**7-Bit LRU Cache Replacement Policy Implementation for Verilog**

For an 8-way associative cache:

Diagram

Description automatically generated

|  |  |
| --- | --- |
| Line LRU State Bits [6..0] | Replace data in this block: |
| xxx0x00 | Block 0 |
| xxx1x00 | Block 1 |
| xx0xx10 | Block 2 |
| xx1xx10 | Block 3 |
| x0xx0x1 | Block 4 |
| x1xx0x1 | Block 5 |
| 0xxx1x1 | Block 6 |
| 1xxx1x1 | Block 7 |

|  |  |
| --- | --- |
| If a read or write (load) access to a line in this block occurs | Set LRU State Bits [6..0] for the line to |
| Block 0 | \_ \_ \_ 1 \_ 1 1 |
| Block 1 | \_ \_ \_ 0 \_ 1 1 |
| Block 2 | \_ \_ 1 \_ \_ 0 1 |
| Block 3 | \_ \_ 0 \_ \_ 0 1 |
| Block 4 | \_ 1 \_ \_ 1 \_ 0 |
| Block 5 | \_ 0 \_ \_ 1 \_ 0 |
| Block 6 | 1 \_ \_ \_ 0 \_ 0 |
| Block 7 | 0 \_ \_ \_ 0 \_ 0 |

* For an 8-way associative cache, 7 LRU bits are needed. Initially, bits are at 7’b0.
* After a Ready or Write, LRU bits pointing to a certain block as defined by their LRU bit are reversed to prevent eviction of a new block.
* If LRU bit = 0, the left node is the LRU
* If LRU bit = 1, the right node is the LRU
* See M68kAssociativeCacheController\_Verilog.v for policy code implementation.

A screenshot of a computer

Description automatically generated with medium confidence

Text

Description automatically generated