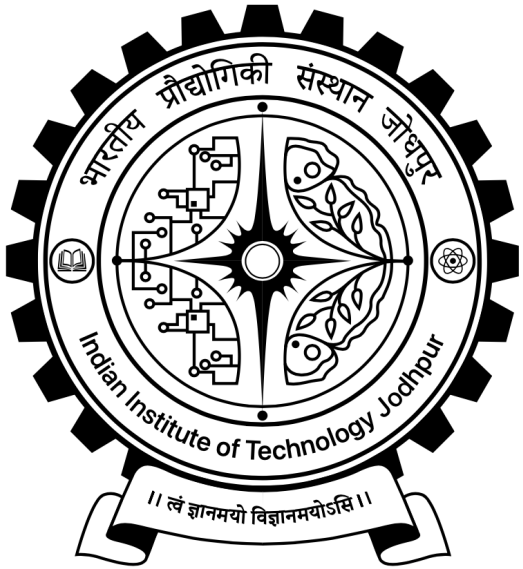


Computer Architecture
CSL3020

Deepak Mishra

http://home.iitj.ac.in/~dmishra/
Department of Computer Science and Engineering
Indian Institute of Technology Jodhpur

Overview of Memory System

Story so far –

Memory is a one large chunk of bytes.

Overview of Memory System

Story so far –

Memory is a one large chunk of bytes.

Memory access (read/write) can be completed within one clock cycle.

