapped cache with 16 KB of data and 4 word blocks, assuming a 32 bit address?

Data size = 16 KB = 4K words = 2^{12} words.

Block size = 4 words (2^2).

Cache Entries = 2^{10} blocks

Block size = $4 \times 32 = 128$ bits.


Tag = $32 - 10 - 2 - 2 = 18$ bits.

Valid bit = 1 bit

Total Cache size = $2^{10} \times (128 + 18 + 1) = 147\text{Kbits} = 18.4 \text{ KB}$

Effect of Larger Blocks

- Increasing the block size usually decreases the miss rate.
- But the miss rate increases if the block size becomes too large.
 1. The number of blocks in the cache will be small.
 2. A block will be bumped out of the cache before many of its words are accessed.
- Increasing the block size will increase the cost of miss.



Improving Miss Penalty due to Larger Block

- **Early Restart:**

Restart execution as soon as the requested word from a block is available.

- **Requested Word First / Critical Word First:**

Requested word is delivered first and then the rest of the block is delivered.