Solutions for Homework 8

CIS 501, Spring 1998
April 15, 1998

1. You purchase an Acme computer with the following features:

   - 95% of all memory accesses are found in the cache.
   - Each cache block is two words, and the whole block is read on any miss.
   - The processor sends references to its cache at the rate of 10^9 words per second.
   - 25% of those references are writes.
   - Assume that the memory system can support 10^9 words per second, reads or writes.
   - The bus reads or writes a single word at a time (the memory system cannot read or write two words at once).
   - Assume at any one time, 30% of the blocks in the cache have been modified.
   - The cache uses write allocate on a write miss.

   You are considering adding a peripheral to the system, but you want to know how much o fthe memory system bandwidth is already used. Calculate the percentage of memory system bandwidth used on the average in the two cases below. Be sure to state your assumptions, if any.

   (a) The cache is write through.

   Bus traffic for a write-through cache consists of cache misses plus writes (since all writes go through to memory). For cache misses, you must read in a whole block (2 words).

   $$\begin{aligned}\text{bus traffic} &= \text{CPU\_demand * miss\_rate * block\_size +} \\ &\quad\ \text{CPU\_demand * write\_freq * word} \\ &= 10^9 * 0.05 * 2 + 10^9 * 0.25 * 1 \\ &= 0.35 * 10^9 \text{ words/sec}\end{aligned}$$

   percentage of memory system bandwidth used = $(0.35 * 10^9)/10^9 = \mathbf{0.35}$

   (b) The cache is write back.

   Bus traffic for a write-back cache consists of cache misses and write backs of dirty cache lines (which could result from reads or writes).

   $$\begin{aligned}\text{bus traffic} &= \text{CPU\_demand * miss\_rate * block\_size * (1 + \% dirty)} \\ &= 10^9 * 0.05 * 2 * (1 + 0.30) \\ &= 0.13 * 10^9 \text{ words/sec}\end{aligned}$$

   percentage of memory system bandwidth used $=(0.13 * 10^9)/10^9 = \mathbf{0.13}$

2. You are comparing the performance of different caches on a DLX machine. Use Figure 5.9, page 391, to obtain the miss rates for various types of caches. Unless otherwise stated, assume that

   - Loading (or storing) a cache line from main memory into cache takes 15 cycles to read (write) the first (4 byte) word, and one cycle for each additional word in the cache line (block).
   - Cache hit time is one cycle.
   - Unless otherwise stated for a particular problem, memory can deliver at most one word per cycle.
   - Initially, all lines in the cache are clean.
   - A write-through cache writes only the modified word back to memory, rather than the whole cache line.

(a) One option is a direct-mapped 4K cache with each cache line (block) 32 bytes long. What is the miss rate for this cache? What is the average memory access time for this cache?

The miss rate for a direct-mapped 4K cache is **0.072**.
The miss penalty is 22 clock cycles (15 for the first word, 1 for each of the remaining 7 words).

$$\begin{aligned} \text{AMAT} &= \text{hit\_time} + \text{miss\_rate} * \text{miss\_penalty} \\ &= 1 + 0.072 * (15 + 7) \\ &= \mathbf{2.584\ clock\ cycles} \end{aligned}$$

(b) Another option is to double the size of the cache. You want to decide whether the new cache should be 8K/1-way or 8K/2-way. What are the average memory access times of these two caches?

$$\begin{aligned} \text{AMAT\_8K/1-way} &= 1 + 0.046 * (15 + 7) \\ &= \mathbf{2.012\ clock\ cycles} \\ \text{AMAT\_8K/2-way} &= 1 + 0.038 * (15 + 7) \\ &= \mathbf{1.836\ clock\ cycles} \end{aligned}$$

(c) After evaluating the above mentioned caches, you decide to improve the cache fill time. By organizing the memory into four banks, your new configuration allows you to read 4 words per cycle instead of just the one previously. In addition, you double the width of the cache from 32 bytes per cache line to 64 bytes per cache line. However, the first group of words still takes 15 cucles to arrive. What is the average memory access time for this cache?

The miss penalty now takes 18 cycles (15 for the first 4 words, 1 for each of th e remaining 3 sets of 4 words).

$$\begin{aligned} \text{AMAT\_8K/2-way} &= 1 + 0.038 * (15 + 3) \\ &= \mathbf{1.684\ clock\ cycles} \end{aligned}$$

(d) Suppose you perform two writes to the cache that modify lines A and B in the cache (A and B are not in the same set in the 2-way case). Further, suppose that another line is loaded in from main memory into the cache and this line maps *into the same set as line A*. Note that for the 1-way case, the set with line A consists of only line A whereas in the 2-way case the set contains the line A as well as one more line.

i. How long will the entire process take (in cycles) on a direct-map (1-way) write-through cache?
direct-mapped write-through cache: The modified words must be written to memory, since the cache is write-through. A whole line (32 bytes) must be read into the cache.
15 (modify A) + 15 (modify B) + 22 (load new line into A) = **52 cycles**

ii. How long will the entire process take (in cycles) on a 2-way write-through cache?
2-way write-through cache: Modified words still must be written back, and new line must still be loaded into cache.
15 (modify A) + 15 (modify B) + 22 (load new line into cache) = **52 cycles**

iii. How long will the entire process take (in cycles) on a direct-map write-back cache?
direct-mapped write-back cache: Modifying lines A and B are not written to memory since the cache is write-back. However, when the new line is loaded into A, the dirty line must be written back to memory.
1 (modify A) + 1 (modify B) + 22 (load new line into A) + 22 (write back dirty line A) = **46 cycles**

iv. How long will the entire process take (in cycles) on a 2-way write-back cache?
2-way write-back cache: Modifying lines A and B are not written to memory since the cache is write-back. Since the cache is 2-way set associative, the new line is written to the other line in the set with line A. Thus, A does not need to be written back to memory.
1 (modify A) + 1 (modify B) + 22 (load new line into other line in set with A) = **24 cycles**