METRO STATE UNIVERSITY

ICS 232 Computer Organization & Architecture
Homework 10 - Chapter 6 - 10 points
Due Date: 7/19/2023

Name: Key

Note: Please post your homework to ICS232 D2L on or before the due date.

Chapter 6 – Memory

Essential Terms and Concepts

4. Explain the concept of memory hierarchy. Why did the authors choose to represent it as a pyramid?

The CPU is faster than memory and therefore needs to access memory as efficiently as possible. The fastest, smallest and most expensive memory is closest to the CPU. Slower, larger and less expensive memory to further away from the CPU.

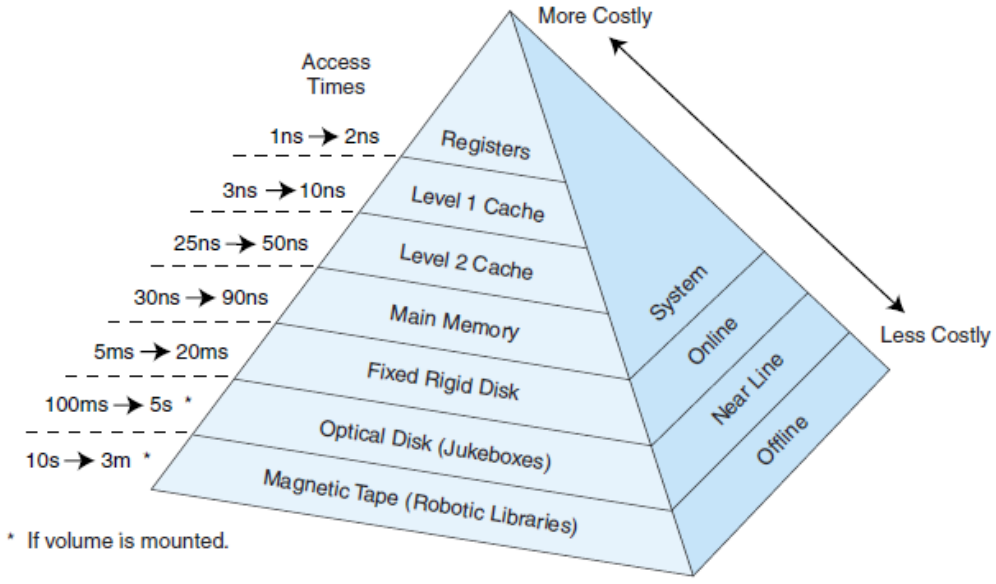

FIGURE 6.1 The Memory Hierarchy

6. What are the three forms of locality?

METRO STATE UNIVERSITY

ICS 232 Computer Organization & Architecture
Homework 10 - Chapter 6 - 10 points
Due Date: 7/19/2023

Temporal Locality – recently accessed
Spatial Locality – near other recently accessed
Sequential Locality – instructions are usually accessed sequentially.

13. Explain how set-associative cache combines the ideas of direct and fully associative cache.

A large cache is too expensive to be fully associative. Set-associative cache divides the cache into multiple groups of fully associative caches. An address is assigned to a group by the upper-bits of its value (the direct portion). Then a parallel (associative) lookup is performed on that part of the cache.

18. What, exactly, is effective access time (EAT)?

Effective access time is the average access time of a memory access. It takes into account the expected hit-ratio in the L1 and L2 caches and the actual access time for the L1, L2, and main memory.

EAT = cache-hit-rate x cache-access-time +
(1 - cache-hit-rate) x main-memory-access-time

33. What is a page fault?

A page fault occurs when a program references an address that is not currently in memory. The operating system then needs to read that page from disk and update the page table.

Exercises

2. Suppose a computer using direct mapped cache has 2^{32} bytes of byte-addressable main memory, and a cache of 1024 blocks, where each cache block contains 32 bytes.

a) How many blocks of main memory are there?

METRO STATE UNIVERSITY

ICS 232 Computer Organization & Architecture
Homework 10 - Chapter 6 - 10 points
Due Date: 7/19/2023

$2^{32}/2^5 = 2^{27}$

b) What is the format of a memory address as seen by the cache, i.e., what are the sizes of the tag, block, and offset fields?
32-bit addresses with 17 bits in the tag field, 10 in the block field, and 5 in the offset field

c) To which cache block will the memory address 0x000063FA map?
000063FA = 00000000000000000 *1100011111* 11010, which implies block 799_{10} ($31F_{16}$)

5. Suppose a computer using fully associative cache has 2^{24} bytes of byte-addressable main memory and a cache of 128 blocks, where each cache block contains 64 bytes.

a) How many blocks of main memory are there?
$2^{24}/2^6 = 2^{18}$

b) What is the format of a memory address as seen by the cache, i.e., what are the sizes of the tag and offset fields?
24-bit addresses with 18 bits in the tag field and 6 in the offset field

c) To which cache block will the memory address 0x01D872 map?
Since it's associative cache, it can map anywhere

16. Assume a direct-mapped cache that holds a total of 4096 bytes, where each block is 16 bytes. Assuming an address is 32 bits and that cache is initially empty, complete the table below. (You should use hexadecimal numbers for all answers.) Which, if any of the addresses will cause a collision (forcing the block that was just brought in to be overwritten) if they are accessed one right after the other?

