
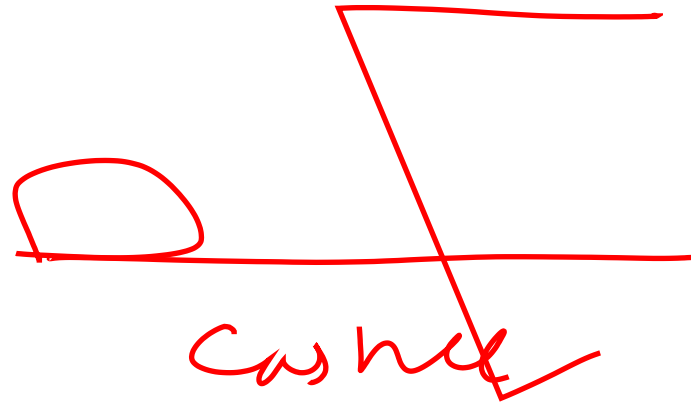


Exploiting Memory Hierarchy

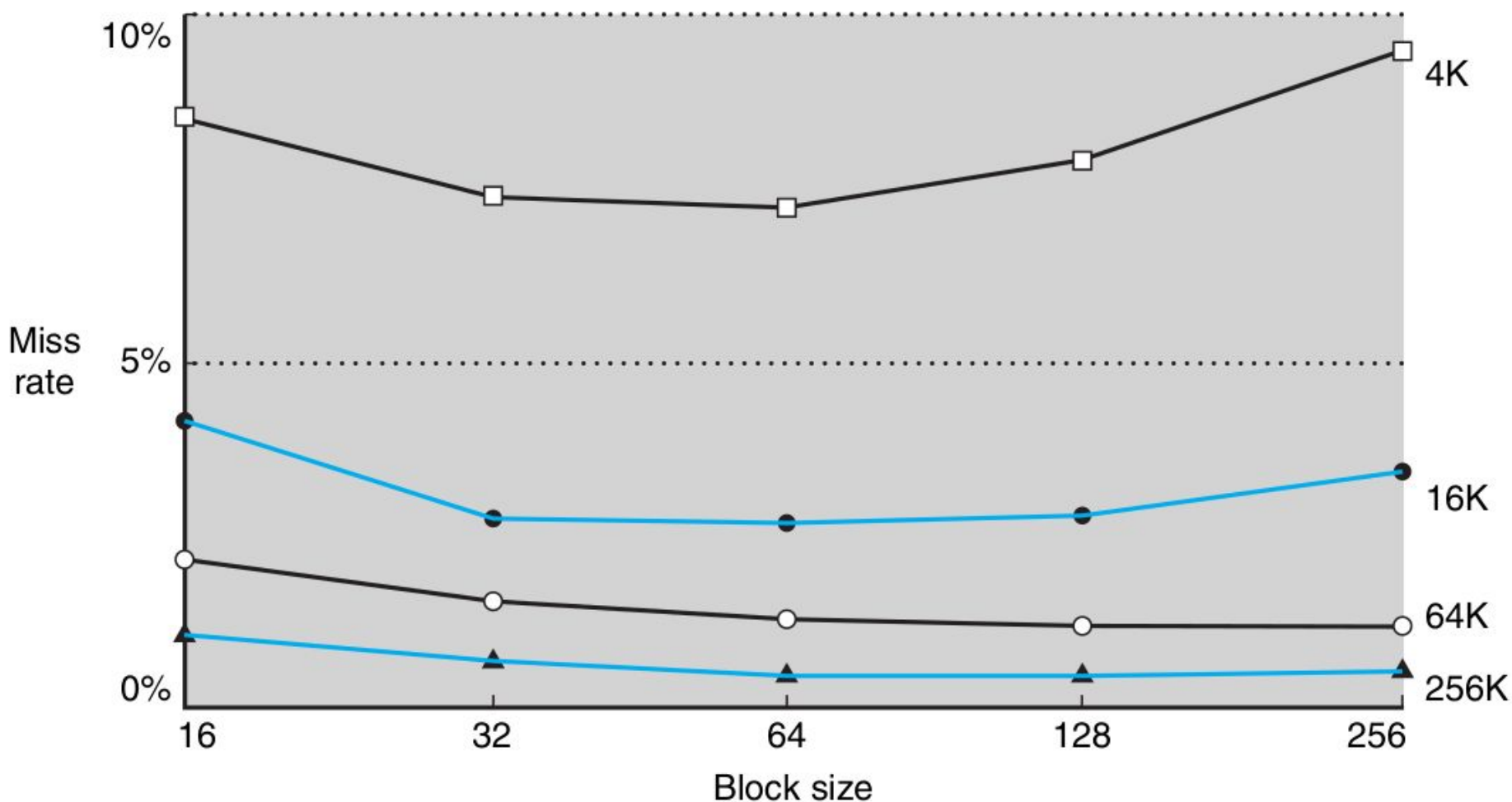
Chapter Five of Book of David A.
Patterson

