Webinar-2
Cache Memory

by

Dr. Selva kumar S

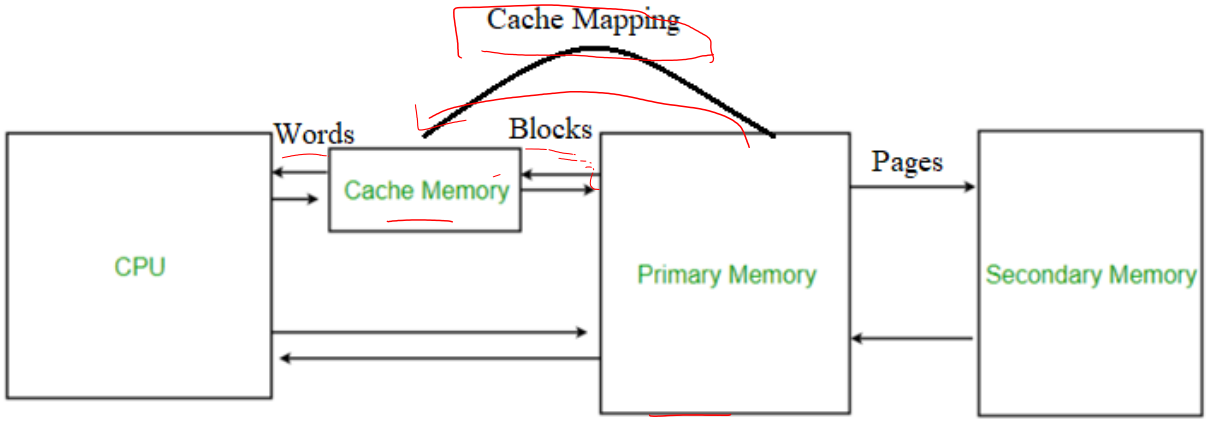
s.selva@wilp.bits-pilani.ac.in

BITS Pilani

Pilani Campus

Cache Mapping- Summary

- The term "cache mapping" describes a method of bringing the information in the main memory into the cache.



- Cache Performance (Hit Ratio)
  - Cache Hit
  - Cache Miss
- Cache Mapping Techniques:
  - Direct Mapping
  - Fully Associative Mapping
  - Set Associative Mapping

2^0 : 1
2^1 : 2
2^2 : 4
2^3 : 8
2^4 : 16
2^5 : 32
2^6 : 64
2^7 : 128
2^8 : 256
2^9 : 512
2^{10} : 1024 or 1K

2^{10} : 1 K
2^{11} : 2 K
2^{12} : 4 K
2^{13} : 8 K
2^{14} : 16 K
2^{15} : 32 K
2^{16} : 64 K
2^{17} : 128 K
2^{18} : 256 K
2^{19} : 512 K
2^{20} : 1024 K or 1M

2^{20} : 1 M
2^{21} : 2 M
2^{22} : 4 M
2^{23} : 8 M
2^{24} : 16 M
2^{25} : 32 M
2^{26} : 64 M
2^{27} : 128 M
2^{28} : 256 M
2^{29} : 512 M
2^{30} : 1024 M or 1G

Direct Mapping

BITS Pilani
Pilani Campus

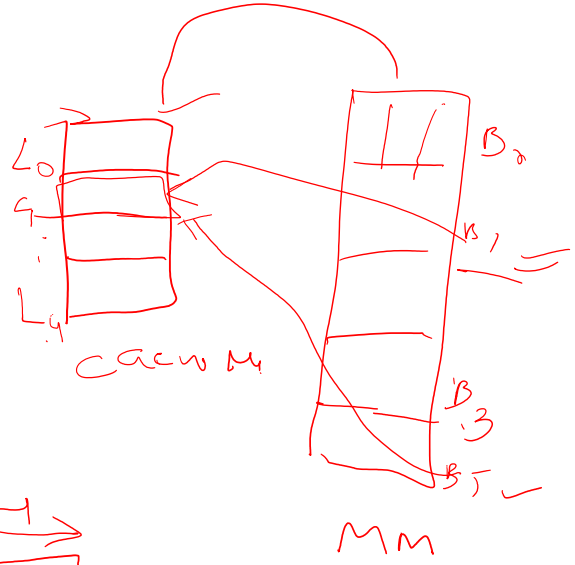
Direct Mapped Cache - Summary

- Each block of main memory maps to only one cache line
  - **if a block is in cache, it must be in one specific place**
  - **i = j modulo m**

  where i = cache line number

  j = main memory block no.

