ECE 585- FALL 2024 FINAL PROJECT

SIMULATION
OF
LAST LEVEL CACHE

GROUP 10

HARSHAL VENKAT KAPPA (harshal@pdx.edu)

MOULYA RAJU MACHOHALLI THIMMARAJU (moulya@pdx.edu)

RAKSHITH RAJASHEKARA BHARADWAJ (rbharadw@pdx.edu)

SUBRAMANYA DHANANJAYA (subrama@pdx.edu)

Objective:

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.

Design Choices:

Used System Verilog for Simulation

Design Specifications:

Cache total capacity : 16MB

Line size : 64-byte

Associativity : 16-way

Cache writes hit policy : Write BackCache

Writes miss policy : Write Allocate

Replacement policy : Pseudo – LRU

Cache Coherence : MESI Protocol

Write Hit Policy : Write Back

Write Miss Policy : Write Allocate

Inclusivity : Ensures all L1 cache lines exist in LLC Caches maintains inclusivity

Format of trace file contents:

n address Where n is

0 read request from L1 data cache

1 write request from L1 data cache

2 read request from L1 instruction cache

3 snooped read request

4 snooped write request

5 snooped read with intent to modify request (RWIM)

6 snooped invalidate command

8 clear the cache and reset all state

9 print contents and state of each valid cache line (doesn't end simulation!)

Assumptions:

Assume CPU address is 32-bit

Processor's next higher-level cache uses 64-byte lines and is 4-way set associative For snoop result, assume LSB 2 bits of Byte select to be

00 HIT

01 HITM

10 NO HIT

SOURCE CODE:

Cache Package: cache_config_pkg

All the Cache Configuration parameters are defined in this package and this package is imported in Cache_simulator.sv

Project Overview:

LLC Cache module: cache_simulator

Reads addresses from the trace file, performs address Partition to determine Tag, index, and Byte Select, determines hit or miss, and parallelly considers snoop results and performs snoop operations.

Bus Operations:

- Models READ, WRITE, INVALIDATE, and RWIM operations.

Snooping:

- Implements **GetSnoop** and **PutSnoop** to simulate the reporting of snoop results by other caches and reporting the results of our snooping bus operations performed by other caches.

MESI Protocol:

- Simulates state transitions during coherence operations.

Replacement Policy:

- Uses pseudo-LRU for evictions.

Simulation Modes:

- Silent Mode: Displays only summary statistics.
- Normal Mode: Displays detailed operations, including bus transactions and snoop results.

Message Types and Their Functionality:

GETLINE:

Purpose: Request data from L1 for a cache line that is in the Modified state.

Use Case:

When L2 needs updated data that may have been modified in L1 (e.g., during an eviction or a coherence operation), it issues a GETLINE to fetch the latest version of the line.

SENDLINE:

Purpose: Send a requested cache line from L2 to L1.

Use Case:

When L1 misses on a line and requests it from L2, L2 responds with a SENDLINE message containing the requested data.

INVALIDATELINE:

Purpose: Invalidate a specific cache line in L1.

Use Case:

When L2 determines that L1 should no longer hold a cache line (e.g., because another core has modified it or due to eviction), it sends INVALIDATELINE to ensure L1 invalidates the line.

EVICTLINE:

Purpose: Evict a line from L1 when L2's replacement policy demands it.

Use Case:

If the line is in the Modified state in L1, L2 issues a GETLINE to retrieve the modified data first, followed by an INVALIDATELINE to evict the line.

If the line is not modified in L1, L2 directly issues an INVALIDATELINE to remove the line.