Address	Tag	Cache location block	Offset within block
0x0FF0FABA	0FF0F	AB	A

METRO STATE UNIVERSITY

ICS 232 Computer Organization & Architecture
Homework 10 - Chapter 6 - 10 points
Due Date: 7/19/2023

0x00000011	00000	01	1
0x0FFFFFFE	0FFFF	FF	E
0x23456719	23456	71	9
0xCAFEBABE	CAFEB	AB collision	E

The first and last addresses both map to cache block 0xAB and therefore will cause a collision.

20. Suppose you have a byte-addressable virtual address memory system with 8 virtual pages of 64 bytes each, and 4-page frames. Assuming the following page table, answer the questions below:

Page #	Frame #	Valid Bit
0	1	1
1	3	0
2	-	0
3	0	1
4	2	1
5	-	0
6	-	0
7	-	0

Note: Page 1 should have the valid bit = 1.

a) How many bits are in a virtual address?

8 virtual pages of size 64 bytes each is $2^3 \times 2^6 = 2^9$. Therefore, each virtual address has 9 bits.

b) How many bits are in a physical address?

METRO STATE UNIVERSITY

ICS 232 Computer Organization & Architecture
Homework 10 - Chapter 6 - 10 points
Due Date: 7/19/2023

There are 4 pages frames of size 64 each, or $2^2 \times 2^6 = 2^8$. So, each physical address has 8 bits.

c) What physical address corresponds to the following virtual addresses (if the address causes a page fault, simply indicate this is the case)?

i) 0x00

i) 0x0 = 000 000000 so this address is on page 0, offset 0. Page 0 maps to frame 1. Substituting 01 for 000, we get 01 000000, or 0x40.

ii) 0x44

ii) 0x44 = 001 000100 so this address is on page 1, offset 4. Page 1 maps to frame 3. Substituting 11 for 001, we get 11 000100, or 0xC4.

iii) 0xC2

iii) 0xC2 = 011 000010 so this address is on page 3, offset 2. Page 3 maps to frame 0. Substituting 00 for 011, we get 00 000010, or 0x02.

iv) 0x80

iv) 0x80 = 010 000000 so this address is on page 2, offset 0. Page 2 is not currently in memory so this generates a page fault.

24. Does a TLB miss always indicate that a page is missing from memory? Explain.

No. While the page could be missing from memory, a TLB miss simply means that page is not cached in the TLB.

27. Consider a system that has multiple processors where each processor has its own cache, but main memory is shared among all processors.
a) Which cache write policy would you use?

METRO STATE UNIVERSITY

ICS 232 Computer Organization & Architecture
Homework 10 - Chapter 6 - 10 points
Due Date: 7/19/2023

b) The Cache Coherency Problem. With regard to the system just described, what problems are caused if a processor has a copy of memory block A in its cache and a second processor, also having a copy of A in its cache, then updates main memory block A? Can you think of a way (perhaps more than one) of preventing this situation, or lessening its effects?

a) Write through should be used to maintain consistency. If some processors have a value cached, and one processor changes that value, the other processors would not know about that value. With a write through cache (in addition to a broadcast "invalidate" message) the processors would not be using stale values.
b) As mentioned in the answer for part a, the problem is stale data. One way to solve this problem is to invalidate stale entries. A way to prevent the situation would be to require processors to specify whether the values were for writing or reading. Shared reading would be ok; however, when a processor wanted to write, it would have to either wait until the readers were done, or send an invalidate message to those readers. Exclusive access by writers would then be allowed. The MESI protocol implements this procedure.

29. Name two ways that, as a programmer, you can improve cache performance.

Programmers should focus on improving the reference locality. This can be done by using cache-conscious algorithms (for example, change a program's data access pattern to optimize locality in nested loops used in matrices by interchanging loops) or a program's data organization and layout (such as using cache-conscious data structures).

Prepare for next class by reading Chapter 7 – Input/Output Systems

Continue working on Project 2

Continue working on Your Group Project