Lab 5: Multi-Level Memory System

ACME Memory Systems Inc developed a data memory for our processors, but they neglected to give us a databook telling us the exact features! To help recover this, I'm having each student test a specific instance and figure out the features. I've already done this for USERID 0, with the filled in results attached to this lab.

The file lab5.sv creates the memory, and has a testbench with lots of helpful tasks – tasks are small subroutines you can call in your testbench to do different actions. A simple testbench is already set up for you. NOTE: DON'T BOTHER INTEGRATING THIS MEMORY WITH LAB #4!!! Just test the memory by itself.

To get started, change the USERID field to your student ID (if you are working in teams, use the larger student ID). Then, test your memory to figure out the different features. Note: I'd STRONGLY advise you to figure out all features you can via read operations without writes, and only do writes for features you cannot figure out in any other way.

Turn-in
Give your TA your filled in answer sheet during lab office hours.

Grading
For each feature you get correct, you get 2 points. If you get a feature wrong, you lose 2 points. Leaving a feature blank is 0 points (i.e. there's no advantage to guessing). Your score will be your total points divided by the possible points for your instance of the lab.

Known Features
- The memory has at least an L1 and an L2. Some also have an L3.
- The blocksize for a level of the cache is never smaller than the blocksize for the cache above it. Blocks are always at least 8 bytes (1 double-word) long.
- The total capacity of a cache (number of blocks * blocksize) increases as you go down the memory hierarchy.
- Hit times are just the time to hit in that cache – remember that a hit in the L2 pays the hit time in the L1 (to detect the L1 miss) plus the hit time in the L2.
- The associativity for a level of the cache is never smaller than the associativity for the cache above it.
- N-way Set Associative caches always have an N that is a power of 2.
- If a write misses on a cache level, that data is not written to that level, but is instead sent to the next level of the memory hierarchy. Some computers do support "write allocate", which creates a new entry on a write miss, but these caches do not.

Helpful terms
Blocksize: The number of bytes of data in each cache entry.
Write Back: If a write hits in a cache level, data is just written there.
Write Through: If a write hits in a cache level, data is also written to the next level as well.

Write Buffer: A 1-entry buffer that handles writing to the next level of the cache. If data must be written to a lower level in the cache, the buffer handles the access while allowing the upper level of the cache to proceed. Thus, if there is an L1 write buffer, and data must be written to the L2, the L1 will tell the processor it is ready for the next access while the write buffer actually does that write. However, a subsequent access (read or write) to the L2 will block until that write finishes. Using the write buffer may or may not take 1 cycle more than hitting at that cache level.

Feature	Value
L1 Cache	
Blocksize (in Bytes):	
Number of Blocks:	
Hit Time (cycles):	
Associativity: (Direct Mapped: 1, Fully Associative:# of blocks)	
Replacement strategy (1: LRU, 0: Random). Skip this question if the cache level is Direct Mapped.	
Write policy: (1: Write-Through, 0: Write-Back)	
Is there a write buffer?	
L2 Cache	
Blocksize (in Bytes):	
Number of Blocks:	
Hit Time (cycles):	
Associativity: (Direct Mapped: 1, Fully Associative:# of blocks)	
Replacement strategy (1: LRU, 0: Random). Skip this question if the cache level is Direct Mapped.	
Write policy: (1: Write-Through, 0: Write-Back)	
Is there a write buffer?	
L3 Cache	
Is there an L3 cache? (If no, skip the rest of the L3 questions)	
Blocksize (in Bytes):	
Number of Blocks:	
Hit Time (cycles):	
Associativity: (Direct Mapped: 1, Fully Associative:# of blocks)	
Replacement strategy (1: LRU, 0: Random). Skip this question if the cache level is Direct Mapped.	
Write policy: (1: Write-Through, 0: Write-Back)	
Is there a write buffer?	
Main Memory	
What is the hit time for just this level?	

Feature	Value
L1 Cache	
Blocksize (in Bytes):	16
Number of Blocks:	8
Hit Time (cycles):	2
Associativity: (Direct Mapped: 1, Fully Associative:# of blocks)	1
Replacement strategy (1: LRU, 0: Random). Skip this question if the cache level is Direct Mapped.	---
Write policy: (1: Write-Through, 0: Write-Back)	0
Is there a write buffer?	No
L2 Cache	
Blocksize (in Bytes):	16
Number of Blocks:	16
Hit Time (cycles):	8
Associativity: (Direct Mapped: 1, Fully Associative:# of blocks)	2
Replacement strategy (1: LRU, 0: Random). Skip this question if the cache level is Direct Mapped.	0
Write policy: (1: Write-Through, 0: Write-Back)	0
Is there a write buffer?	No
L3 Cache	
Is there an L3 cache? (If no, skip the rest of the L3 questions)	No
Blocksize (in Bytes):	---
Number of Blocks:	---
Hit Time (cycles):	---
Associativity: (Direct Mapped: 1, Fully Associative:# of blocks)	---
Replacement strategy (1: LRU, 0: Random). Skip this question if the cache level is Direct Mapped.	---
Write policy: (1: Write-Through, 0: Write-Back)	---
Is there a write buffer?	---
Main Memory	
What is the hit time for just this level?	60