Computer Organization and Structure

Homework #5
Due: 2012/1/3

1. Caches are important to providing a high performance memory hierarchy to processors. Below is a list of 32-bit memory address references, given as word addresses.

a.	1,134,212,1,135,213,162,161,2,44,41,221
b.	6,214,175,214,6,84,65,174,64,105,85,215

   a. For each of these references, identify the binary address, the tag, and the index given a direct-mapped cache with 16 one-word blocks. Also list if each reference is a hit or a miss, assuming the cache is initially empty.

   b. For each of these references, identify the binary address, the tag, and the index given a direct-mapped cache with two-word blocks and a total size of eight blocks. Also list if each reference is a hit or a miss, assuming the cache is initially empty.

   c. You are asked to optimize a cache design for the given references. There are three direct-mapped cache designs possible, all with a total of eight words of data: C1 has one-word blocks, C2 has two-word blocks, and C3 has four-word blocks. In terms of miss rate, which cache design is the best? If the miss stall time is 25 cycles, and C1 has an access time of 2 cycles, C2 takes 3 cycles, and C3 takes 5 cycles, which is the best cache design?

There are many different design parameters that are important to a cache's overall performance. The table below lists parameters for different direct-mapped cache designs.

	Cache data size	**Cache block size**	**Cache access time**
a.	64 KB	1 word	1 cycle
b.	64 KB	2 word	2 cycle

   d. Calculate the total number of bits required for the cache listed in the table, assuming a 32-bit address. Given that total size, find the total size of the closest direct-mapped cache with 16-word blocks of equal size or grater. Explain why the second cache, despite its larger data size, might provide slower performance than the first cache.

   e. Generate a series of read requests that have a lower miss rate on a 2 KB two-way set associative cache than the cache listed in the table. Identify one possible solution that would make the cache listed in the table have an equal or lower miss rate than the 2 KB cache. Discuss the advantages and disadvantages of such a solution.

   f. (Block address) modulo (Number of blocks in the cache) shows the typical method to index a direct-mapped cache. Assuming a 32-bit address and 1024 blocks in the cache, consider a different indexing function, specially (Block address[31:27] XOR Block address[26:22]). Is it possible to use this to index a direct-mapped cache? If so, explain why and discuss any changes that might need to be made to the cache. If it is not possible, explain why.

2. Recall that we have two write policies and write allocation policies, their combinations can be implemented at either in L1 or L2 cache.

	L1	**L2**
a.	Write-back, write allocate	Write-through, non write allocate
b.	Write-back, write allocate	Write-through, write allocate

   a. Buffers are employed between different levels of memory hierarchy to reduce access latency. For this given configuration, list the possible buffers needed between L1 and L2 caches, as well as L2 cache and memory.
   b. Describe the procedure of handling an L1 write miss, considering the component involved and the possibility of replacing a dirty block.
   c. For a multilevel cache (a block can only reside in one of the L1 and L2 caches) configuration, describe the procedure of handling an L1 write miss, considering the component involved and the possibility of replacing a dirty block.

   Consider the following program and cache behaviors.

	Data reads per 1000 instructions	**Data writes per 1000 instructions**	**Instruction cache miss rate**	**Data cache miss rate**	**Block size (byte)**
a.	200	160	0.20%	2%	8
b.	180	120	0.20%	2%	16

   d. For a write-through, write-allocate cache, what is the minimum read and write bandwidths (measured by byte-per-cycle) needed to achieve a CPI of 2?
   e. For a write-back, write-allocate cache, assuming 30% of replaced data cache blocks are dirty, what is the minimal read and write bandwidths needed for a CPI of 2?
   f. What are the minimal bandwidths needed to achieve the performance of CPI=1.5?

3. The following table is a stream of virtual addresses as seen on a system. Assume 4KB pages, a four-entry fully associative TLB, and true LRU replacement. If pages must be brought in from disk, increment the next largest page number.

a.	4095,31272,15789,15000,7193,4096,8912
b.	9452,30964,19136,46502,38110,16653,48480

   TLB

Valid	Tag	Physical Page Number
1	11	12
1	7	4
1	3	6
0	4	9

Page table

Valid	Physical page or in disk
1	5
0	Disk
0	Disk
1	6
1	9
1	11
0	Disk
1	4
0	Disk
0	Disk
1	3
1	12

a. Given the address stream in the table, and the shown initial state of the TLB and page table, show the final state of the system. Also list for each reference if it is a hit in the TLB, a hit in the page table, or a page fault.

b. Repeat the above sub-problem, but this time use 16KB pages instead of 4KB pages. What would be some of the advantages of having a larger page size? What are some of the disadvantages?

c. Show the final contents of the TLB if it is two-way set-associative. Also show the contents of the TLB if it is direct-mapped? Discuss the importance of having a TLB to high performance. How would virtual memory accesses be handled if there were no TLB?

There are several parameters that impact the overall size of the page table. Listed below are several key page table parameters.

	Virtual address size	Page size	Page table entry size
a.	32 bits	4KB	4 bytes
b.	64 bits	16KB	8 bytes

d. Given the parameters in the table above, calculate the total page table size for a system running five applications that utilize half of the memory available.

e. Given the parameters in the table above, calculate the total page table size for a system running five applications that utilize half of the memory available, given a two-level page table approach with 256 entries. Assume each entry of the main page table is 6 bytes. Calculate the minimum and maximum amount of memory required.

f. A cache designer wants to increase the size of a 4KB virtually indexed, physically tagged cache. Given the page size listed in the table above, is it possible to make a 16KB direct-mapped cache, assuming two words per block? How would the designer increase the data size of the cache?