Outline of Lecture

- Finding Memory References in Cache
- Cache Memory Performance
- Designing Main Memory to Support Cache
- Improving Cache Memory Performance

Address Calculation

Given a main memory reference address of a block, where do we find that block a direct-mapped cache?

Answer:

If block size = 1 word, then

address in cache = (block address of main memory) modulo (number of blocks in cache)

For example, if the number of blocks in cache is equal to 8, then the main memory address 10110_2 (22_{10}) will be found in 110_2 (6_{10}).

In MIPS, the byte address is 32 bit, the word address is 30 bit, and cache size is 1024 blocks (1 ord = 1 block), thus the block address will be

address in cache (10 bits) = (30 bit word address in main memory) modulo (1024)

Address Calculation

If block size = 4 word, then

address in cache = (block address of main memory) modulo (number of blocks in cache)

The block address is simply the word address divided by the number of words in the block (or equivalently, the byte address divided by the number of bytes in a block).

Example

Consider a cache with 64 blocks and a block size of 16 bytes. What cache block number does byte address 1200 map to?

Answer:

With 16 bytes per block, byte address 1200 is block address

$$\left\lfloor \frac{1200}{16} \right\rfloor = 75$$

which maps to cache block number (75 mod 64) = 11.

Cache Performance

- **Hit:** data or instructions appears in some block in the cache (example: Block X)
  - ➔ **Hit Rate:** the fraction of memory accesses found in the cache.
  - ➔ **Hit Time:** Time to access the cache which consists of cache access time + Time to determine hit/miss.
- **Miss:** data needs to be retrieved from a block in the main memory (example: Block Y)
  - ➔ **Miss Rate:** 1 - Hit Rate.
  - ➔ **Miss Penalty:** Time to replace a block in the cache + time to deliver the block to the processor.
- Hit Time << Miss Penalty

Mean Memory Access Time

It is given as follows:

Mean memory access time = Hit time + (1 - hit rate) x miss penalty

Example

A cache system has a 95% hit ratio, an access time of 10 nsec on a cache hit and an access time of 80 nsec on a cache miss.

What is the mean memory access time?

Answer

Mean memory access time = Hit time + (1 - hit rate) x miss penalty

= 10 + (1 - 0.95) x 80

= 14 nsec.

Main Memory

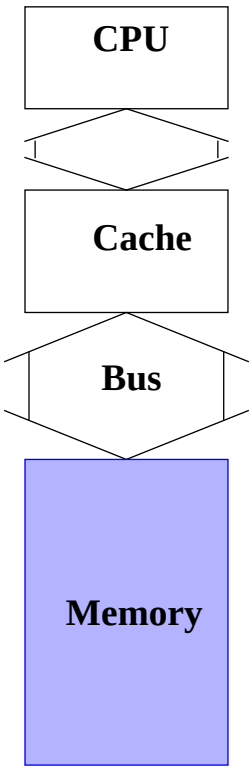
- **Main memory is *DRAM*: Dynamic Random Access Memory**
  - ➔ **High density, low power, cheap, slow**
  - ➔ ***Dynamic:*** **It needs to be refreshed periodically (8 msec) - it is typically built using capacitors.**
- **Cache uses *SRAM*: Static Random Access Memory**
  - ➔ **Low density, high power, expensive, fast**
  - ➔ **No refresh (6 transistors/bit vs. 1 transistor-capacitor/bit)**
- **Size: DRAM/SRAM: 4 - 8.**

Memory technology	Typical access time	$ per MByte in 1993
SRAM	8 - 35 nsec	$100-$400
DRAM	90-120 nsec	$25-$50
Magnetic disk	10,000,000-20,000,000 nsec	$1-$2

Improving Main Memory Performance

- The performance improvement of DRAMs is about 7% per year. It is *much less* than the yearly improvement of CPUs (60%).
- Innovative organizations of main memory are needed.
- Improving the memory organization should be done to decrease the access time and/or to *increase the bandwidth* - it is much easier and more efficient to improve bandwidth.
- Assume the performance of a main memory is as follows:
  - ➔ 4 clock cycles to send the address
  - ➔ 24 clock cycles for the access time per word
  - ➔ 4 clock cycles to send a word of data