MESI PROTOCOL:

- **Read Miss:**
    - If the line is in the Shared state, it can be read directly.
    - If the line is in the Exclusive or Modified state, it transitions to the Shared state and is read.
    - If the line is Invalid, it is fetched from memory and transitions to the Exclusive state.
- **Write Miss:**
    - The line is fetched from memory and transitions to the Modified state.
    - A BusRdX signal is sent to invalidate other caches holding the line.
- **Write Hit:**
    - If the line is in the Modified state, no action is required.
    - If the line is in the Exclusive state, it transitions to the Modified state.
    - If the line is in the Shared state:
        - A BusUpgr signal is sent to other caches.
        - The local cache transitions to the Modified state.
        - Other caches holding the line transition to the Invalid state.

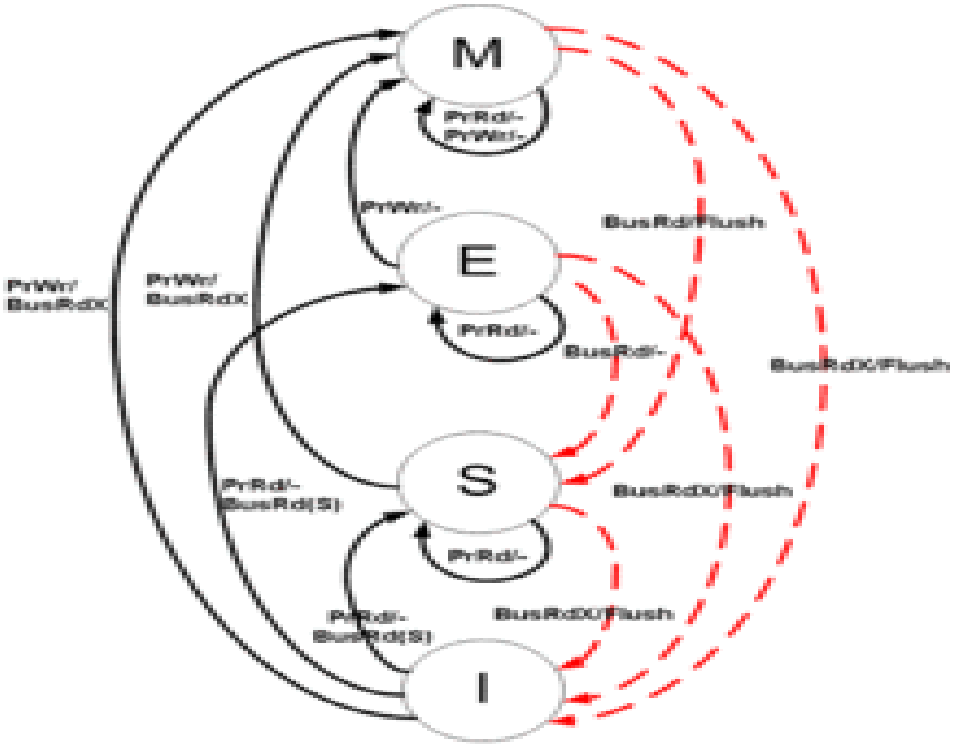


Fig: State diagram for MESI protocol
Red: Bus initiated transaction. Black: Processor initiated transaction

PLRU IMPLEMENTATION:

PLRU (Pseudo Least Recently Used) is a simplified version of the Least Recently Used (LRU) cache replacement policy. It keeps track of which cache lines were recently accessed using a binary tree of bits instead of tracking the full usage order.

- For a 16-way set associative cache, PLRU uses a 15-bit tree. Each bit points to either the left or right half of the cache, helping decide which part was accessed less recently.
- When a new cache line needs to replace an old one, the PLRU bits guide the search for the "least recently used" line.
- After a cache line is accessed or replaced, the PLRU tree bits are updated to reflect the most recent access.

For a 16-way set associative cache, the PLRU (Pseudo-LRU) replacement policy uses a binary tree structure with 15 bits to track usage across 16 cache lines (leaves).

1. Tree Structure:
    - The tree has 15 nodes, each storing a single bit indicating whether the left or right subtree was accessed more recently.

- The leaves represent the 16 cache lines (L0 to L15).

