

University of California, Santa Barbara
Dept. of Electrical and Computer Engineering
ECE154 – Computer Architecture
Homework #6

1. **(10 points)** Using the series of addresses below, show the hits and misses and final cache contents for a two-way set-associative cache with one-word blocks and a total size of 16 words. Assume LRU replacement. Note that these are *word addresses*, not byte addresses.

0, 1, 16, 0, 32, 8, 3, 24, 32, 16

2. **(10 points)** This exercise concerns caches of unusual sizes. Can you make a fully associative cache containing exactly 3K words of data? How about a set-associative cache or a direct-mapped cache containing exactly 3K words of data? For each of these, describe how or why not. Remember that $1K = 2^{10}$.

4. **(10 points)** To capture the fact that the time to access data for both hits and misses affects performance, designers often use average memory access time (AMAT) as a way to examine alternative cache designs. Average memory access time is the average time to access memory considering both hits and misses and the frequency of different access. The unit is either expressed in cycles / access or time / access, depending on what the problem gives you.

The most basic equation is: $AMAT = AccessTime + missrate * misspenalty$.

You want to look at the performance of two caches configurations by comparing the AMATs of each one. The first is a single, large cache. This 2M cache has an access time of 8 cycles and hit rate of 99%. The second configuration has two caches. The first is a 256K cache with an access time of 1 cycle and hit rate of 96%. The second cache is a 1M cache with an 7 cycle access time and a 65% hit rate. Assume main memory requires 100 cycles. Hint: You still want to write AMAT with respect to the top level cache. Just think of what its miss penalty must be (since it is no longer the constant DRAM time).

5. **(10 points)** Cache C1 is direct-mapped with 16 one-word blocks. Cache C2 is direct-mapped with 4 four-word blocks. Assume that the miss penalty for C1 is 8 clock cycles and the miss penalty for C2 is 11 clock cycles. Assuming that the caches are initially empty, find a reference string for which C2 has a **lower miss rate** but spends **more cycles on cache misses** than C1. Use word addresses.

6. **(10 points)** The virtual address space is 28 bits, and the physical address space is 20 bits. The page size is 8kB, and the computer is byte addressable. The TLB initial state is shown below. Use FIFO replacement, and assume they were inserted beginning from the first entry, proceeding downwards.

V	VPN	PPN
---	-----	-----

1	0x5	0x7
1	0x7f	0x8
1	0x7	0x9
1	0x58	0xa
1	0x10	0x3
0	0x23	0x15
0	0x0	0x0
0	0x1	0x2

Tell which of the following accesses are hits and which are misses, and show the final state of the TLB.

0xa5f1 0xe7f8 0x7f30 0x202a8 0x7583 0x580392 0xa304

7. (40 points)

- a) (5 points) Assume that we have a 32-bit processor (with 32-bit words) and that this processor is byte-addressed (i.e. addresses specify bytes). Suppose that it has a 512-byte cache that is two-way set-associative, has 4-word cache lines, and uses LRU replacement. Split the 32-bit address into “tag”, “index”, and “cache-line offset” pieces. Which address bits comprise each piece? Remember: 1 word = 4 bytes

Tag:

Index:

Cache-line offset:

- b) (5 points) How many sets does this cache have?
- c) (20 points) Draw a block diagram for this cache. Show a 32-bit address coming into the diagram and a 32-bit data result and “Hit” signal coming out. Include all of the comparators in the system and any muxes as well. Include the data storage memories (indexed by the “Index”), the tag matching logic, and any muxes. You can indicate RAM with a simple block, but make sure to label address widths and data widths. Make sure to label the function of various blocks and the width of any buses.
- d) (15 points) Below is a series of memory read references set to the cache from part (a). Assume that the cache is initially empty and classify each memory references as a hit or a miss. Identify each miss as either **compulsory, conflict, or capacity**. One example is shown. *Hint: start by splitting the address into components. Show your work.*

Address	Hit/Miss?	Miss Type?
0x300	Miss	Compulsory
0x1BC		
0x206		
0x109		
0x308		
0x1A1		
0x1B1		
0x2AE		
0x3B2		
0x10C		
0x205		
0x301		
0x3AE		
0x1A8		
0x3A1		
0x1BA		

e) (5 points) Calculate the miss rate and hit rate.