# Cache Test Plan

# Unit Test Plan

# pLRU Test Plan

1. pLRU initializes properly
   1. When CREATE is called, all pLRU bits are set to 1
2. On update, set TOGGLE to correct value until traverse complete
   1. 0 if traverse left
   2. 1 if traverse right
   3. End at given index
3. On evict, traverse properly
   1. Right if toggle 0
   2. Left if toggle 1
      1. Complement toggle while traversing
   3. Return proper index when traverse complete
4. On FLUSH
   1. Traverse array
      1. Set TOGGLE in all elements(call CREATE)

# Set Test Plan

1. Coherence???
2. Miss on first access
   1. Generate Read to DRAM
      1. Ensure way VALID bit set
      2. Ensure dirty bit set if WRITE
3. Hit if data present
   1. Tell pLRU to update on hit
   2. If WRITE, update DIRTY
      1. If DIRTY, do nothing
      2. If NOT DIRTY, set DIRTY
   3. NEVER DEASSERT DIRTY ON HIT
4. Miss if data not present
   1. Do NOT tell pLRU to evict if not full
      1. Generate pLRU update request
      2. Generate read to DRAM
      3. Update VALID bit
   2. Call pLRU evict if set is full
      1. If DIRTY, generate write to DRAM prior to eviction
5. Properly respond to FLUSH
   1. If DIRTY
      1. Generate write to DRAM
      2. Deassert DIRTY
      3. Deassert VALID
   2. If NOT DIRTY
      1. Deassert VALID
6. Placeholder for expanding Test Plan

# Environment Test Plan

# Integration Test Plan

# Trace Test Plan