

1. Suppose a computer using direct mapped cache has 2^{20} bytes of byte-addressable main memory and a cache of 32 blocks, where each cache block contains 16 bytes.

a) How many blocks of main memory are there?

$$2^{20}/2^4 = 2^{16} \text{ blocks}$$

b) What is the format of a memory address as seen by the cache; that is, what are the sizes of the tag, block, and offset fields?

20 bit addresses with 7 bits in the tag field, 5 in the block field, and 4 in the offset Field.

Tag	block	offset
7	5	4

c) To which cache block will the memory address 0x0DB63 map?

$$0000\ 1101\ 1011\ 0110\ 0011 \rightarrow 00001101101\ 10110\ 0011$$

Block 10110 or 22

4. Suppose a computer using fully associative cache has 2^{16} bytes of byte-addressable main memory and a cache of 64 blocks, where each cache block contains 32 bytes.

a) How many blocks of main memory are there?

$$2^{16} / 2^5 = 2^{11} \text{ blocks}$$

b) What is the format of a memory address as seen by the cache; that is, what are the sizes of the tag and offset fields?

16 bit addresses with 5 bits in the tag field, 6 in the block field, and 5 in the offset Field.

Tag	Block	offset
5	6	5

c) To which cache block will the memory address 0xF8C9 map?

$$1111\ 1000\ 1100\ 1001 \rightarrow 11111\ 000110\ 01001 = \text{block } 110 = 6$$

9. Suppose a byte-addressable computer using set associative cache has 2^{16} bytes of main memory and a cache of 32 blocks, and each cache block contains 8 bytes.

a) If this cache is 2-way set associative, what is the format of a memory address as seen by the cache; that is, what are the sizes of the tag, set, and offset fields?

16 bit addresses with 9 bits in the tag field, 4 in the set field, and 3 in the offset Field.

Tag	Set	offset
9	4	3

b) If this cache is 4-way set associative, what is the format of a memory address as seen by the cache?

16 bit addresses with 9 bits in the tag field, 3 in the set field, and 3 in the offset Field.

Tag	Set	offset
10	3	3

11. Suppose we have a computer that uses a memory address word size of 8 bits. This computer has a 16-byte cache with 4 bytes per block. The computer accesses a number of memory locations throughout the course of running a program. Suppose this computer uses direct-mapped cache. The format of a memory address as seen by the cache is shown here:

Tag 4 bits	Block 2 bits	Offset 2 bits
---------------	-----------------	------------------

The system accesses memory addresses in this exact order: 0x6E, 0xB9, 0x17, 0xE0, 0x4E, 0x4F, 0x50, 0x91, 0xA8, 0xA9, 0xAB, 0xAD, 0x93, and 0x94. The memory addresses of the first four accesses have been loaded into the cache blocks as shown below. (The contents of the tag are shown in binary, and the cache “contents” are simply the address stored at that cache location.)

	Tag contents	Cache Contents (represented by address)
Block 0	1110	0xE0
		0xE1
		0xE2
		0xE3

	Tag contents	Cache Contents (represented by address)
Block 1	0001	0x14
		0x15
		0x16
		0x17

	Tag contents	Cache Contents (represented by address)
Block 2	1011	0xB8
		0xB9
		0xBA
		0xBB

	Tag contents	Cache Contents (represented by address)
Block 3	0110	0x6C
		0x6D
		0x6E
		0x6F

a) What is the hit ratio for the entire memory reference sequence given above, assuming that we count the first four accesses as misses?

m m m m m h m m m m h h h m Hit 4 = 4/14 = 0.4

b) What memory blocks will be in the cache after the last address has been accessed?

	Tag contents	Cache Contents (represented by address)
Block 0	1001	0x90
		0x91
		0x92
		0x93

	Tag contents	Cache Contents (represented by address)
Block 1	1001	0x94
		0x95
		0x96
		0x97

	Tag contents	Cache Contents (represented by address)
Block 2	1010	0xA8
		0xA9
		0xAA
		0xAB

	Tag contents	Cache Contents (represented by address)
Block 3	0100	0x4C
		0x4D
		0x4E
		0x4F

16. Assume a direct mapped cache that holds 4096 bytes, in which each block is 16 bytes. Assuming that an address is 32 bits and that cache is initially empty, complete the table that follows. (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
0 x 0FF0FABA	0x0FF0	0xFAB	0xA
0 x 00000011	0x0000	0x001	0x1
0 x 0FFFFFFF	0x0111	0x111	0xF
0 x 23456719	0x2345	0x671	0x9
0 x CAFEBABE	0xCAFE	0xBAB	0xE

19. Suppose a process page table contains the entries shown below. Using the format shown in Figure 6.22a, indicate where the process pages are located in memory.

Frame	Valid bit
-	0
3	1
-	0
-	0
-	1
2	1
0	1
-	0
1	1

Frame 0 refer to page 1

Frame 1 refer to page 3

Frame 2 refer to page 0

Frame 3 refer to page 5

20. Suppose you have a byte-addressable virtual address memory system with eight virtual pages of 64 bytes each, and four 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

a) How many bits are in a virtual address?

$$64\text{bytes} \times 8 = 2^9 = 9 \text{ bits}$$

b) How many bits are in a physical address?

Virtual address have 3 bits for page but in physical memory have 4 frames that mean we use 2 bits instead of 3 bits . so $6+2 = 8$ 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. **0x0** = 000 00000 page 0 physical address => 01 000000

ii. **0x44** = 010 00100 page 2 physical address = -

iii. **0xC2** = 110 00010 page 6 physical address = -

iv. **0x80** = 100 00000 page 4 physical address => 10 00000

23. Given a virtual memory system with a TLB, a cache, and a page table, assume the following:

- A TLB hit requires 5ns.
- A cache hit requires 12ns.
- A memory reference requires 25ns.
- A disk reference requires 200ms (this includes updating the page table, cache, and TLB).
- The TLB hit ratio is 90%.
- The cache hit rate is 98%.
- The page fault rate is .001%.
- On a TLB or cache miss, the time required for access includes a TLB and/or cache update, but the access is not restarted.
- On a page fault, the page is fetched from disk, and all updates are performed, but the access is restarted.
- All references are sequential (no overlap, nothing done in parallel).

For each of the following, indicate whether or not it is possible. If it is possible, specify the time required for accessing the requested data.

a) TLB hit, cache hit

$$5 + 12 = 17 \text{ ns}$$

b) TLB miss, page table hit, cache hit

$$5 + 25 + 12 = 42 \text{ ns}$$

c) TLB miss, page table hit, cache miss

$$5 + 25 + 12 + 25 = 67 \text{ ns}$$

d) TLB miss, page table miss, cache hit

$$5 + 25 + 200000 + 12 = 200.042 \text{ ms}$$

e) TLB miss, page table miss

$$5 + 25 + 200000 + 12 + 25 = 200.067 \text{ ms}$$

Write down the equation to calculate the effective access time.

$$\begin{aligned} \text{EAT} &= H_{\text{TLB}}(5 + H_c(12) + (1 - H_c)(12 + 25)) + (1 - H_{\text{TLB}})(5 + H_p(25 + H_c(12) + (1 - H_c)(12 + 25)) + (1 - H_p)(200030 + H_c(12) + (1 - H_c)(12 + 23))) \\ &= (0.9)(5 + (0.98)(12) + (0.02)(37)) + (0.1)(5 + (0.99)(25 + 12.5) + (0.01)(200030 + 12.5)) \\ &= 220.00055 \text{ ns} \end{aligned}$$