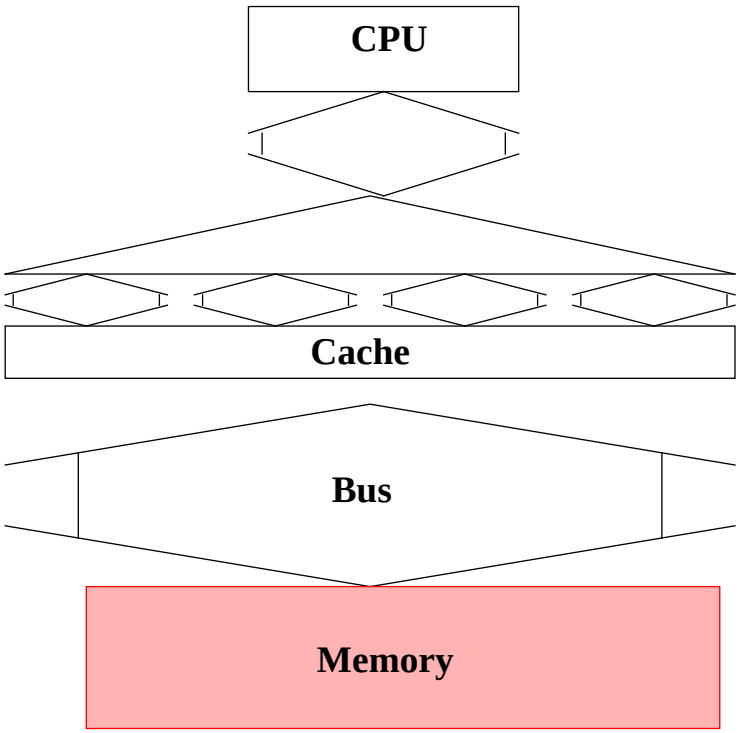
Simple Memory System



One-word-wide
memory organizationi

- Given a cache block of 4 words, it would take 4 x (4 + 24 + 4) = 128 clock cycles for a cache miss.
- The memory bandwidth is 16/128 = 0.125 bytes per clock cycle — which is quite low.

