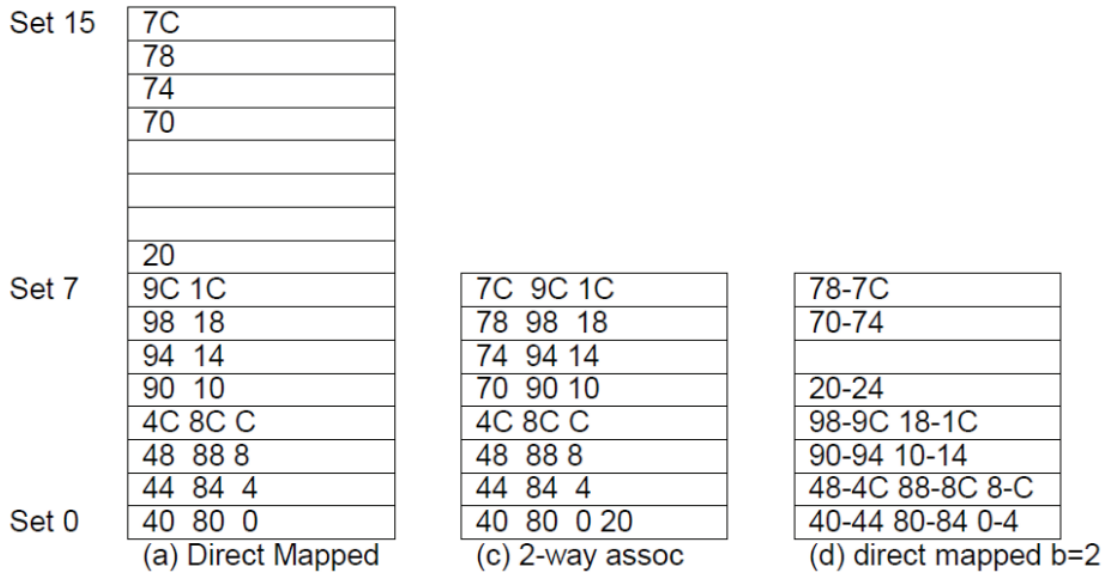
Examples from problem section and homework hints

Exercise 8.9

The figure below shows where each address maps for each cache configuration.



(a) **80% miss rate.** Addresses 70-7C and 20 use unique cache blocks and are not removed once placed into the cache. Miss rate is 20/25 = 80%.

(b) **100% miss rate.** A repeated sequence of length greater than the cache size produces no hits for a fully-associative cache using LRU.

(c) **100% miss rate.** The repeated sequence makes at least three accesses to each set during each pass. Using LRU replacement, each value must be replaced each pass through.

(d) **40% miss rate.** Data words from consecutive locations are stored in each cache block. The larger block size is advantageous since accesses in the given sequence are made primarily to consecutive word addresses. A block size of two cuts the number of block fetches in half since two words are obtained per block fetch. The address of the second word in the block will always hit in this type of scheme (e.g. address 44 of the 40-44 address pair). Thus, the second

Consecutive word access always hit: 44, 4C, 94, 9C, 24, 74, 7C, 84, 8C, 14, 1C, 4, and C. In addition, three of the eight blocks will never be replaced (70-74, 20-24, and 78-7C). Thus the hit rate is 16/25=64% and the miss rate is 9/25=36%.

Exercise 8.14

(a) The word in memory might be found in two locations, one in the on-chip cache, and one in the off-chip cache.

(b) For the first-level cache, the number of sets, S = 512 / 4 = 128 sets. Thus, 7 bits of the address are set bits. The block size is 16 bytes / 4 bytes/word = 4 words, so there are 2 block offset bits. Thus, the number of tag bits for the first-level cache is 32 - (7+2+2) = **21 bits.**

For the second-level cache, the number of sets is equal to the number of blocks, S = 256 Ksets. Thus, 18 bits of the address are set bits. The block size is 16 bytes / 4 bytes/word = 4 words, so there are 2 block offset bits. Thus, the number of tag bits for the second-level cache is 32 - (18+2+2) = **10 bits.**

(c) From Equation 8.2, $AMAT = t_{cache} + MR_{cache}(t_{MM} + MR_{MM}\, t_{VM})$. In this case, there is no virtual memory but there is an L2 cache. Thus,

$$AMAT = t_{cache} + MR_{cache}(t_{L2cache} + MR_{L2cache}\, t_{MM})$$

where, MR is the miss rate. In terms of hit rate, $MR_{cache} = 1 - HR_{cache}$, and $MR_{L2cache} = 1 - HR_{L2cache}$. Using the values given in Table 8.4,

$$AMAT = t_a + (1 - A)(t_b + (1 - B)\, t_m)$$

(d) When the first-level cache is enabled, the second-level cache receives only the "hard" accesses, ones that don't show enough temporal and spatial locality to hit in the first-level cache. The "easy" accesses (ones with good temporal and spatial locality) hit in the first-level cache, even though they would have also hit in the second-level cache. When the first-level cache is disabled, the hit rate goes up because the second-level cache supplies both the "easy" accesses and some of the "hard" accesses.

Hints on the assignment:

- 8.10 similar to 8.10 with a total of 14 accesses: 74, A0, 78, 38C, AC, 84, 88, 8C, 7C, 34, 38, 13C, 388, and 18C.
- 8.11: should be easy
- 8.13: includes some hardware components
- 8.15: considers different replacement strategies (very interesting general discussion).