Effect of Larger Blocks

- ✓ It uses spatial locality to lower miss rates. 
- ✓ 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.



Miss Rate vs Block Size



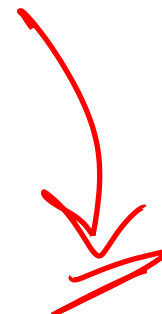
Improving Miss Penalty due to Larger Block

- ✓ Early Restart:

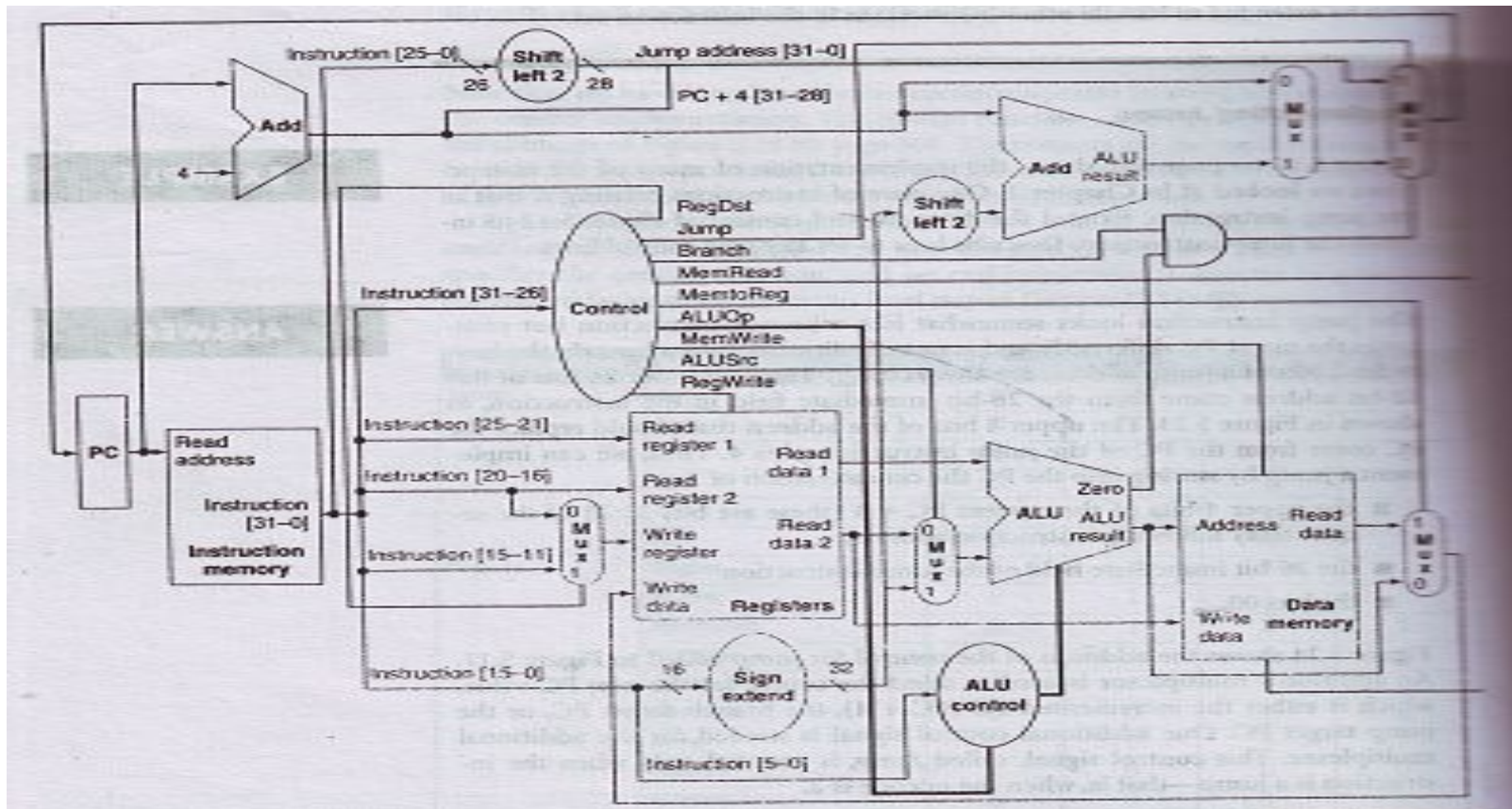
Restart execution as soon as the requested word from a block is available.