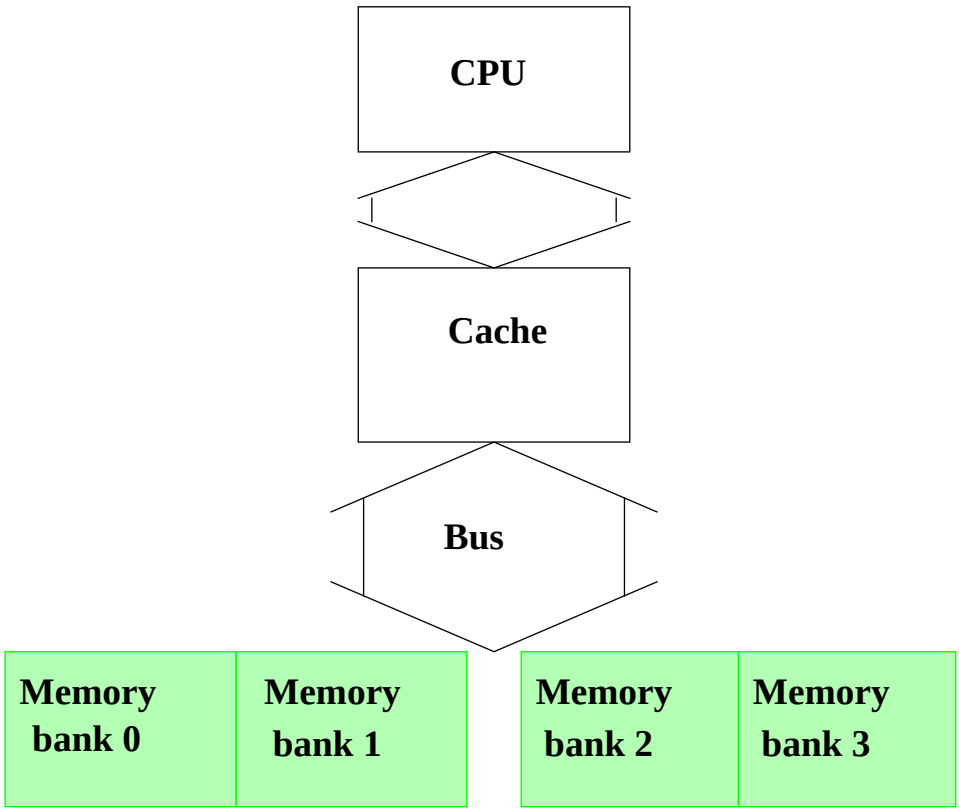
Higher Bandwidth: Wider Main Memory



Wide memory organization

- With a main memory width of two words, the miss penalty in our example would drop from 128 clock cycles to 2 x (4 + 24 + 4) = 64 clock cycles.
- At 4 words wide, the miss penalty is just 4 + 24 + 4 = 32 clock cycles.
- Wider memory bus needs more hardware cost.

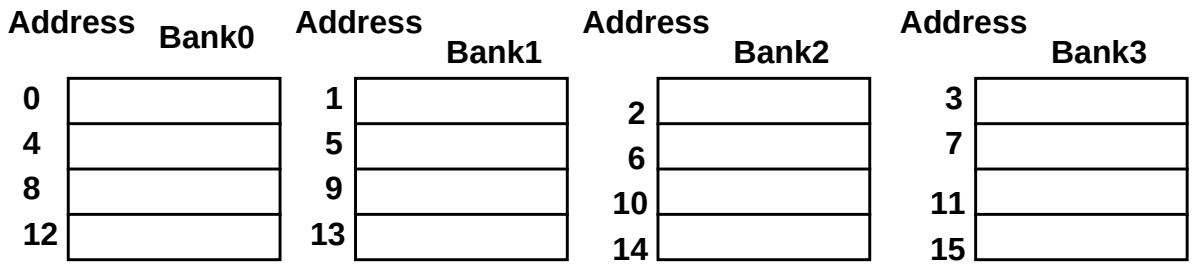
Higher Bandwidth: Simple Interleaved memory



Interleaved memory organization.

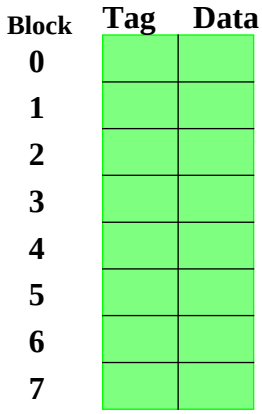
- **Memory chips can be organized in banks to read or write multiple words at a time rather than a single word.**
- **It takes 4 + 24 + 4 x 4 = 44 clock cycles for a miss penalty.**

- If we have 4 memory banks, addressing the 4 memory banks can be done as follows:

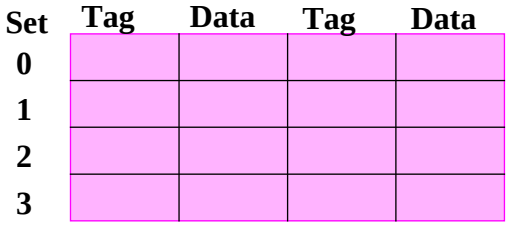


Decreasing Miss Ratio with Associativity

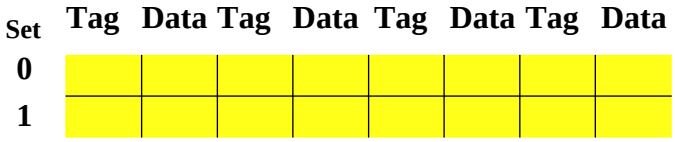
direct (mapped)



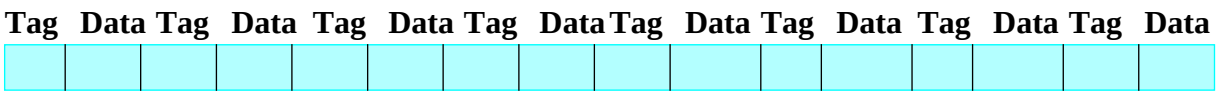
Two-way set associative



Four-way set associative



Eight-way set associative (fully associate)