  m = no.of lines in the cache

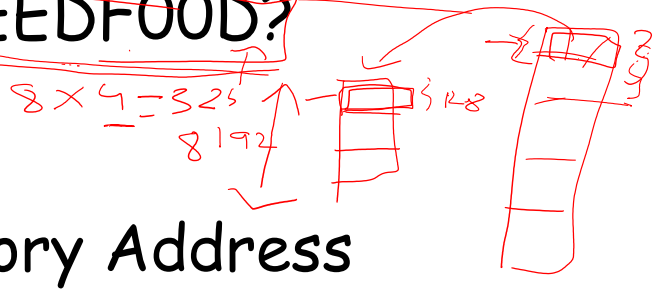


- Address is split in three parts:
  - Tag
  - Line
  - Word

decimal	*binary*	*hex*
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Problem- 1: Direct mapping

Consider a cache memory of 8192 KB with line size of 128 B. Find the number of bits required for TAG, LINE and WORD fields of the main memory address 0xFEEDF00D?



Solution-1:

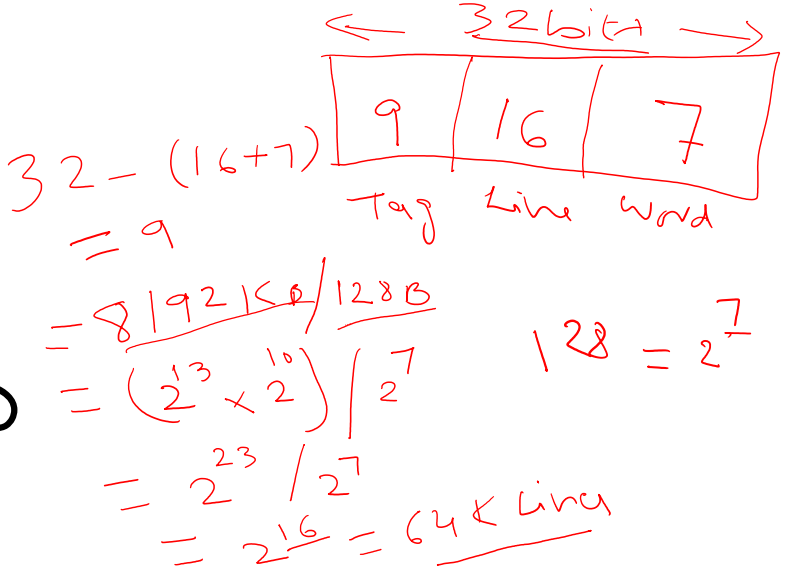
- No. of bits needed for the Main Memory Address

  = 32 bits
- Block Size

  = 128Bytes

- No. of bits to represent WORD

  128 Bytes = 2^7 => 7 bits



Problem- 1: Direct mapping

innovate achieve lead

Consider a cache memory of 8192 KB with line size of 128 B. What is the tag, line and word for the main memory address 0xFEEDF00D?

Solution-1

- Cache Size = 8192 KB
- Total no. of Cache Lines
  = Cache size / Line size
  = 8192 KB/128B = $(2^{13}.2^{10}) / 2^{7} = 2^{16}$ = 64K lines

