Test Plan for Last Level Cache (LLC)

Viraj Pashte, Zoheb Mir, Shushruth Paduru, Sanket Vinchurkar

ECE_585_Fall_2022

AIM: Simulation of the last level cache (LLC) for a new processor that can be used with up to three other processors in a shared memory configuration.

CONDITIONS: We must model communication between our LLC and the next higher-level cache, bus operations that our LLC performs, snoop results that our LLC reports on the bus in response to snooping the simulated bus operations of other processors and their caches.

Coding Language: SystemVerilog 100%

Technical Specification:

The L2 cache design implemented in the project has the following specifications:

▫ Total Capacity	=	16MB
▫ Byte Line	=	64 bytes
▫ Associativity	=	8-way set
▫ Total Line	=	2^{15} – 32768 sets
▫ Coherence Protocol	=	MESI
▫ Replacement Policy	=	Pseudo-LRU
▫ Policy decisions for write	=	Write back
▫ Policy decisions for write miss	=	Write allocate

L1 cache specifications:

▫ Byte Line	=	64 bytes
▫ Associativity	=	4-way set

Tests:

1. HIT Test: Check the Indexes of the given addresses and see if it matches with the request. If it matches, then it is a cache hit.
2. MISS Test: Check the Indexes of the given addresses and see if it matches with the request. It should not match, so it is a cache miss.
3. HIT & MISS SET Test: Combination of addresses which shows the functional working of our Last Level Cache. It should have addresses which will show a HIT and after some reads show MISS. This will test the code for a combination of multiple results
4. READ Test: Check the Instruction given to the cache by the processor and if a HIT, read the address and send it to the processor. If a MISS, send a DRAM request through BUS Operation and write the address in L1 &L2 cache.
5. WRITE Test: Check the Instruction given to the cache by the processor and if a HIT, write the bytes to cache line. If a MISS, Initiate DRAM request and write back to cache L1 & L2.
6. MESI Protocol: Read the cache line and put it in the correct state. Check if the address is in the correct state [Modified, Exclusive, Shared, Invalid].
7. PLRU: Check if all the lines of a set are full and the latest request is a write. If it is a miss, take the data from the bus and the code must identify which line to evict to store the latest request in the L2 cache.
8. SNOOPING(READ): The first address writes data from cache line and initial state of the address gets modified and a snoop request is made to same address. State moves to shared.

9. SNOOPING(WRITE): Performing a snoop write on an address which is not present in cache. The initial and final states will be invalidated.
10. SNOOPING (INVALIDATE): The first address reads the data from cache line and the initial state of the address is shared and a snoop invalidate is invoked on same address. The state moves to invalidate to the address.
11. SNOOPING (READ WITH INTENT TO MODIFY): The first address reads the data from cache line and the initial state of the address is Exclusive and a snoop rwim is invoked on same address. The state moves to invalidate to the address.
12. BUS OPERATIONS: Performing read and write operations and checking for bus operations using $display.

EXPECTED TEST RESULTS:

a. Cache Reads:

Tag	index (set)	MESI
0	0000000	S
1	0000000	S
2	0000000	E
3	0000000	E
4	0000000	S
5	0000000	S
6	0000000	S
7	0000000	E

CACHE READS: 8

CACHE MISS : 8

b. Cache Write

Tag	index (set)	MESI
0	0000000	M
1	0000000	M
2	0000000	M
3	0000000	M
4	0000000	M
5	0000000	M
6	0000000	M
7	0000000	M

CACHE WRITES: 8

CACHE MISSES: 8

c. Cache Set Fill

Tag	Set (Index)		MESI
24	45DF		
19	5D17		
	34	2429	
	7E	0890	
	03	29BE	

CACHE READS: 8

CACHE HITS: 3

CACHE MISS: 5

d. Snoop Write

CACHE READ: 1

CACHE MISS: 1

MESI: I

e. SNOOP Read

TAG	INDEX(SET)	MESI
09	0D15	S

Cache Read: 2

Cache Write: 1

Cache Hit: 2

Cache Miss: 1

f. Bus Operation

“BusOp: READ, Address: 0000000, Snoop Result: HIT”

“BusOp: READ, Address: 0020001, Snoop Result: HITM”

“BusOp: READ, Address: 0040002, Snoop Result: NOHIT”

“BusOp: READ, Address: 0060003, Snoop Result: NOHIT”