# Test Plan

**a) Testcase for Compulsory Miss:**

- Verifies the compulsory miss condition which happens at the initial read/write to Cache line.
- After identifying Compulsory miss occurs, Cache line is loaded with items from the next level hierarchy.

**b) Testcase for Write Hits:**

- The Data is written to the cache when the tag and index bits match the requested address resulting in Write Hit.
- Write Back Policy: The write back policy ensures data is written to DRAM only when necessary. A dirty bit in the tag array tracks modifications, and updates are written back to memory only when the cache line is evicted.

**c) Testcase for Write Miss:**

- While writing to Cache we stall the CPU and evict the line when there is a mismatch in valid and tag bits.
- Write Allocate Policy: On a write miss, the cache fetches the block from DRAM, allocates it, and writes the data in Cache while stalling the CPU.

**d) Testcase for Read Hits:**

- Perform a read to a cached address. Return data directly from cache if the valid bit is set and tag matches, without accessing memory.

**e) Testcase for Read Miss:**

- Perform a read to an address that is not in the cache.
- Cache miss triggers a memory fetch, stalls CPU, and loads the block into cache, evicting if necessary.

**f) Testcase Scenario for PLRU-**

- Initially write 16 blocks to a cache set and verify the eviction choice on the next write, ensuring the PLRU policy is followed.
- Test read patterns to show PLRU inaccuracies by observing if it fails to evict the true least recently used block.

**g) Coherence Testcase Scenario using MESI:**

- The MESI protocol maintains cache coherence by managing state transitions (Modified, Exclusive, Shared, Invalid) across L1 and L2 caches.
- L1 uses a write-once policy, while L2 ensures inclusivity and employs a write-allocate policy for memory updates.
- Reads and writes trigger state changes, with L1 invalidating or updating L2 and other cores as needed.

Pseudo-LRU replacement ensures efficient eviction, writing back modified data to L2 to maintain consistency.

# Technical Specification:

The last level cache (L2) design implemented in the project has the following specifications:

- Total Capacity = 16MB
- Byte Line = 64-byte lines
- Associativity = 16-way set
- Write hit policy = Write Back
- Write miss policy = Write Allocate
- Coherence Protocol = MESI
- Replacement Policy = pseudo-LRU scheme

# Data Structure:

The last level cache (L2) design implemented in the project has the following data structures:

- Address Bit: 32 bits
- Byte Offset Select: 6 bits
- Index Bits: 14 bits
- Tag Bits: 12 bits

| | | |
|---|---|---|
| **TAG BITS [31:20]** | **INDEX BITS [19:6]** | **BYTE OFFSET [5:0]** |

**TOTAL ADDRESS Bits [31:0]**

- Tag array consists of { Valid Bit , Dirty Bit , Tag Bits, MESI Bits } + 15 PLRU Bits per index.

| | | | |
|---|---|---|---|
| **Valid Bit [15]** | **Dirty Bit [14]** | **Tag Bits [13:2]** | **MESI Bits[1:0]** |

**TAG ARRAY [15:0]**