CONCORDIA UNIVERSITY

DEPARTMENT OF
COMPUTER SCIENCE AND SOFTWARE ENGINEERING

COMP 228 Winter 2008

ASSIGNMENT 5

Issued: March 25, 2008 Due: April 10, 2008

NOTE: No late submission will be accepted. Solution will be posted April 11. (Final Exam is scheduled for April 15.)

1. Cache Memory Design and Performance [60%]

You are given the following cache-based memory specification:
Cache memory size: 1M bytes
Main memory (RAM) size: 2G bytes
Block/line size: 256 bytes
Cache hit time: 1 cycle
Cache miss penalty: 28 cycles
Main memory access time: 4 cycles
Cache mapping method: direct mapping

Consider the execution of the following program fragment:

```
again  mov   eax, [ebx + esi*4]     //S1
       add   [edx + esi*4], eax     //S2
       inc   esi                    //S3
       loop  again                  //S4
       ::::::
```

Assume that prior to the execution, ebx = 1001100h, ecx = 1000h, edx = 101000h, esi = 0, and eip = 2001000h. [Hint: register eip contains the address corresponding to again.]

(a) Determine the following:
- (i) The total number of cache blocks.
- (ii) Cache address format (the address fields in a cache address and the number of bits in each field).
- (iii) Memory address format.

(b) Determine the following:
- (i) The cache line (number) that is used to buffer memory address 1001100h.
- (ii) The cache line (number) that is used to buffer 2001000h.
- (iii) The cache lines that will be accessed for data operands during the execution of the given program fragment.
- (iv) The cache lines that will be accessed for instructions during the execution of the given program fragment.

(c) Identify the cache misses that occur during the first two iterations of the given program loop.

(d) Consider the execution of the given program fragment. Determine the following:
   (i) The total number of instructions executed (and hence fetched).
   (ii) The toal number of data operand accesses.
   (iii) The total number of cache misses caused by instruction fetches.
   (iv) The total number of cache misses caused by data operand accesses.
   (v) The hit ratio.
   (vi) The memory speedup ratio arising from the use of the cache.

(e) Will increasing the cache size from 1M bytes to 2M bytes improve the hit ratio of the given program? Why?

(f) Will increasing the block size from 256 bytes to 512 bytes improve the hit ratio of the given program? Why?

(g) Will changing the mapping method from direct mapping to set-associative mapping with a set size of two (cache lines) improve the hit ratio of the given program? Why?

(h) Why is register storage typically much smaller than cache storage? What is the main difference between these two storages in a computer system?

(i) Explain how temporal and spatial locality often exist in fetching instructions from a sequential program.

2. Input/Output techniques [40%]

(a) Consider the following three distinct input/output devices: mouse, printer, and disk.
   (i) Compare their usage characteristics in terms of frequency and synchrony of interaction (between the device/external world and the CPU/memory system).
   (ii) Why is programmed I/O unlikely to be a good choice in managing these three types of devices in a personal computer environment?
   (iii) Will interrupt I/O be the most reasonable choice in managing a mouse? A printer? A disk? Justify your reason.
   (iv) Repeat (iii) for DMA I/O.

(b) Explain carefully the sharing of the system bus by DMA interfaces and the CPU(s) in a computer system, addressing in particular the following aspects.
   (i) How does a DMA interface steal bus cycles from the CPU to perform a memory transaction?
   (ii) Under what situations the CPU may not be slowed down at all even though some DMA(s) are conducting transactions with the memory.
   (iii) What problem may arise if DMA transactions are delayed for too long?

(c) Do problem 2 (page 381) of your text.

(d) Do problem 5 (page 381) of your text.

(e) Do problem 19 (page 382) of your text. [Note: assume in this problem, the track-to-track seek time is actually the average seek time for the right track.]