Overview of Memory System

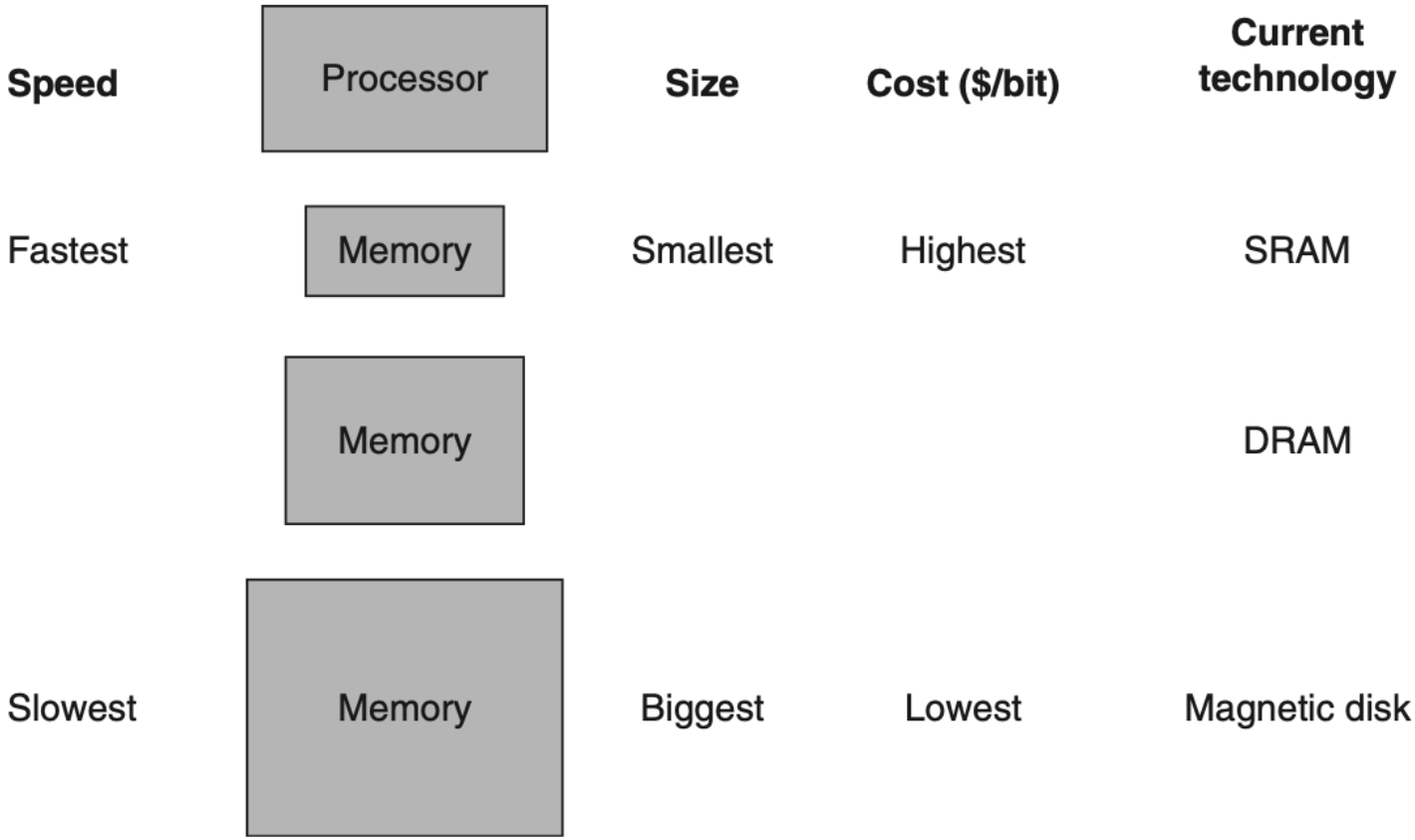
Story so far –

Memory is a one large chunk of bytes.

Memory access (read/write) can be completed within one clock cycle.

All our programs take less than 4 GB of space.

Overview of Memory System



Area, power, latency trade-off

Overview of Memory System

A memory hierarchy can consist of multiple levels, but data is copied between only two adjacent levels at a time.

Overview of Memory System

A memory hierarchy can consist of multiple levels, but data is copied between only two adjacent levels at a time.

Temporal locality: The principle stating that if a data location is referenced then it will tend to be referenced again soon.

Overview of Memory System

A memory hierarchy can consist of multiple levels, but data is copied between only two adjacent levels at a time.

Temporal locality: The principle stating that if a data location is referenced then it will tend to be referenced again soon.

Spatial locality: The principle stating that if a data location is referenced, data locations with nearby addresses will tend to be referenced soon.

Overview of Memory System

How to take the advantage of temporal locality?

Overview of Memory System

How to take the advantage of temporal locality?

Use memory hierarchy

Overview of Memory System

How to take the advantage of temporal locality?

Use memory hierarchy

When the data requested by the processor appears in upper level (cache), it is called a hit otherwise a miss.

Overview of Memory System

How to take the advantage of temporal locality?

Use memory hierarchy

When the data requested by the processor appears in upper level (cache), it is called a hit otherwise a miss.

- Hit rate: The fraction of memory accesses found in the upper level.
- Miss rate = 1 - hit rate
- Miss penalty: It is the time taken to transfer data from lower level to upper level, plus the hit time.

Overview of Memory System

How to take the advantage of spatial locality?

Overview of Memory System

How to take the advantage of spatial locality?

Group memory addresses into blocks and transfer blocks between the levels in memory hierarchy instead of transferring single byte.

- Block: The minimum unit of information that can be either present or not present in the two-level hierarchy is called a block or a line.

Basics of Cache

Using main memory (RAM) for the IM and DM units of the processor we design is inefficient.

Basics of Cache

Using main memory (RAM) for the IM and DM units of the processor we design is inefficient.

Cache represents the level of the memory hierarchy between the processor and main memory.

Basics of Cache

Using main memory (RAM) for the IM and DM units of the processor we design is inefficient.

Cache represents the level of the memory hierarchy between the processor and main memory.

How do we know if a data item is in the cache? If it is, how do we find it?

Basics of Cache

Using main memory (RAM) for the IM and DM units of the processor we design is inefficient.

Cache represents the level of the memory hierarchy between the processor and main memory.

