Master of Computer Applications

CAPOL403R01: Computer Organization & Architecture

Unit III: Lecture 3
Cache mapping - Direct Mapping

Dr. D. MURALIDHARAN
Assistant Professor
School of Computing
SASTRA Deemed to be University

Agenda

- Direct mapping
- Numerical examples

Direct mapping

- Cache memory requirements
  - Number of lines should be in the power of 2 (i.e) $C = 2^L$
  - Number of words should be in the power of 2 (i.e) 2^W
- Main memory requirements
  - Words (bytes) per block should be in the power of 2

Direct mapping…

- The ‘i’th main memory block Bi will be mapped on the ‘j’th cache line Lj using the formula

  $j = i \bmod n$ where $n = 2^L$

- Question: On which cache line, the main memory block B255 will be mapped? Given: The number of cache lines are 16.

  $j = i \bmod n = 255 \bmod 16 = 15$

  B255 will be mapped on L15

Direct mapping…

- Address length = n = (t+l+w) bits
- Number of addressable units = 2^n words / bytes
- Block size = line size = 2^w words / bytes
- Number of blocks in main memory = 2^{n-w}
- Number of lines in cache = 2^l
- Size of cache = 2^{l+w}
- Width of tag = t = n-(l+w) bits

Direct mapping…

- Consider the following toy examples
- Total number of main memory blocks and cache lines are (i) 32 & 2 (ii) 32 & 4 (iii) 32 & 8
- The main memory blocks are mapped on cache lines as given below

Direct mapping…

Scenario (i)

Line	Block
0	0,2,4,8,10,12,14,16,18,20,22,24,26,28,30
1	1,3,5,7,9,11,13,15,17,19,21,23,25,27,29,31

Scenario (ii)

Line	Block
0	0,4,8,12,16,20,24,28
1	1,5,9,13,17,21,25,29
2	2,6,10,14,18,22,26,30
3	3,7,11,15,19,23,27,31

Scenario (iii)

Line	Block
0	0,8,16,24
1	1,9,17,25
2	2,10,18,26
3	3,11,19,27
4	4,12,20,28
5	5,13,21,29
6	6,14,22,30
7	7,15,23,31

Importance of tag

- Tag bits indicate the block number which is currently resides in the cache line
- For example, in scenario (ii), there are 8 different blocks are mapped on a single cache line
- Hence 3 tag bits are needed to differentiate the blocks

Line	Tag							
	000	001	010	011	100	101	110	111
00 (0)	0	4	8	12	16	20	24	28
01 (1)	1	5	9	13	17	21	25	29
10 (2)	2	6	10	14	18	22	26	30
11 (3)	3	7	11	15	19	23	27	31

Bit fields in direct mapping

- Memory address is divided into three parts

Tag	Line	Word

  - Number of bits for word field = $\log_2$(No.of.words/line)
  - Number of bits for line field = $\log_2$(No.of.cache lines)
  - Tag bit size = memory address bits – (l+w)

    = Bits per block – bits per line (from previous example)

- *Note1 : No. of. Words per line = No. of words per Block*
- *Note2 : Total number of blocks = 2^{T+L}*

Bit fields in direct mapping

- Consider 256 addressable words are available in main memory
  - 8 bits are needed to identify one address

- Consider scenario (ii)
  - Main memory is divided into 32 blocks
  - Hence, 8 addressable words are available per block
  - Three bits are needed to identify one addressable word