- ✓ Requested Word First / Critical Word First:

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



Datapath for Multicycle Processor



Handling Cache Misses

Steps Taken on an Instruction Cache Miss:

1. Send the original PC value (PC-4) to the memory.
2. Instruct main memory to perform a read and wait for the memory to complete its access.
3. Write the cache entry.
4. Restart the execution at the first step.

Handling Writes

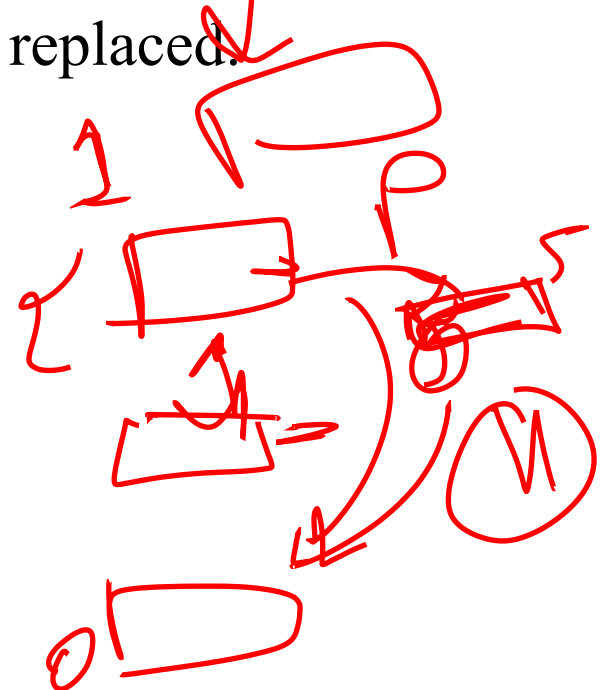
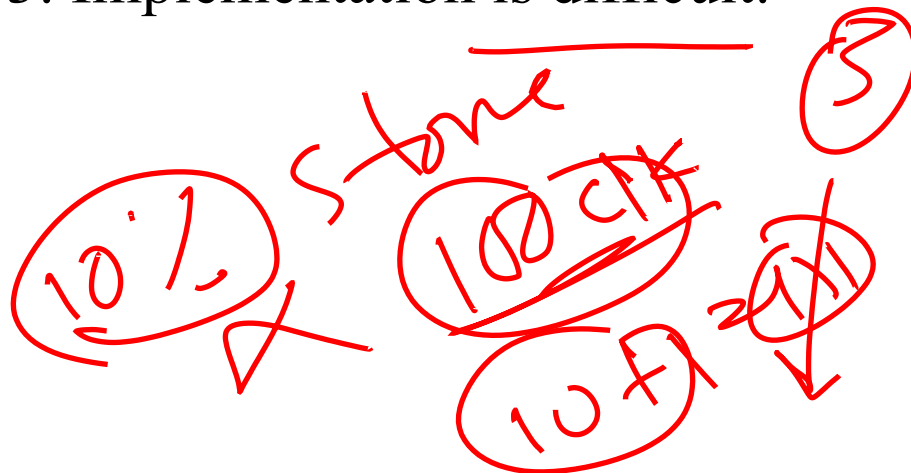
✓ Write Through:

1. Writes always update both the cache and the memory, ensuring that data is always consistent between two.
2. Memory write operation will take longer time and slow down the processor.
3. Let CPI is 1.0 without cache miss and about 10% of the instruction is store and every write require 100 clock cycle, then $CPI = 1.0 + 100 * 10\% = 11$.
4. Solution to this problem is write buffer.
5. The processor must stall for the write buffer to become empty.
6. If the rate of generating write by the processor is larger than the rate at which the memory can accept, then it will create problem.