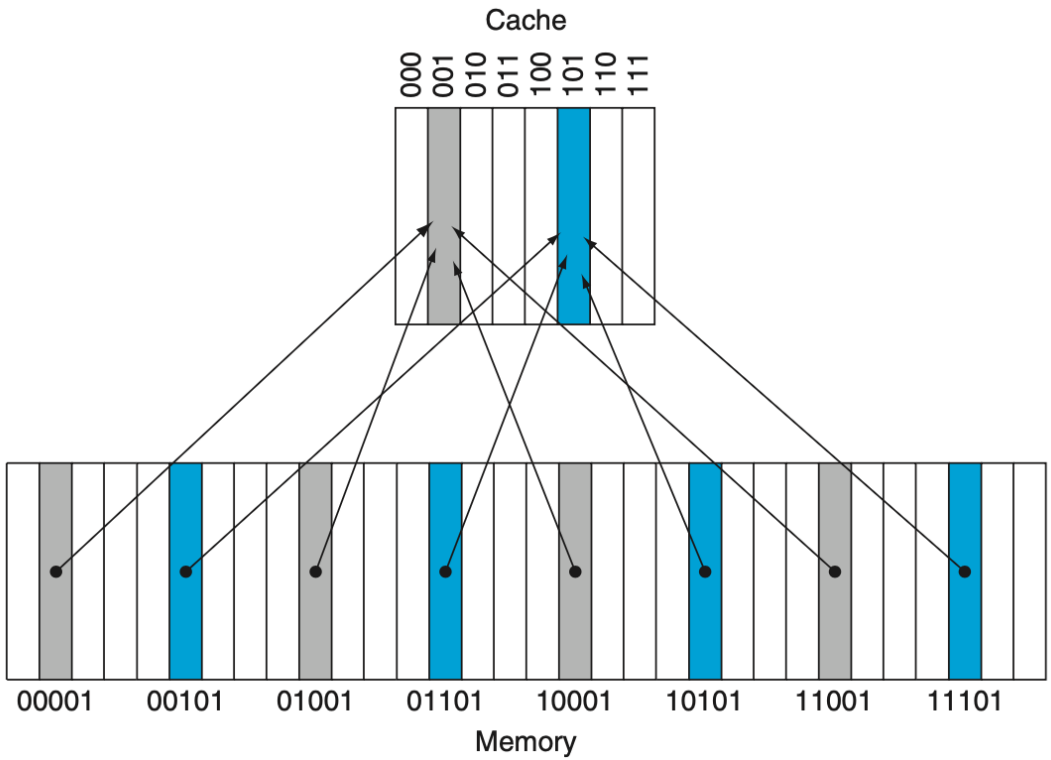
How do we know if a data item is in the cache? If it is, how do we find it?

We need a mapping scheme to answer these questions.

Basics of Cache

Direct mapping: Each (main) memory location is mapped to exactly one location in the cache.

Cache location = (Block address) modulo (Number of blocks in the cache)



Basics of Cache

As multiple addresses are mapped to a single cache location, the referenced address is divided into two fields:

- *Tag:* contains the address information required to identify associated block
- *Cache index:* used to select the block

The index of a cache block, together with the tag, uniquely specifies the corresponding memory address.

Basics of Cache

As multiple addresses are mapped to a single cache location, the referenced address is divided into two fields:

- *Tag:* contains the address information required to identify associated block
- *Cache index:* used to select the block

The index of a cache block, together with the tag, uniquely specifies the corresponding memory address.