- Total bits to represent line field
  = 16 bits

- Total bits to represent Tag filed is:
  = 32 – Line – Word = 9 bits

Problem- 1: Direct mapping

Given Address : 0xFEEDF00D

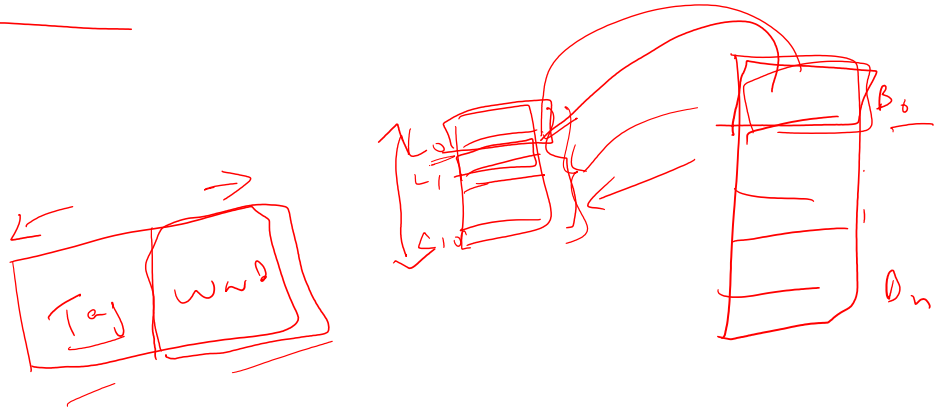
F				E				E				D				F				0				0				D			
1	1	1	1	1	1	1	0	1	1	1	0	1	1	0	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	0	1
Tag bits (9)									**Line bits (16)**															**Block offset (7)**							

Associative Mapping

BITS Pilani
Pilani Campus

Associative Mapping - Summary

- A main memory block can load into any line of cache
- Memory address is interpreted as tag and word
- Tag uniquely identifies block of memory
- Every line's tag is examined for a match
- Cache searching gets expensive



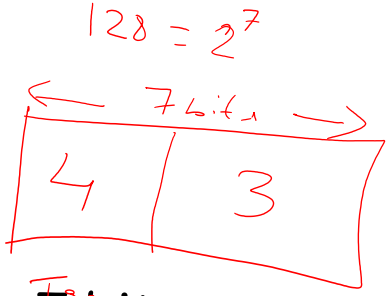
Problem 2: Fully Associative

A system has main memory of size 128 Byte with on chip cache 32 Bytes and block size of 8 Bytes. The system uses fully associative cache mapping. Find the following:

a) The number of main memory address bits.

Main Memory Size = 128 Bytes = 2^7 Bytes

Thus, the number of bits needed for Physical Address = 7 bits.

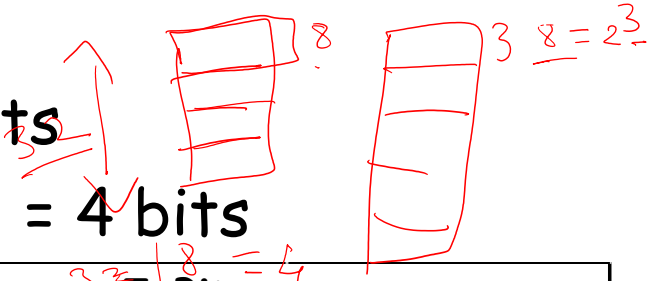


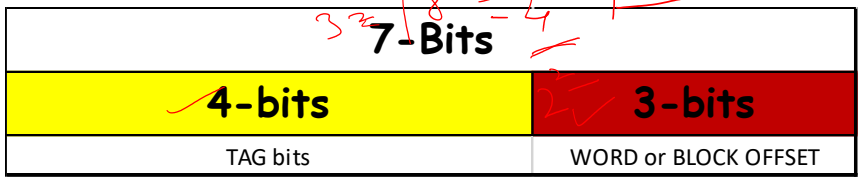
b) The number of Tag bits and Word bits.

