Review Questions

4.1 How do the terms word, addressable units and unit of transfer relate to internal memory?

4.2 What is the transfer rate for a random access memory and a non-random access memory?

4.3 How does the principle of locality relate to the use of multiple memory levels?

4.4 What are the differences among direct mapping, associative mapping, and set-associative mapping?

4.5 For a direct-mapped cache, a main memory address is viewed as consisting of three fields. List and define the three fields.

4.6 For an associative cache, a main memory address is viewed as consisting of two fields. List and define the two fields.

4.7 For a set-associative cache, a main memory address is viewed as consisting of three fields. List and define the three fields.

4.8 What are the advantages of using a unified cache?

4.9 List and briefly describe the four major components of a Pentium 4 processor core.

Problems

4.1 A two-way set-associative cache consists of 128 lines, or slots, divided into several sets. The main memory contains 8K blocks of 256 words each. Show the format of main memory addresses.

4.2 A four-way set-associative cache has lines of 32 bytes and a total size of 4 kB. The 32-MB main memory is byte addressable. Show the format of main memory addresses.

4.3 For the hexadecimal main memory addresses 111111, 666666, BBBBBB, show the following information, in hexadecimal format:

a. Tag, Line, and Word values for a direct-mapped cache, using the format of Figure 4.10

b. Tag and Word values for an associative cache, using the format of Figure 4.12

c. Tag, Set, and Word values for a two-way set-associative cache, using the format of Figure 4.15

4.4 List the following values:

a. For the direct cache example of Figure 4.10: address length, number of addressable units, block size, number of blocks in main memory, number of lines in cache, size of tag

b. For the associative cache example of Figure 4.12: address length, number of addressable units, block size, number of blocks in main memory, number of lines in cache, size of tag

c. For the two-way set-associative cache example of Figure 4.15: address length, number of addressable units, block size, number of blocks in main memory, number of lines in set, number of sets, number of lines in cache, size of tag

4.5 Consider a 32-bit microprocessor that has an on-chip 16-kB four-way set-associative cache. Assume that the cache has a line size of four 32-bit words. Draw a block diagram of this cache showing its organization and how the different address fields are used to determine a cache hit/miss. Where in the cache is the word from memory location ABCDE8F8 mapped?

4.6 Given the following specifications for an external cache memory: four-way set associative; line size of two 16-bit words; able to accommodate a total of 4K 32-bit words from main memory; used with a 16-bit processor that issues 24-bit addresses. Design the cache structure with all pertinent information and show how it interprets the processor's addresses.

4.7 The Intel 80486 has an on-chip, unified cache. It contains 8 kB and has a four-way set-associative organization and a block length of four 32-bit words. The cache is organized into 128 sets. There is a single "line valid bit" and three bits, B0, B1, and B2 (the "LRU" bits), per line. On a cache miss, the 80486 reads a 16-byte line from main memory in a bus memory read burst. Draw a simplified diagram of the cache and show how the different fields of the address are interpreted.

.8 Consider a machine with a byte addressable main memory of 2^{16} bytes and block size of 8 bytes. Assume that a direct mapped cache consisting of 32 lines is used with this machine.

a. How is a 16-bit memory address divided into tag, line number, and byte number?

b. Into what line would bytes with each of the following addresses be stored?

0001	0001	0001	1011
1100	0011	0011	0100
1101	0000	0001	1101
1010	1010	1010	1010

c. Suppose the byte with address 0001 1010 0001 1010 is stored in the cache. What are the addresses of the other bytes stored along with it?

d. How many total bytes of memory can be stored in the cache?

e. Why is the tag also stored in the cache?

9 For its on-chip cache, the Intel 80486 uses a replacement algorithm referred to as **pseudo least recently used**. Associated with each of the 128 sets of four lines (labeled L0, L1, L2, L3) are three bits B0, B1, and B2. The replacement algorithm works as follows: When a line must be replaced, the cache will first determine whether the most recent use was from L0 and L1 or L2 and L3. Then the cache will determine which of the pair of blocks was least recently used and mark it for replacement. Figure 4.19 illustrates the logic.

a. Specify how the bits B0, B1, and B2 are set and then describe in words how they are used in the replacement algorithm

b. Show that the 80486 al... sider tha...

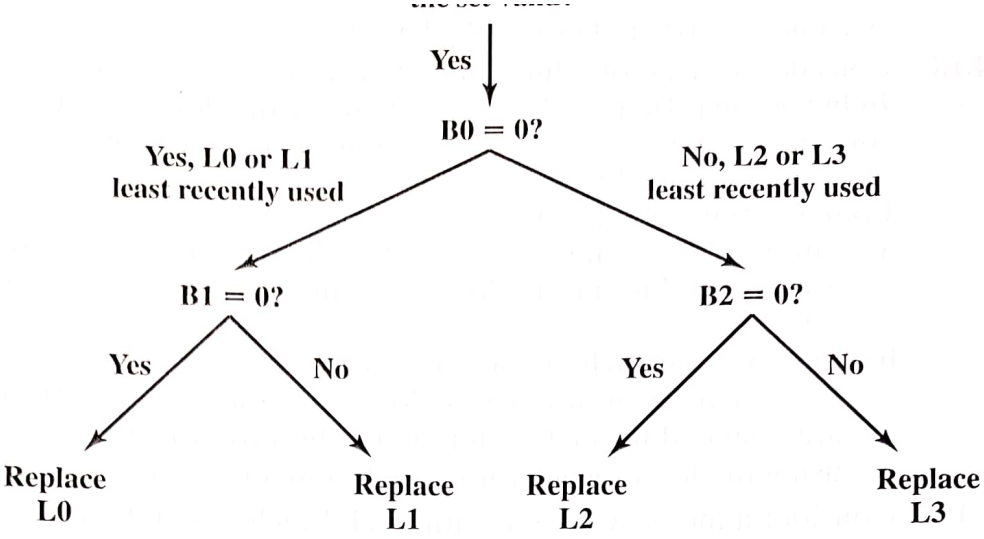

Figure 4.19 Intel 80486 On-Chip Cache Replacement Strategy

b. Assume an associative cache. Show the address format and determine the following parameters: number of addressable units, number of blocks in main memory, number of lines in cache, size of tag.