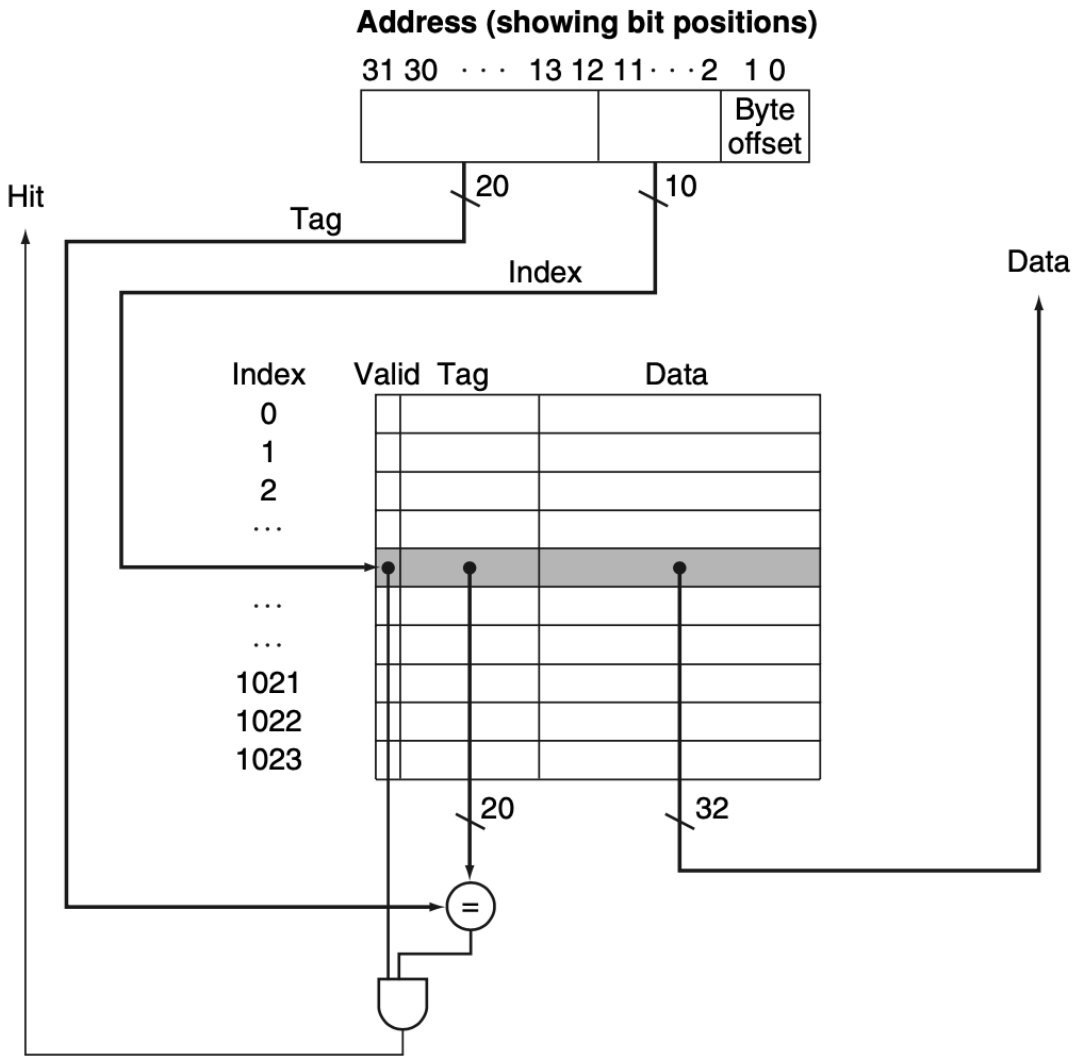
Valid bit: A field in the tables of a memory hierarchy that indicates that the associated block in the hierarchy contains valid data.

Example:

Binary address of reference	Hit or miss in cache	Assigned cache block (where found or placed)
10110_{two}	miss (5.6b)	$(10110_{two} \bmod 8) = 110_{two}$
11010_{two}	miss (5.6c)	$(11010_{two} \bmod 8) = 010_{two}$
10110_{two}	hit	$(10110_{two} \bmod 8) = 110_{two}$
11010_{two}	hit	$(11010_{two} \bmod 8) = 010_{two}$
10000_{two}	miss (5.6d)	$(10000_{two} \bmod 8) = 000_{two}$
00011_{two}	miss (5.6e)	$(00011_{two} \bmod 8) = 011_{two}$
10000_{two}	hit	$(10000_{two} \bmod 8) = 000_{two}$
10010_{two}	miss (5.6f)	$(10010_{two} \bmod 8) = 010_{two}$
10000_{two}	hit	$(10000_{two} \bmod 8) = 000_{two}$

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

Cache Misses



Assuming that the block size is one word and address is 32 bit long

Cache Misses

Example: 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?

Cache Misses

Example: 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?