In a set-associative cache, the set containing a memory is given by

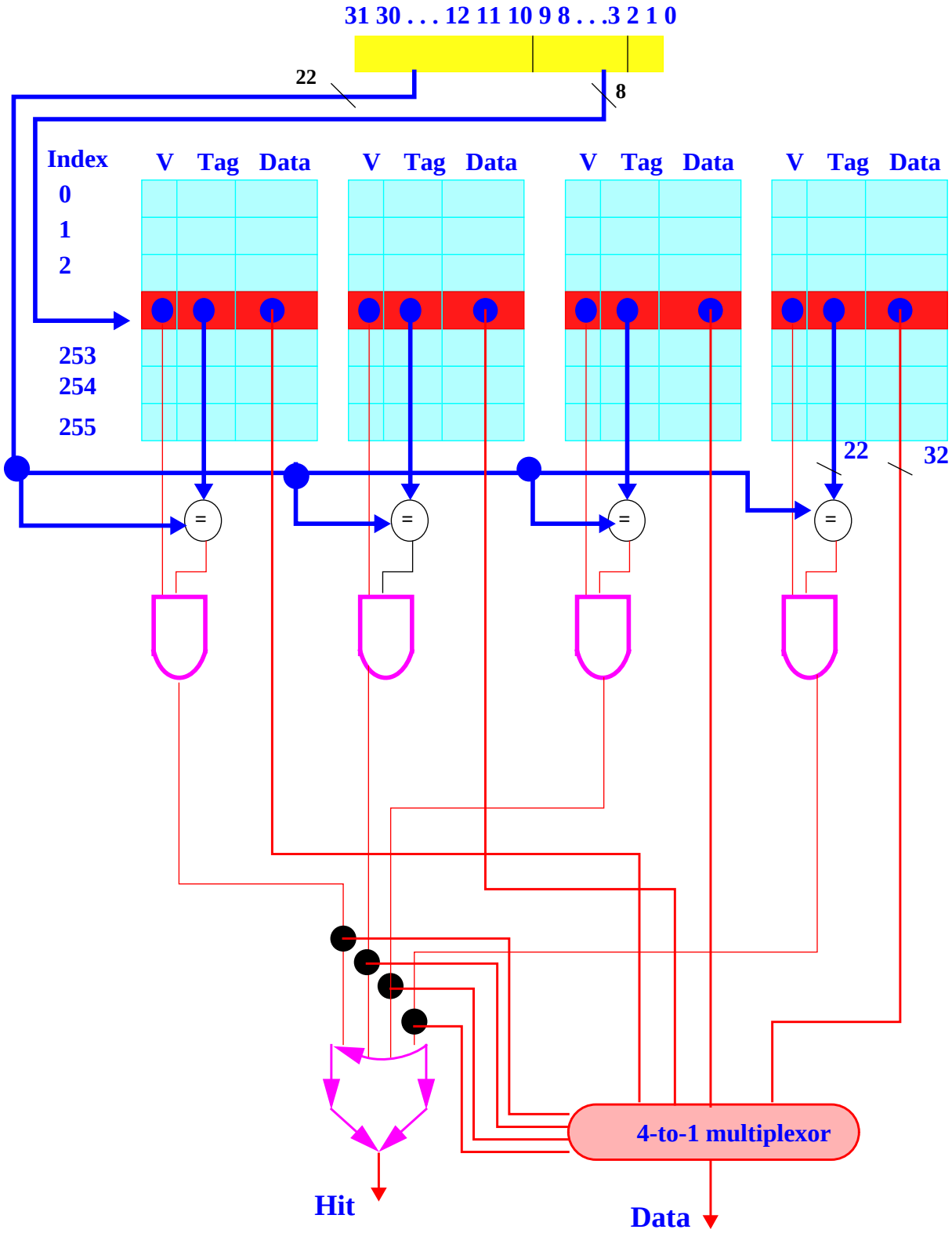
set in cache = (block number)

modulo (number of sets in cache)