c. Assume a four-way set-associative cache with a tag field in the address of 9 bits. Show the address format and determine the following parameters: number of addressable units, number of blocks in main memory, number of lines in set, number of sets in cache, number of lines in cache, size of tag.

4.12 Consider a computer with the following characteristics: total of 1 MB of main memory; word size of 1 byte; block size of 16 bytes; and cache size of 64 kB.

a. For the main memory addresses of F0010, 01234, and CABBE, give the corresponding tag, cache line address, and word offsets for a direct-mapped cache.

b. Give any two main memory addresses with different tags that map to the same cache slot for a direct-mapped cache.

c. For the main memory addresses of F0010 and CABBE, give the corresponding tag and offset values for a fully-associative cache.

d. For the main memory addresses of F0010 and CABBE, give the corresponding tag, cache set, and offset values for a two-way set-associative cache.

4.13 Describe a simple technique for implementing an LRU replacement algorithm in a four-way set-associative cache.

4.14 Consider again Example 4.3. If the cache has a line size of 64 bytes and the main memory requires 45 ns to access the first-word and 10 ns for each word thereafter, how does the answer change?

4.15 Consider the following code:

```
for (m = 15; m > 0; m--)
    while (n < 27)
        a[m] = a[m] + n
```

a. Give one example of the spatial locality in the code.

b. Give one example of the temporal locality in the code.

4.16 Generalize Equations (4.2) and (4.3), in Appendix 4A, to *N*-level memory hierarchies.

4.17 A computer system contains a main memory of 32K 16-bit words. It also has a 4K word cache divided into four-line sets with 64 words per line. Assume that the cache is initially empty. The processor fetches words from locations 0, 1, 2, . . . , 4351 in that

4.18 Consider a cache of 4 lines of 16 bytes each. Main memory is divided into blocks of 16 bytes each. That is, block 0 has bytes with addresses 0 through 15, and so on. Now consider a program that accesses memory in the following sequence of addresses:
Once: 63 through 70.
Loop ten times: 15 through 32; 80 through 95.

a. Suppose the cache is organized as direct mapped. Memory blocks 0, 4, and so on are assigned to line 1; blocks 1, 5, and so on to line 2; and so on. Compute the hit ratio.

b. Suppose the cache is organized as two-way set associative, with two sets of two lines each. Even-numbered blocks are assigned to set 0 and odd-numbered blocks are assigned to set 1. Compute the hit ratio for the two-way set-associative cache using the least recently used replacement scheme.

4.19 Consider a memory system with L1, L2 caches and the following parameters:

$C_{c1} = 10^{-3}$ \$/bit $C_{c2} = 10^{-4}$ \$/bit $C_m = 10^{-5}$ \$/bit

$T_{c1} = 200$ ns $T_m = 1000$ ns

a. What is the cost of 500 KB of main memory using L1 cache memory?
b. What is the cost of 500 KB of main memory using L2 cache memory?
c. If the effective access time is 25% greater than the L1 cache access time, what is the hit ratio H?

4.20 **a.** Consider an L1 cache with an access time of 5 ns and a hit ratio of $H = 0.9$. Suppose that the memory access time is 100 ns and we can increase the cache access time to 6 ns. What must be the hit ratio, H, for this change to result in improved performance?

b. Repeat the above question for a cache access time of 10 ns.

4.21 Consider a single-level cache with an access time of 5 ns, a line size of 32 bytes, and a miss rate of 11%. Main memory uses a block transfer capability that has a first-word (8 bytes) access time of 100 ns and an access time of 10 ns for each word thereafter.

a. What is the access time when there is a cache miss? Assume that the cache waits until the line has been fetched from main memory and then re-executes for a hit.

b. Suppose that increasing the line size to 64 bytes reduces the miss rate to 7%. Does this reduce the average memory access time?

4.22 A computer has a cache, main memory, and a disk used for virtual memory. If a referenced word is in the cache, 9 ns are required to access it. If it is in main memory but not in the cache, 80 ns are needed to load it into the cache, and then the reference is started again. If the word is not in main memory, 8 ms are required to fetch the word from disk, followed by 80 ns to copy it to the cache, and then the reference is started again. The cache miss rate is 9% and the main memory miss rate is 30%. What is the average time in nanoseconds required to access a referenced word on this system?

4.23 Consider a cache with a line size of 64 bytes. Assume that on average 30% of the lines in the cache are dirty. A word consists of 8 bytes.

a. Assume there is a 3% miss rate (0.97 hit ratio). Compute the amount of main memory traffic, in terms of bytes per instruction for both write-through and write-back policies. Memory is read into cache one line at a time. However, for write back, a single word can be written from cache to main memory.

b. Repeat part a for a 5% rate.

c. Repeat part a for a 7% rate.

d. What conclusi