Ans: 147 Kbits

Cache Misses

Steps to be taken on an instruction cache miss:

Cache Misses

Steps to be taken on an instruction cache miss:

1. Send the original PC value (current PC – 4) to the memory.

Cache Misses

Steps to be taken on an instruction cache miss:

1. Send the original PC value (current PC – 4) to the memory.
2. Instruct main memory to perform a read and wait (stall) for the memory to complete its access.

Cache Misses

Steps to be taken on an instruction cache miss:

1. Send the original PC value (current PC – 4) to the memory.
2. Instruct main memory to perform a read and wait (stall) for the memory to complete its access.
3. Write the cache entry

Cache Misses

Steps to be taken on an instruction cache miss:

1. Send the original PC value (current PC – 4) to the memory.
2. Instruct main memory to perform a read and wait (stall) for the memory to complete its access.
3. Write the cache entry
   - putting the data from memory in the data portion of the entry,
   - writing the upper bits of the address (from the ALU) into the tag field,
   - turn the valid bit on.

Cache Misses

Steps to be taken on an instruction cache miss:

1. Send the original PC value (current PC – 4) to the memory.
2. Instruct main memory to perform a read and wait (stall) for the memory to complete its access.
3. Write the cache entry
   - putting the data from memory in the data portion of the entry,
   - writing the upper bits of the address (from the ALU) into the tag field,
   - turn the valid bit on.
4. Restart the instruction execution at the first step.

Cache Misses

The control of the cache on a data access is essentially identical: on a miss, we simply stall the processor until the memory responds with the data.

Cache Misses

Handling the write miss:

- *Write-through:* A scheme in which writes always update both the cache and the next lower level of the memory hierarchy.

Cache Misses

Handling the write miss:

- *Write-through:* A scheme in which writes always update both the cache and the next lower level of the memory hierarchy.
- *Write buffer:* A queue that holds data while the data is waiting to be written to memory.

Cache Misses

Handling the write miss:

- *Write-through:* A scheme in which writes always update both the cache and the next lower level of the memory hierarchy.
- *Write buffer:* A queue that holds data while the data is waiting to be written to memory.
- *Write-back:* Updating values only to the block in the cache, then writing the modified block to the lower level of the hierarchy when the block is replaced.

Cache Performance Analysis

How to evaluate cache performance

Cache Performance Analysis

How to evaluate cache performance

$$\text{CPU time} = (\text{CPU execution clock cycles} + \text{Memory-stall clock cycles}) \times \text{Clock cycle time}$$

Cache Performance Analysis

How to evaluate cache performance

$$\text{CPU time} = (\text{CPU execution clock cycles} + \text{Memory-stall clock cycles}) \times \text{Clock cycle time}$$

$$\text{Memory-stall clock cycles} = \text{Memory accesses per program} \times \text{Miss rate} \times \text{Miss penalty}$$

Cache Performance Analysis

How to evaluate cache performance

$$\text{CPU time} = (\text{CPU execution clock cycles} + \text{Memory-stall clock cycles}) \times \text{Clock cycle time}$$

$$\text{Memory-stall clock cycles} = \text{Memory accesses per program} \times \text{Miss rate} \times \text{Miss penalty}$$

$$\text{Memory-stall clock cycles} = \frac{\text{Instructions}}{\text{Program}} \times \frac{\text{Misses}}{\text{Instructions}} \times \text{Miss penalty}$$

Cache Performance Analysis

Example: Assume the miss rate of an instruction cache is 2% and the miss rate of the data cache is 4%. If a processor has a CPI of 2 without any memory stalls and the miss penalty is 100 cycles for all misses, determine how much faster a processor would run with a perfect cache that never missed. Assume the frequency of all loads and stores is 36%.

Cache Performance Analysis

Example: Assume the miss rate of an instruction cache is 2% and the miss rate of the data cache is 4%. If a processor has a CPI of 2 without any memory stalls and the miss penalty is 100 cycles for all misses, determine how much faster a processor would run with a perfect cache that never missed. Assume the frequency of all loads and stores is 36%.