Block size = 8 Bytes = 2^3 Bytes

# of bits for WORD offset field = 3 bits

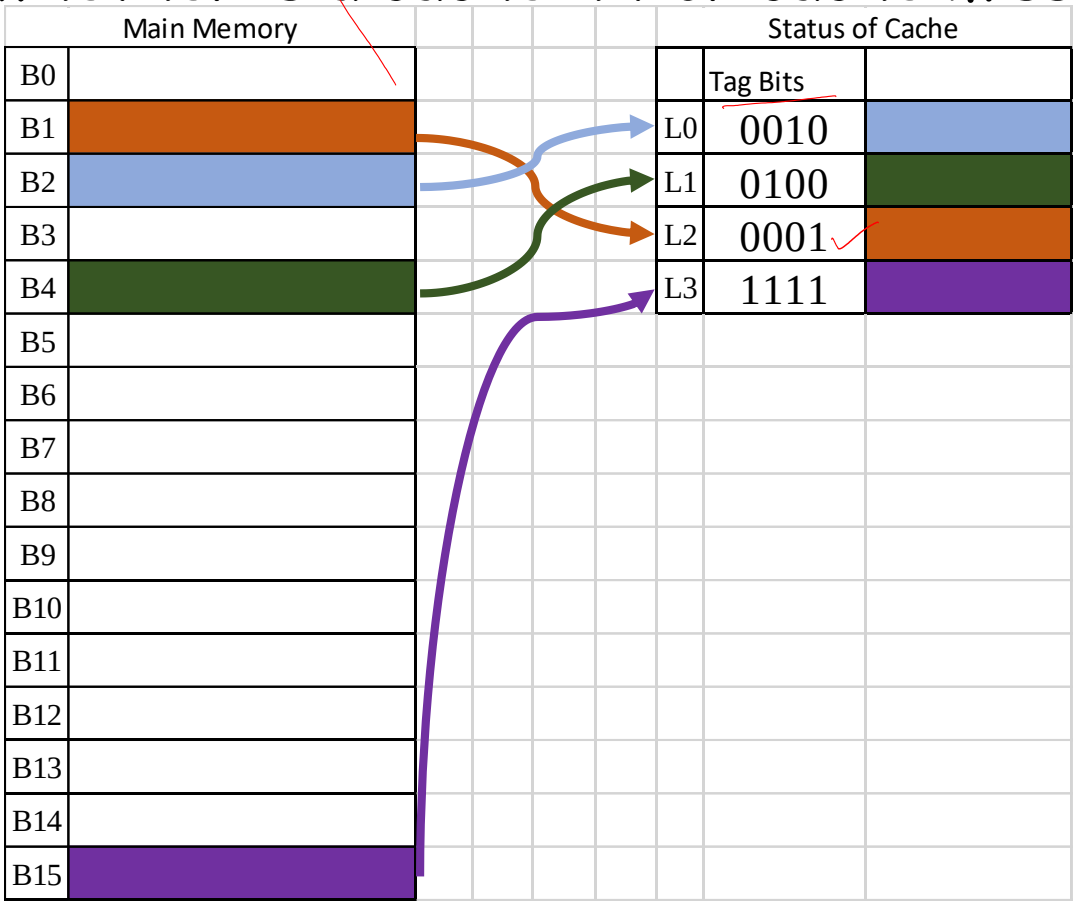
# of Tag bits = 7 – WORD offset = 7-3 = 4 bits





Fully Associative

c) If the CPU request the addresses 0001100, 0011001 find whether is a cache hit or cache miss for each of the address.



0001100

0001100 - Hit

0011001

0011001-Miss

Problem 3: Fully Associative

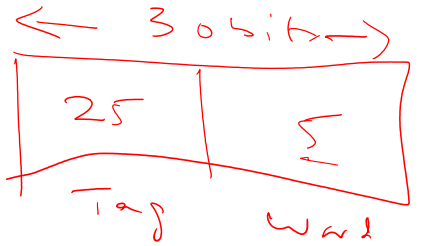
An 8KB associative cache has a line size of 32 Bytes. Main memory size is 1GB. Find the number of TAG bits and number of comparators required for search.

Given Data:

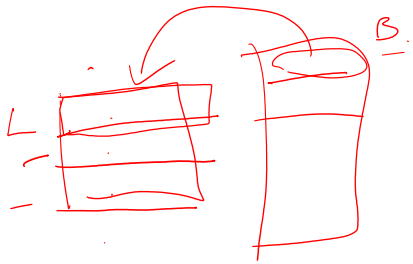
Cache size = 8KB = $2^3.2^{10}$ = 2^{13} Bytes