Handling Writes

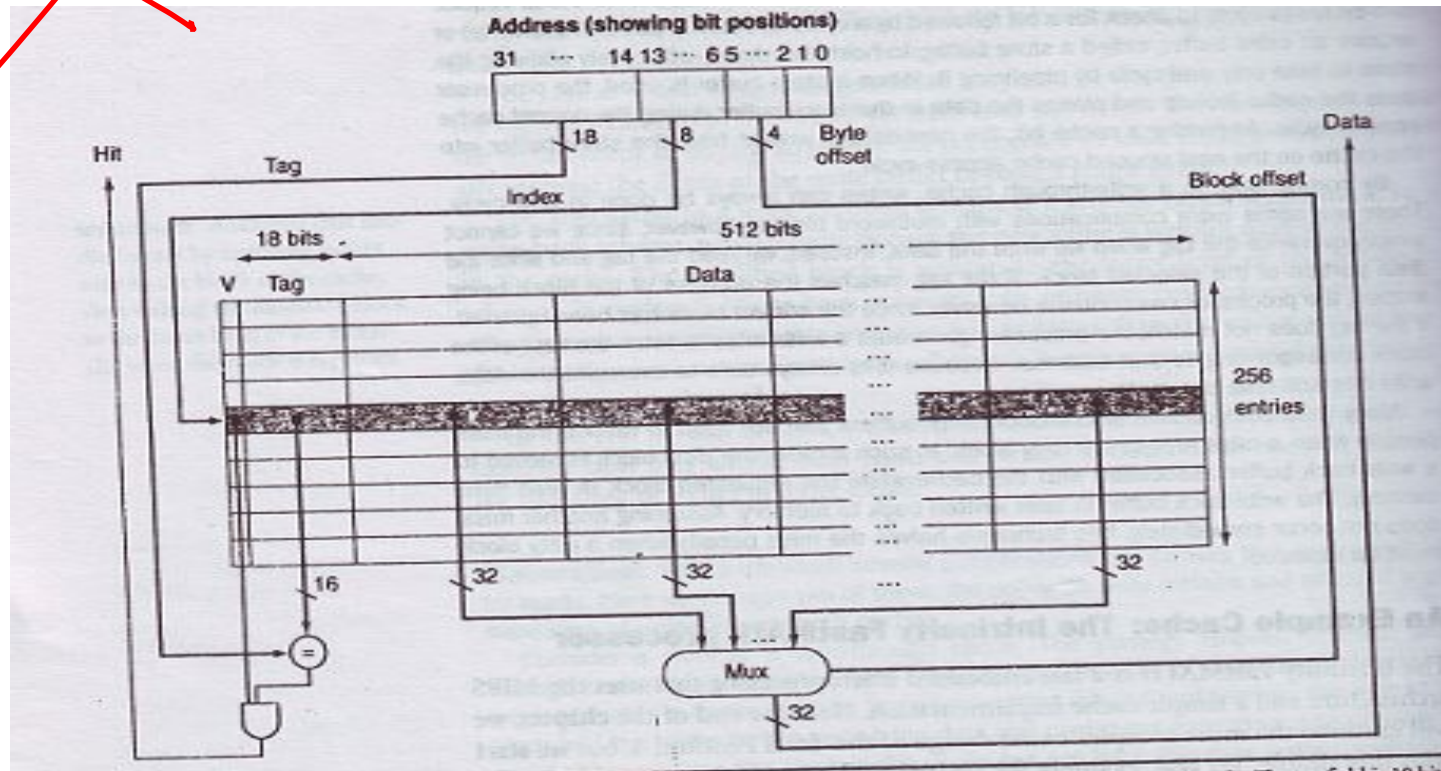
✓ Write Back:

1. It handles write by 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.
2. It improves performance.
3. Implementation is difficult.



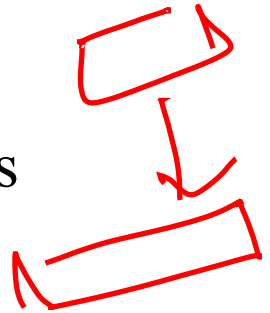
Cache Representation of Intrinsity

FastMATH Processor



Designing Memory System to Support Cache

- ✓ We can reduce the miss penalty by Increasing the memory bandwidth.
- ✓ The speed of bus connecting memory and cache will also affect the miss penalty.
- ✓ 1 memory bus clock cycle to send the address.
15 memory bus clock cycles for each DRAM access initiated.
1 memory bus clock cycle to send a word of data.