Bit fields in direct mapping

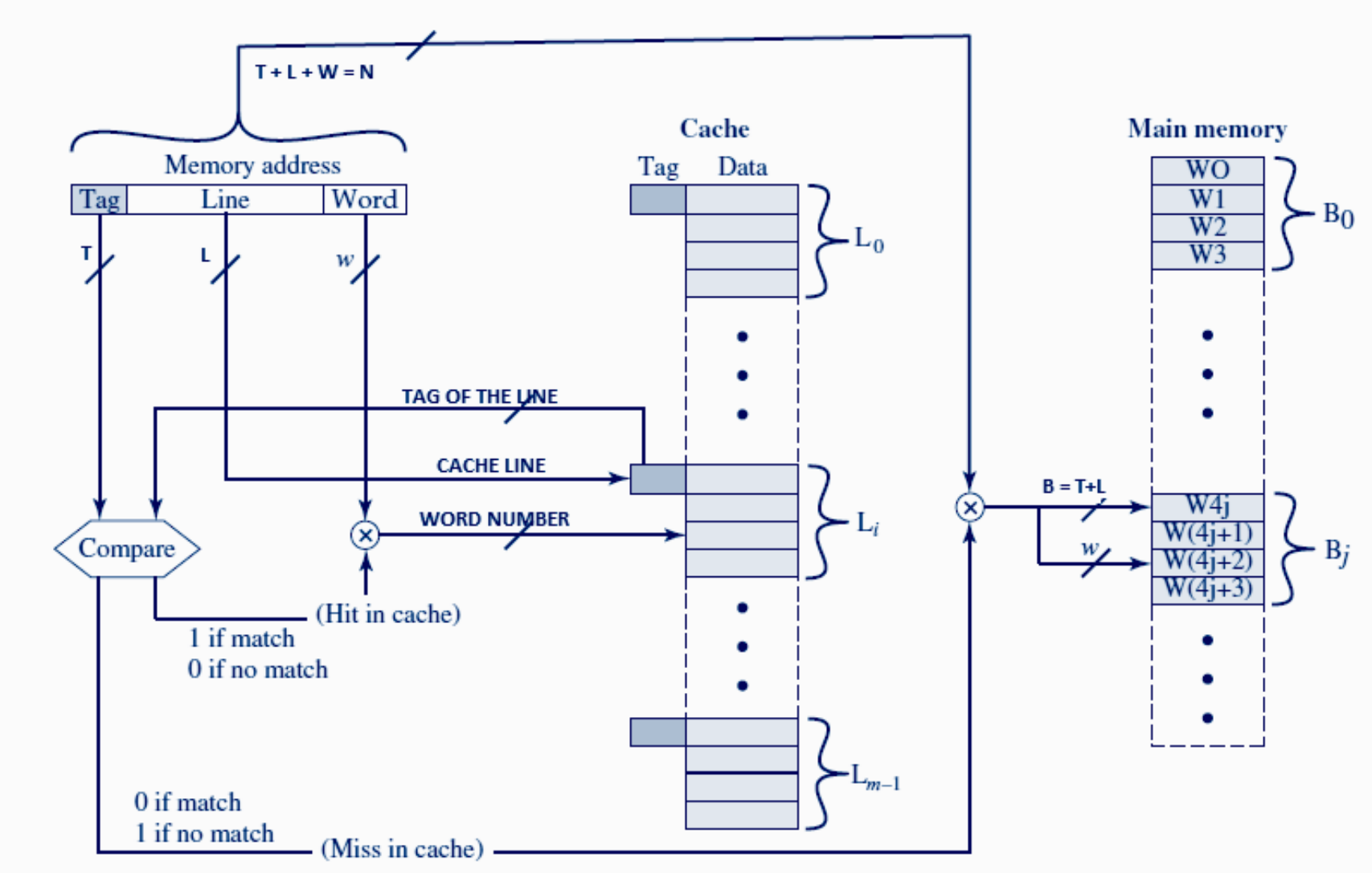
- As 4 lines are available, 2 bits are needed to differentiate a line
- Hence, tag bits are 8-(2+3) = 3
- The 8 bit address is split as follows

Tag (7,6,5)	Line (4,3)	Word(2,1,0)

- Consider the addresses “100_01_000” (B17) and “101_01_000” (B25)
  - Both addresses will be mapped on the first word of (000) line 1
  - Cache hit / miss is decided by tag bit

Tag	Line	W0	W1	W2	W3	W4	W5	W6	W7
100	01	000	001	010	011	100	101	110	111

Memory access flow



Courtesy: Computer organization and architecture: designing for performance, William Stallings, Eighth edition

Problems

1. On which cache line address 137 will be available?
2. Processor wants to read the content of address 234. State the condition for cache hit.
3. Suppose a cache miss occurs when processor tries to access the address 47. Find the addresses which are moved to cache due to this cache miss?

Problems…

4. A cache can hold 64K Bytes. Data is transferred between the cache and the main memory in blocks of 4 bytes each. Main memory consists of 64M Bytes.
   - How many blocks are in main memory?
   - How many lines are in cache?
   - How many bits are allocated for tag?
   - What are the lower and higher values of tag in HEX?
   - Find the starting addresses of blocks which are assigned in line number 0, 2, 4 and last line of the cache.
5. State the principal merit and demerit of direct mapping technique

Answers

1. The binary equivalent of 137 is 10001001 which is organized as below:

Tag	Line	Word
100	01	001

So, the address 137 is mapped on L1

Answers

2. The binary equivalent of 234 is 11101010 which is organized as below:

Tag	Line	Word
111	01	010

The expected condition for cache hit is: The tag bits of L1 should be “111”.

If any other combinations from “000” to “110” is available, then it is cache miss; 234 is not on cache!

Answers

3. The binary equivalent of 47 is 00101111 which is organized as below:

Tag	Line	Word
001	01	111

So, the block “00101” is moved to cache line 01. In other words the addresses 00101000 to 00101111 will be moved to L1. (In decimal, the addresses 40 to 47 will be moved to cache line L1)

Answers

4. (i) As one block has 4 bytes, 16 M (=64M/4) blocks are available in main memory

(ii) As one line has 4 bytes, 16 K lines are available in cache memory

(iii) No.of.bits allocated for tag = n – (l+w)

$n=\log_2(64M)=26$ bits; $l=\log_2(16K)=14$ bits w= 2 bits

So, t=26-16= 10 bits

(iv)The range of tag bits in hex: 000H to 3FFH

(00_0000_0000B=000H&11_1111_1111B=3FFH)

Recapitulation

- ✓ Direct mapping technique
- ✓ Starting addresses of blocks which could be mapped on a particular line
- ✓ All the addresses which could be mapped on a particular line
- ✓ Principal advantage and disadvantage of cache mapping

Thank you