Ans: 2.72 times

Reducing Cache Misses

Flexible Mapping of Blocks

- *Fully Associative:* A cache structure in which a block can be placed in any location in the cache.

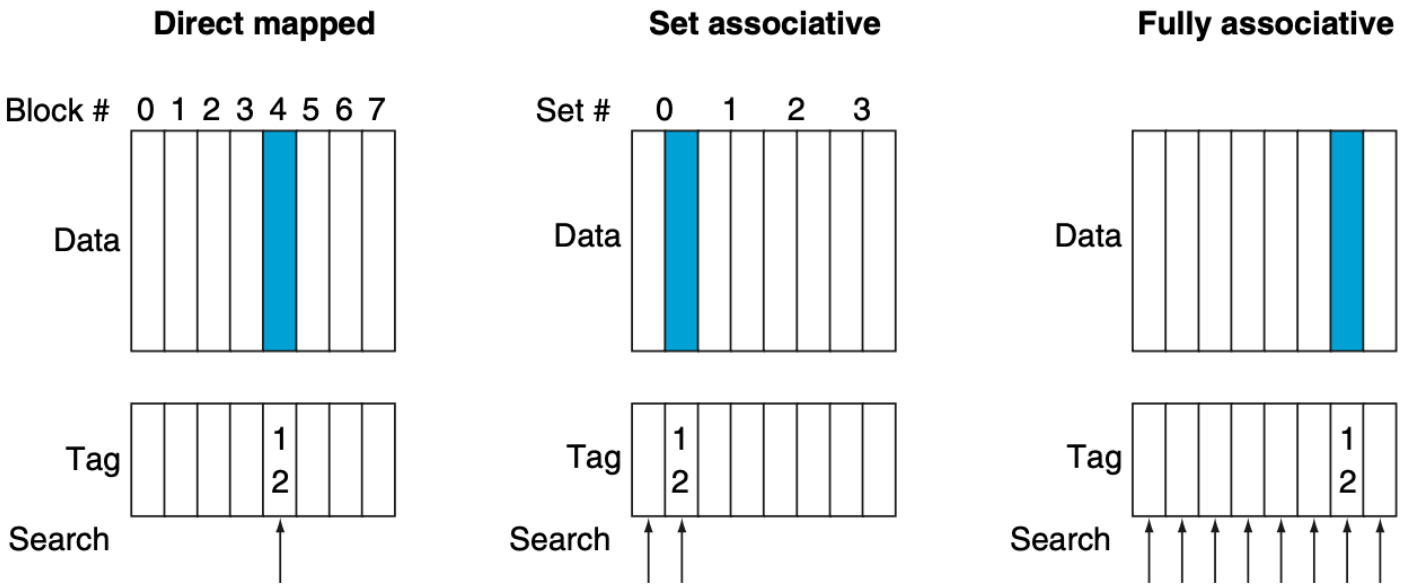
Reducing Cache Misses

Flexible Mapping of Blocks

- *Fully Associative:* A cache structure in which a block can be placed in any location in the cache.
- *Set-associative:* A cache that has a fixed number of locations (at least two) where each block can be placed. A set-associative cache with n locations for a block is called an n-way set-associative cache.

Flexible Mapping of Blocks

- *Fully Associative:* A cache structure in which a block can be placed in any location in the cache.
- *Set-associative:* A cache that has a fixed number of locations (at least two) where each block can be placed. A set-associative cache with n locations for a block is called an n-way set-associative cache.