One-Word-Wide Memory Organization

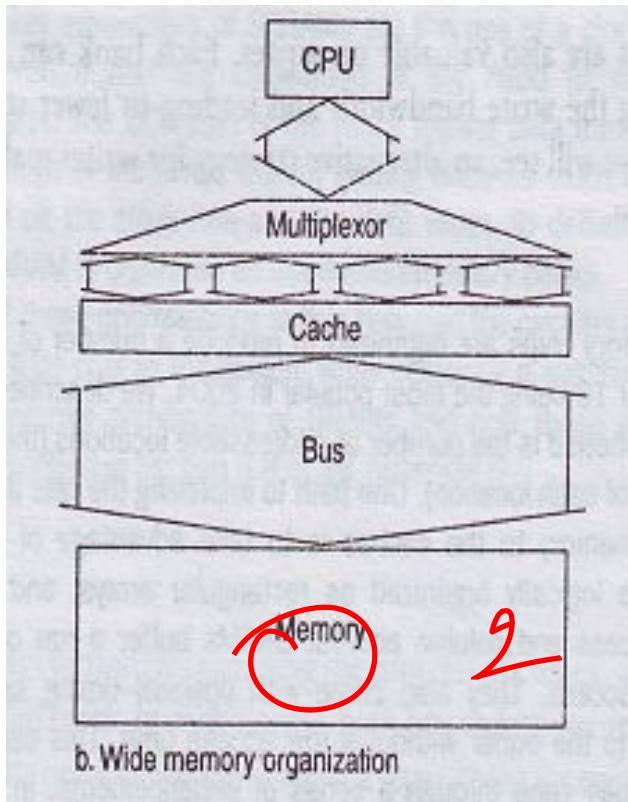


- ✓ Cache block = 4 words.
- Memory width = 1 word.
- Miss penalty = $1 + 4 * 15 + 4 * 1 = 65$ memory bus clock cycles.
- The number of bytes transferred per bus clock cycle for a single miss = $4 * 4 / 65 = 0.25$

Handwritten calculations and corrections in red ink:

- $16 / 65$ (crossed out)
- $4 * 4 / 65$ (circled)
- $16 / 65$ (crossed out)
- $4 * 4 / 65$ (circled)

Wide Memory Organization



Cache block = 4 words.

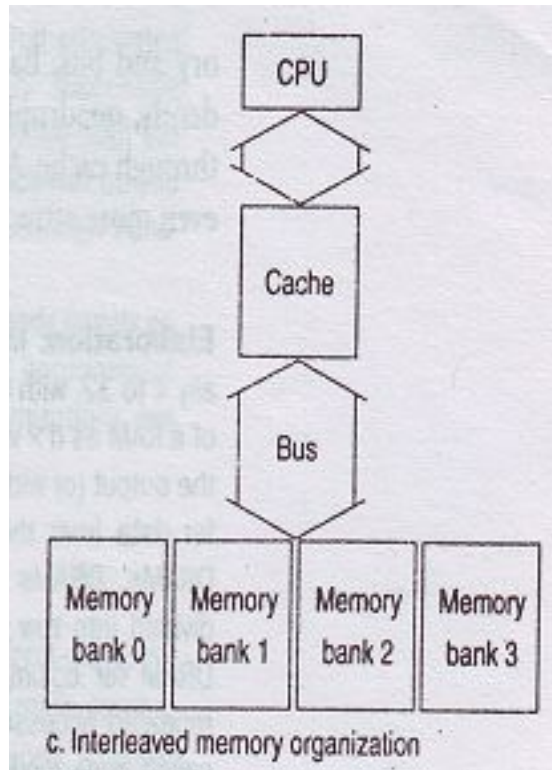
Memory width = 2 words.

Miss penalty = $1 + 2 \times 15 + 2 \times 1 = 33$ memory bus clock cycles.

The number of bytes transferred per bus clock cycle for a single miss =

$$4 \times 4 / 33 = 0.48$$

Interleaved Memory Organization



- ✓ Cache block = 4 words.
Memory width = 4 words.
Miss penalty = $1 + 1 * 15 + 4 * 1 = 20$ memory bus clock cycles.
The number of bytes transferred per bus clock cycle for a single miss = $4 * 4 / 20 = 0.80$
- ✓ It also improves write miss.