

**Appendix E: An illustration of the Bit-Pseudo LRU Algorithm**

To implement the Bit-Pseudo LRU algorithm, we use a specific bit, named MRU, to record the usage status of each cache block. A block with MRU=1 indicates that the block is recently used. When a cache miss occurs, we replace the lowest indexed block of MRU=0 with new data and set MRU=1. If all blocks have MRU=1, then reset all MRUs to 0, except the one just replaced.

As an illustration, assume an 8-block example with its initial MRU values listed below:

Block index	0	1	2	3	4	5	6	7
MRU	1	0	1	1	0	0	0	0

When a cache miss occurs, the lowest indexed block with MRU=0 is block 1. Its MRU is set to 1.

Block index	0	1	2	3	4	5	6	7
MRU	1	1	1	1	0	0	0	0

If at a certain time point, there is one last block with MRU=0 as shown below.

Block index	0	1	2	3	4	5	6	7
MRU	1	1	1	1	0	1	1	1

Then if another cache miss occurs, the lowest indexed block with MRU=0 is block 4. Its MRU is set to 1.

Block index	0	1	2	3	4	5	6	7
MRU	1	1	1	1	1	1	1	1

However, now since every block has MRU=1, we reset all MRUs except the one just replaced.

Block index	0	1	2	3	4	5	6	7
MRU	0	0	0	0	1	0	0	0

Note:

- 1) The initial MRU contents are all set to 0.
- 2) When the set associativity of the cache is not fully-associative, the reset of MRUs is by the set which all blocks of MRU = 1.