Reducing Cache Misses

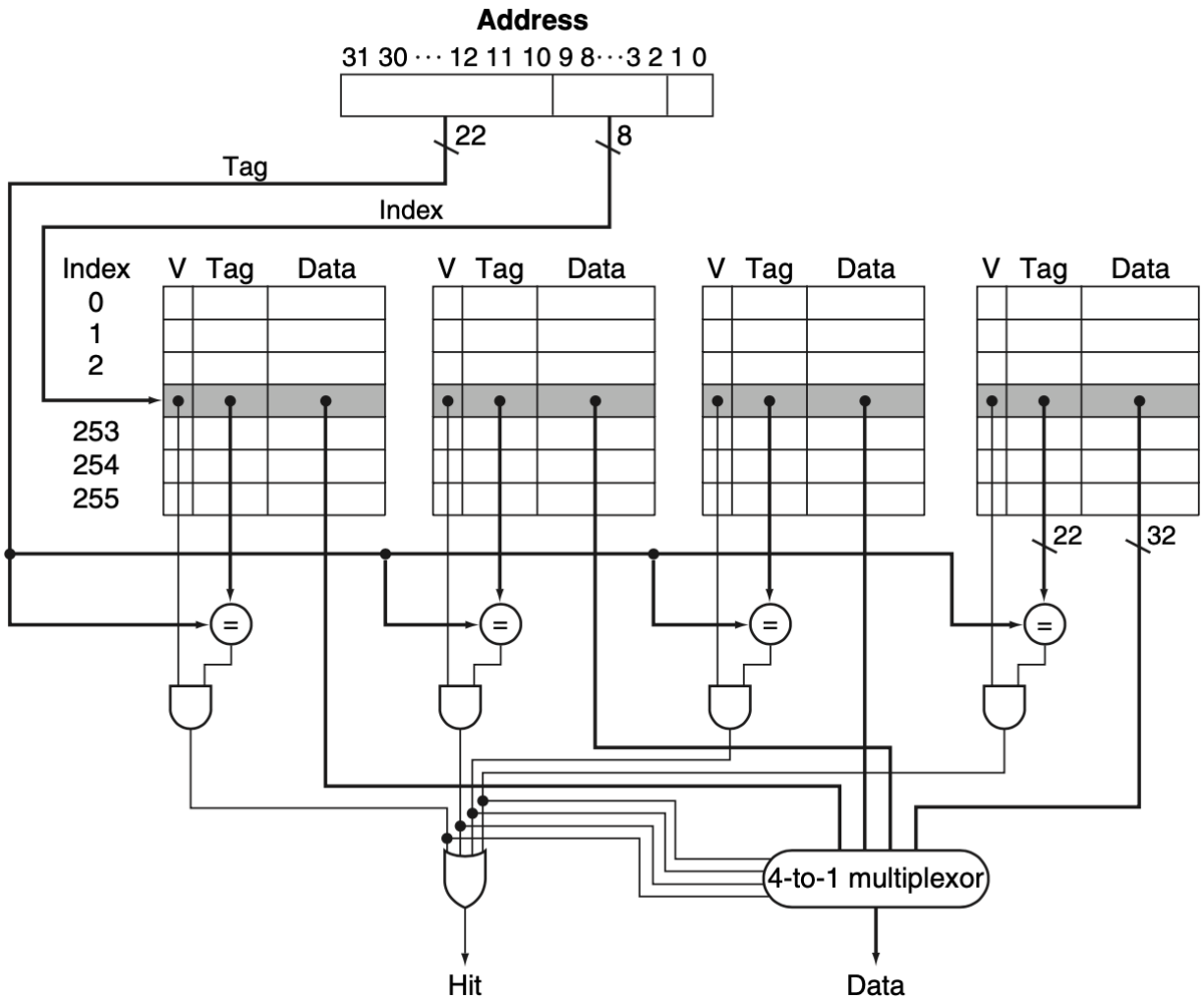
Example: Assume there is a small cache consisting of eight one-word blocks. Given the following sequence of block addresses: 0, 8, 0, 5, 12, 9, 1, 5, and 8, find the number of misses for 2-way, 4-way, and 8-way set-associative mapping with LRU replacement scheme.

Reducing Cache Misses

Example: Assume there is a small cache consisting of eight one-word blocks. Given the following sequence of block addresses: 0, 8, 0, 5, 12, 9, 1, 5, and 8, find the number of misses for 2-way, 4-way, and 8-way set-associative mapping with LRU replacement scheme.

Ans: 8, 6, 6

Reducing Cache Misses



Implementation of a 4-way set-associative cache requires four comparators and a 4-to-1 multiplexor.

Reducing Cache Misses - Multilevel Cavhe