2. Replacement Decision:
    - To find the least recently used line, traverse the tree starting from the root. Follow the direction indicated by each node's bit (e.g., 0 for left, 1 for right) until reaching a leaf.
3. Update on Access:
    - When a cache line is accessed or replaced, the tree bits along the traversal path are updated to indicate the recently accessed direction.

Output:

Trace_0:

```
vsim -c +trace_file=./traces/t0.din cache_simulator -do "run -all" -do "quit -sim"

# vsim -c "+trace_file=./traces/t0.din" cache_simulator -do "run -all" -do "quit -sim"

# Start time: 10:03:32 on Dec 10,2024

# ** Note: (vsim-8009) Loading existing optimized design _opt

# Loading sv_std.std

# Loading work.cache_config_pkg(fast)

# Loading work.CACHE_NEW_2_sv_unit(fast)

# Loading work.cache_simulator(fast)

# run -all

# Successfully opened the trace file './traces/t0.din'.

# L2: SENDLINE 00000000

#

# Tag and state are as follows tag:0 , updated_mesi_state:Shared

# BusOp: READ, Address: 0, Snoop Result: HIT

# L2: SENDLINE 0000003f

#

# Tag and state are as follows tag:0 , updated_mesi_state:Shared

# printing contents and state of each valid cache line 0000,000,00,0010

# Sucessfully parsed

# Cache hit ratio = 0.500000
```

Number of cache reads= 2

Number of cache writes = 0

Number of cache hits 1.000000

Number of cache misss 1.000000

Finished reading the file.

quit -sim

End time: 10:03:36 on Dec 10,2024, Elapsed time: 0:00:04

Errors: 0, Warnings: 1

Trace_output2:

vsim -c +trace_file=./traces/t2.din cache_simulator -do "run -all" -do "quit -sim"

vsim -c "+trace_file=./traces/t2.din" cache_simulator -do "run -all" -do "quit -sim"

Start time: 10:04:13 on Dec 10,2024

** Note: (vsim-8009) Loading existing optimized design _opt

Loading sv_std.std

Loading work.cache_config_pkg(fast)

Loading work.CACHE_NEW_2_sv_unit(fast)

Loading work.cache_simulator(fast)

run -all

Successfully opened the trace file './traces/t2.din'.