Block Size = 32 Bytes = 2^5 Bytes

Memory size = 1GB = 2^{30} Bytes



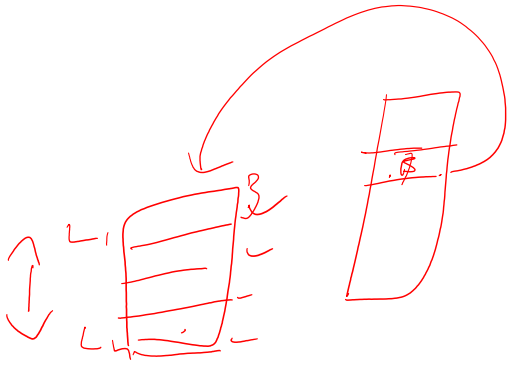
- # of bits for main memory address = 30 bits
- # of bits for Word offset field = 5 bits
- # of bits for Tag field = 30 – 5 = 25 Bits



30 bits	
25	5
Tag	Word

Problem 3: Fully Associative

- # of cache lines
  = (Cache Size)/(Line Size) = $2^{13}/2^5 = 2^8$ lines = 256 lines
- # of Comparators required = # of Cache lines = 256
- Size of comparator = size of tag bits
  = 25-bit comparator



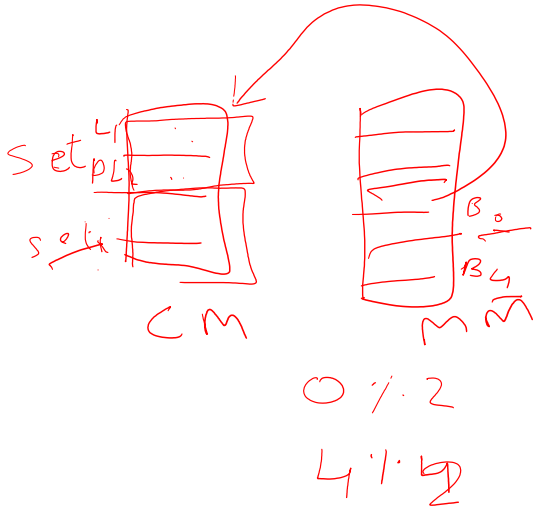
Set Associative - Summary

- Makes use of advantages of Direct Mapped and Associative mapping
  - Block is mapped to a set (Direct mapping)
  - Within the block can be placed in any line (Associative)
- K-Way associative cache ➔ K lines in a set
- Mapping function : i= j % v

Where i = cache set number

j = main memory block number

v = number of sets in the cache

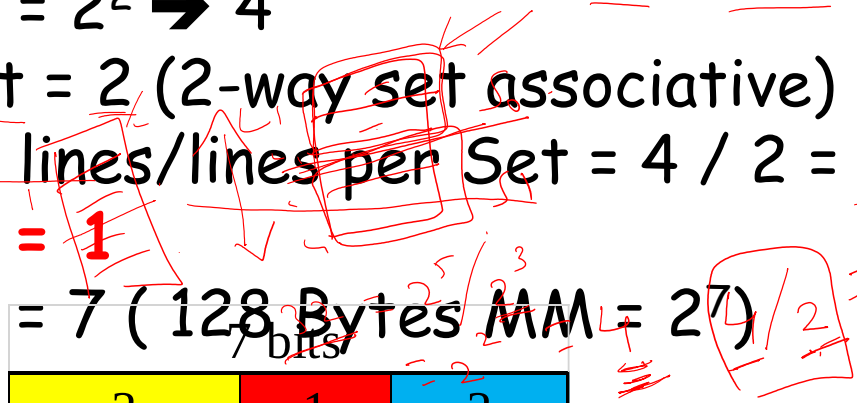


Problem : Set Associative

innovate achieve lead

A system has a main memory of 128 Bytes and a cache size of 32 Bytes with 8 Bytes per cache block. Assume that the size of each memory word is 1 byte. if the cache is organized as a 2-way set-associative cache. Find out TAG, SET, and WORD field bits.