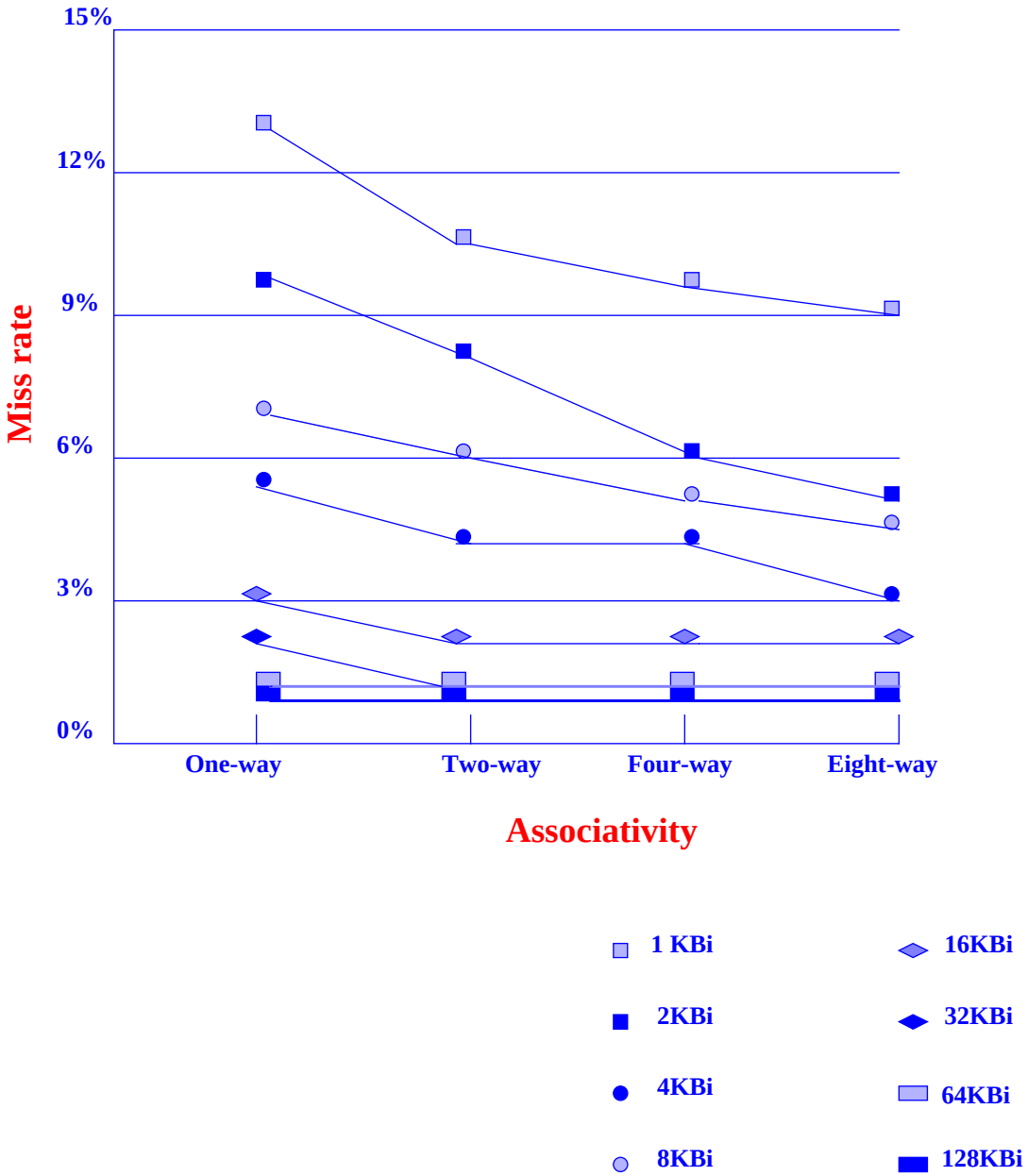
Set-Associative Cache

- **Since the block may be placed in any of the set, all elements of the set must be searched.**
- **In a fully associative cache, the block can go anywhere, so all blocks must be searched.**

An Implementation



Performance



Decreasing Miss Penalty with Multilevel Caches

Add a second level cache

- **often primary cache is on the same chip as the processor**
- **use SRAMs to add another cache above primary memory (DRAM)**
- **miss penalty goes down if data is in 2nd level cache**

Further Reading

Chapter 7. **David A. Patterson and John L. Hennessy. *Computer Organization & Design: The Hardware / Software Interface*. Morgan Kaufman (page 558-576).**