L2: SENDLINE 00000000

#

Tag and state are as follows tag:0 , updated_mesi_state:Shared

BusOp: READ, Address: 0, Snoop Result: HIT

L2: SENDLINE 00200001

#

Tag and state are as follows tag:2 , updated_mesi_state:Shared

BusOp: READ, Address: 200001, Snoop Result: HITM

L2: SENDLINE 00400002

#

Tag and state are as follows tag:4 , updated_mesi_state:Exclusive

BusOp: READ, Address: 400002, Snoop Result: NOHIT

L2: SENDLINE 00600003

#

Tag and state are as follows tag:6 , updated_mesi_state:Exclusive

BusOp: READ, Address: 600003, Snoop Result: NOHIT

L2: SENDLINE 00800004

#

Tag and state are as follows tag:8 , updated_mesi_state:Shared

BusOp: READ, Address: 800004, Snoop Result: HIT

L2: SENDLINE 00a00005

#

Tag and state are as follows tag:a , updated_mesi_state:Shared

BusOp: READ, Address: a00005, Snoop Result: HITM

L2: SENDLINE 00c00006

#

Tag and state are as follows tag:c , updated_mesi_state:Exclusive

BusOp: READ, Address: c00006, Snoop Result: NOHIT

L2: SENDLINE 00e00007

#

Tag and state are as follows tag:e , updated_mesi_state:Exclusive

BusOp: READ, Address: e00007, Snoop Result: NOHIT

L2: SENDLINE 01000000

#

Tag and state are as follows tag:10 , updated_mesi_state:Shared

BusOp: READ, Address: 1000000, Snoop Result: HIT

L2: SENDLINE 01200001

#

Tag and state are as follows tag:12 , updated_mesi_state:Shared

BusOp: READ, Address: 1200001, Snoop Result: HITM

L2: SENDLINE 01400002

#

Tag and state are as follows tag:14 , updated_mesi_state:Exclusive

BusOp: READ, Address: 1400002, Snoop Result: NOHIT

L2: SENDLINE 01600003

#

Tag and state are as follows tag:16 , updated_mesi_state:Exclusive

BusOp: READ, Address: 1600003, Snoop Result: NOHIT

L2: SENDLINE 01800004

#

Tag and state are as follows tag:18 , updated_mesi_state:Shared

BusOp: READ, Address: 1800004, Snoop Result: HIT

L2: SENDLINE 01a00005

#

Tag and state are as follows tag:1a , updated_mesi_state:Shared

BusOp: READ, Address: 1a00005, Snoop Result: HITM

L2: SENDLINE 01c00006

#

Tag and state are as follows tag:1c , updated_mesi_state:Exclusive

BusOp: READ, Address: 1c00006, Snoop Result: NOHIT

L2: SENDLINE 01e00007

#

Tag and state are as follows tag:1e , updated_mesi_state:Exclusive

BusOp: READ, Address: 1e00007, Snoop Result: NOHIT

printing contents and state of each valid cache line 0000,000,00,0010

printing contents and state of each valid cache line 0000,000,00,0010

printing contents and state of each valid cache line 0000,000,00,0001