- Block size = 8 bytes = 2^3 Bytes ➔ **WORD = 3bits**
- Cache size = 32 Bytes = 2^5 Bytes
- Number of lines = Cache size / Block size
  $= 2^5/2^3$ Bytes = 2^2 ➔ 4
- Number of cache lines per set = 2 (2-way set associative)
- Number of Sets = Number of lines/lines per Set = 4 / 2 = 2
- # bits for identifying a **SET = 1**
- Total number of address bits = 7 (128 Bytes MM = 2^7)
- **TAG = 7-(1+3)=3bits**

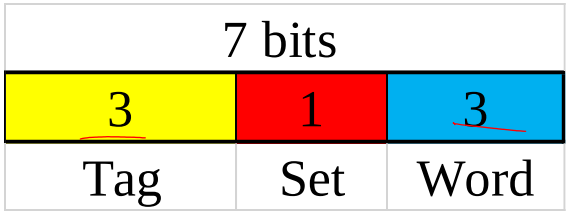


Problem : Set Associative

b) When a program is executed, the processor requests data from the following word addresses:

0001010 , 0010010

For each of the above addresses, find out for which set it will map.



- For the word address: 0001010

  000 1 010 → Set-1

- For the word address: 0010010

  000 0 010 → Set-0

Home work

A set associative cache memory consists of 128 lines divided into four line sets. The main memory consists of 16,384 blocks and each block contains 256 eight bit words.

a) How many bits are required for addressing the main memory?

b) How many bits are needed to represent the TAG, SET and WORD fields?

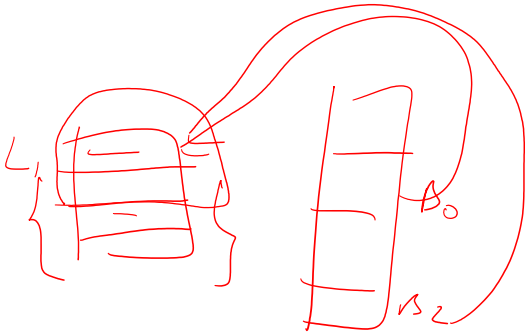
Replacement Algorithms

BITS Pilani
Pilani Campus

Replacement Algorithms

innovate achieve lead

- Needed in Associative & Set Associative mapped cache
- Methods:
  - Least Recently Used (LRU)
  - Least Frequently Used (LFU)
  - First In First Out (FIFO)
  - Random



Replacement Algorithms

- Least Recently used (LRU): Replace the block in the set that has been in the cache longest with no reference to it
- Least frequently used: Replace block which has had fewest hits
  - Uses counter with each line
- First in first out (FIFO): Replace block that has been in cache longest
  - Round robin or circular buffer technique
- Random

Problem

Consider a reference pattern that accesses the sequence of blocks 4, 7, 6, 1, 7, 6, 1, 2, 7, 2, 5, 4 Assuming that the cache uses associative mapping, find the hit ratio with a cache of four lines

a) LRU
b) LFU
c) FIFO

LRU

Ref	4	7	6	1	7	6	1	2	7	2	5
time	0	1	2	3	4	5	6	7	8	9	10
L0	4_0	4_0	4_0	4_0	4_0	4_0	4_0	2_7	2_7	2_9	2_9
L1		7_1	7_1	7_1	7_4	7_4	7_4	7_4	7_8	7_8	7_8
L2			6_2	6_2	6_2	6_5	6_5	6_5	6_5	6_5	5_{10}
L3				1_3	1_3	1_3	1_6	1_6	1_6	1_6	1_6
H/M	M	M	M	M	H	H	H	M	H	H	M

cache miss

cache hit = 5/11

LFU

Ref	4	7	6	1	7	6	1	2	7	2	5
L0	4_1	4_1	4_1	4_1	4_1	4_1	4_1	2_1	2_1	2_2	2_2
L1		7_1	7_1	7_1	7_2	7_2	7_2	7_2	7_3	7_3	7_3
L2			6_1	6_1	6_1	6_2	6_2	6_2	6_2	6_2	5_1
L3				1_1	1_1	1_1	1_2	1_2	1_2	1_2	1_2
H/M	M	M	M	M	H	H	H	M	H	H	M

FIFO

Ref	4	7	6	1	7	6	1	2	7	2	5
time	0	1	2	3	4	5	6	7	8	9	10
L0	4_0	4_0	4_0	4_0	4_0	4_0	~~4_0~~	2_7	2_7	2_7	2_7
L1		7_1	7_1	7_1	7_1	7_1	7_1	7_1	7_1	~~7_1~~	5_{10}
L2			6_2	6_2	6_2	6_2	6_2	6_2	6_2	6_2	6_2
L3				1_3	1_3	1_3	1_3	1_3	1_3	1_3	1_3
H/M	M	M	M	M	H	H	H	M	H	H	M

Problem : Set Associative (LRU)

Consider a 2 – way set associative cache with 4 cache lines (0-3). The memory block requests are in the order-

4, 3, 25, 8, 4, 6, 25, 8, 16, 35, 4

If LRU replacement policy is used, calculate the hit ratio and miss ratio

Problem : Set Associative (LRU)

		4	3	25	8	4	6	25	8	16	35	4
	Time	0	1	2	3	4	5	6	7	8	9	10
Set 0	L0											
	L1											
Set 1	L2											
	L3											
	H/M											

Problem : Set Associative (LRU)

innovate achieve lead

BN (j)	4	3	25	8	4	6	25	8	16	35	4
#of Sets(v	2	2	2	2	2	2	2	2	2	2	2
i=j%v	0	1	1	0	0	0	1	0	0	1	0

		4	3	25	8	4	6	25	8	16	35	4
	Time	0	1	2	3	4	5	6	7	8	9	10
Set 0	L0	4_0	4_0	4_0	4_0	4_4	4_4	4_4	8_7	8_7	8_7	4_{10}
	L1				8_3	8_3	6_5	6_5	6_5	16_8	16_8	16_8
Set 1	L2		3_1	3_1	3_1	3_1	3_1	3_1	3_1	3_1	35_9	35_9
	L3			25_2	25_2	25_2	25_2	25_6	25_6	25_6	25_6	25_6
H/M		M	M	M	M	H	M	H	M	M	M	M

Problem : Set Associative (FIFO)

Consider a 2 – way set associative cache with 4 cache lines (0-3). The memory block requests are in the order-

4, 3, 25, 8, 4, 6, 25, 8, 16, 35, 4

If FIFO replacement policy is used, calculate the hit ratio and miss ratio

Problem : Set Associative (FIFO)

		4	3	25	8	4	6	25	8	16	35	4
	Time	0	1	2	3	4	5	6	7	8	9	10
Set 0	L0											
	L1											
Set 1	L2											
	L3											
	H/M											

innovate achieve lead

Problem : Set Associative (FIFO)

BN (j)	4	3	25	8	4	6	25	8	16	35	4
#of Sets(v)	2	2	2	2	2	2	2	2	2	2	2
i=j%v	0	1	1	0	0	0	1	0	0	1	0

		4	3	25	8	4	6	25	8	16	35	4
	Time	0	1	2	3	4	5	6	7	8	9	10
Set 0	L0	4_0	4_0	4_0	4_0	4_0	6_5	6_5	6_5	6_5	6_5	4_{10}
	L1				8_3	8_3	8_3	8_3	8_3	16_8	16_8	16_8
Set 1	L2		3_1	3_1	3_1	3_1	3_1	3_1	3_1	3_1	35_9	35_9
	L3			25_2	25_2	25_2	25_2	25_2	25_2	25_2	25_2	25_2
	H/M	M	M	M	M	H	M	H	H	M	M	M

Home work Problem :

A Research center wants to check whether implementing cache replacement using two existing cache memory replacement algorithms LFU and FIFO would help reduce the miss rate. The proposed algorithm would work in two phases. The first 6 clocks (0-5) follow LFU and the next 6 clocks (6-11) follow FIFO. The main memory block sequence is 0, 4, 0, 2, 1, 5, 0, 1, 2, 5, 0, 2

What is the Hit Ratio for the proposed new replacement algorithm?

Justify your answer by filling in the following table. In LFU, in case of a tie between cache lines for replacement, select the line which has been there for a longer time in the cache.

Problem : Set Associative (FIFO)

Ref	0	4	0	2	1	5	0	1	2	5	0	2
time	0	1	2	3	4	5	6	7	8	9	10	11
L0												
L1												
L2												
L3												
H/M												

innovate achieve lead

Thank You