printing contents and state of each valid cache line 0000,000,00,0001

printing contents and state of each valid cache line 0000,000,00,0010

printing contents and state of each valid cache line 0000,000,00,0010

printing contents and state of each valid cache line 0000,000,00,0001

printing contents and state of each valid cache line 0000,000,00,0001

printing contents and state of each valid cache line 0000,000,00,0010

printing contents and state of each valid cache line 0000,000,00,0010

printing contents and state of each valid cache line 0000,000,00,0001

printing contents and state of each valid cache line 0000,000,00,0001

printing contents and state of each valid cache line 0000,000,00,0010

printing contents and state of each valid cache line 0000,000,00,0010

printing contents and state of each valid cache line 0000,000,00,0001

printing contents and state of each valid cache line 0000,000,00,0001

Sucessfully parsed

Cache hit ratio = 0.000000

Number of cache reads= 16

Number of cache writes = 0

Number of cache hits 0.000000

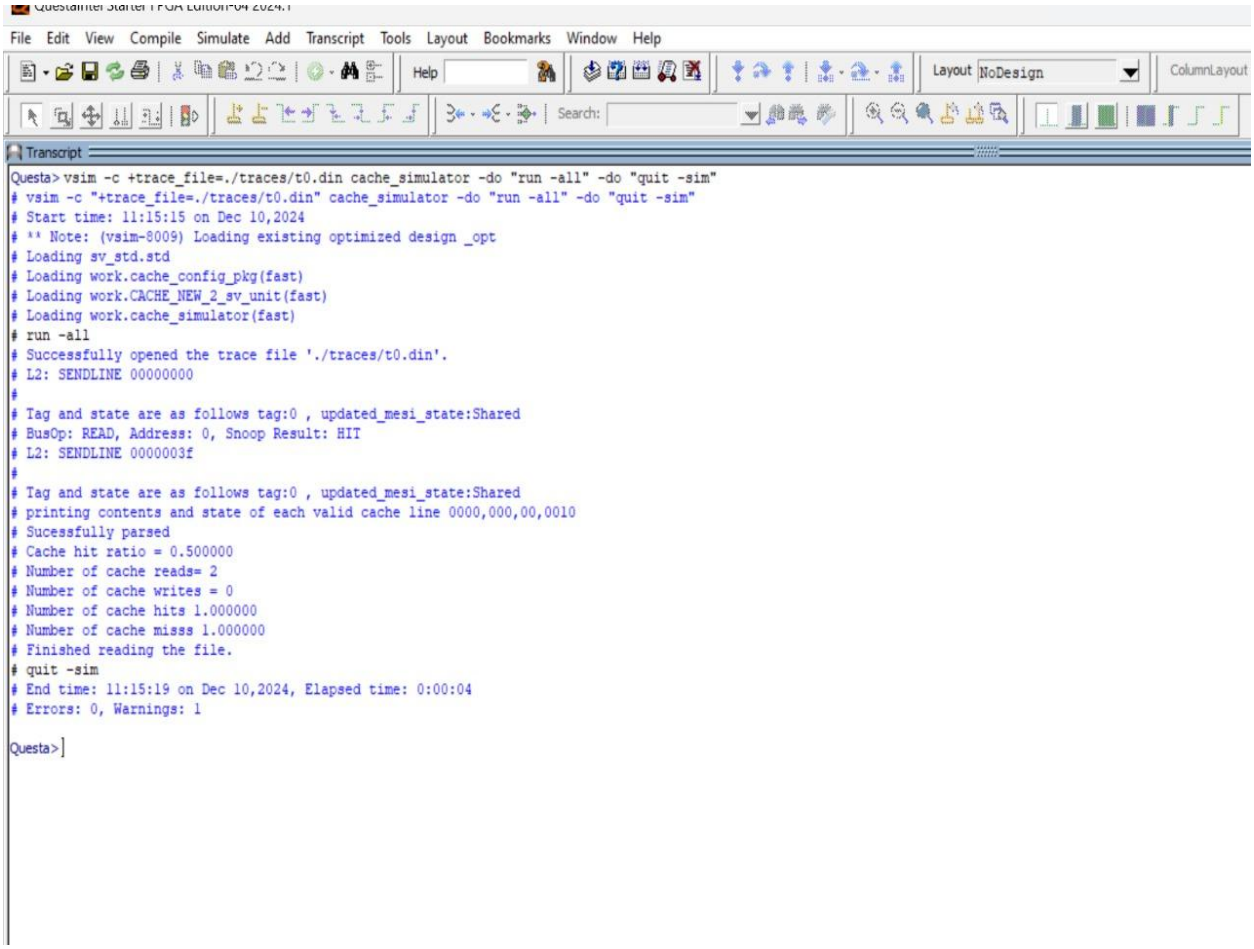
Number of cache misss 16.000000

Finished reading the file.

quit -sim

End time: 10:04:17 on Dec 10,2024, Elapsed time: 0:00:04

Errors: 0, Warnings: 1


File Edit View Compile Simulate Add Transcript Tools Layout Bookmarks Window Help
Help
Layout NoDesign
ColumnLayout
Search:
Transcript
Questa> vsim -c +trace_file=./traces/t0.din cache_simulator -do "run -all" -do "quit -sim"
vsim -c "+trace_file=./traces/t0.din" cache_simulator -do "run -all" -do "quit -sim"
Start time: 11:15:15 on Dec 10,2024
** Note: (vsim-8009) Loading existing optimized design _opt
Loading sv_std.std
Loading work.cache_config_pkg(fast)
Loading work.CACHE_NEW_2_sv_unit(fast)
Loading work.cache_simulator(fast)
run -all
Successfully opened the trace file './traces/t0.din'.
L2: SENDLINE 00000000
#
Tag and state are as follows tag:0 , updated_mesi_state:Shared
BusOp: READ, Address: 0, Snoop Result: HIT
L2: SENDLINE 0000003f
#
Tag and state are as follows tag:0 , updated_mesi_state:Shared
printing contents and state of each valid cache line 0000,000,00,0010
Sucessfully parsed
Cache hit ratio = 0.500000
Number of cache reads= 2
Number of cache writes = 0
Number of cache hits 1.000000
Number of cache misss 1.000000
Finished reading the file.
quit -sim
End time: 11:15:19 on Dec 10,2024, Elapsed time: 0:00:04
Errors: 0, Warnings: 1
Questa>