

Name: Sadia Afran Tamanna

ID: 1812030042

Answer to the question no - 1

Given that,

RAM size = 64 KB

Block size = 4

Cache size = 128 bytes

$K = 2$

~~total number of sets~~

total number of lines = $\frac{128}{8 \times 4} = 16$ 32

" " " set, $S = \frac{16}{2} = 8$

total number of lines, $= \frac{128}{16} = 8$ 32

" " " set, $S = \frac{32}{2} = 16$

memory address	Block of RAM	Hit/miss	Set no	Line no	consequence
3	0	miss	$s = 0 \bmod 8$ $= 0$	1	Block 0 move to Line-1 of set = 0
5	1	miss	$s = 1 \bmod 8$ $= 1$	1	Block -1 move to set = 1
10	2	miss	$s = 2 \bmod 8$ $= 2$	1	Block -2 move to line - 1 of set = 2
65	16	miss	$s = 16 \bmod 8$ $= 0$	2	Block -16 move to line - 2 of set = 0
66	16	Hit	$s = 16 \bmod 8$ $= 0$	2	read from cache.
129	32	miss	$s = 32 \bmod 8$ $= 0$	1	Block = 32 will replace Block = 0 in Line-1 of set = 0
130	32	Hit	$s = 32 \bmod 8$ $= 0$	1	read from cache.
69	17	miss	$s = 27 \bmod 8$ $= 1$	2	Block = 17 move to Line = 2 of set = 1

7	1	Hit	$S = 1 \bmod 16$ = 1	2 2 1	Read from cache
133	33	miss	$S = 33 \bmod 16$ = 1	2	Block = 33 will replace Block - 17 in Line: 2 set: 1
72	18	miss	$S = 18 \bmod 16$ = 2	2	Block = 18 move to Line = 2 of set = 2
7h	18	Hit	$S = 18 \bmod 16$ = 2	2	Read from cache
75	18	Hit	$S = 18 \bmod 16$ = 2	2	Read from cache
11	2	Hit	$S = 2 \bmod 16$ = 2	1	Read from cache
137	34	miss	$S = 34 \bmod 16$ = 2	2	Block = 34 will replace Block = 18 in Line = 2 of set = 2
1024	256	miss	$S = 256 \bmod 16$ = 0	2	Block = 256 replace Block = 18 in Line = 2 of set = 0

$$\text{Hit ratio} = \frac{6}{16} \times 100\% = 37.5\%$$

Block replaced = 4

point to be noted, Here we used "Least

recently used (LRU)" as replacement
algorithm to replace block in line of cache.

(Answer)

Answer to the question no - 2

(a)

In this problem of above, if we use the 4-way set associative mapping then there would shows some merits and demerits as the given below,

Demerits :

(i) If we increase the value of K so the line number in every set will increase so it will take more time to match the tag of memory with a tag of cache.

(ii) There will be lower set number than before. So, may be the large number of instruction will be problem.

Merits :

(i) Since, there we have more lines in every set, so we have to use less replacement in the cache.

(ii) Hit ratio will be greater than before as we can store more instruction in sets with less replacement.

Now, ~~Avg~~ Average access time

if the hit ratio is greater then the average access time will be lower,

So, we can say, hit ratio $\propto \frac{1}{\text{Avg access time}}$

Again, the 4-way has more hit ratio than the before, so it will take less ~~average~~ avg. access time than 2-way.

So, Avg access time of 2way $>$ Avg access time of 4 way.

(b)

Merits :-

- (i) Here, there is no restriction to access cache in full associative mapping firstly, so the more instructions can be stored in cache.
- (ii) Cache will be used more than before.
- (iii) The hit ratio will be greater than before.

Demerits :-

- (i) This will take more time to check the tag of RAM with tag of each than the before.
- (ii) We can't decide earlier where a instruction will be stored in cache.

Now, Avg. Access Time

Mentioned in part a,

$$\text{hit ratio} \propto \frac{1}{\text{Avg. Access time}}$$

In the full associative mapping hit ratio will be greater than before for the given problem So, the avg access time will much lower than the before for instruction per data for the given problem.

(Answer)

Answer to the question no - q3

Data bit = 8 bit

Parity bit = 4 bit

Bit position	12	11	10	9	8	7	6	5	4	3	2	1
Position number	1100	1011	1010	1001	1000	0111	0110	0101	0100	0011	0010	0001
Data bit and check bit	D8	D7	D6	D5	C8	D4	D3	D2	C4	D1	C2	G
word read out	1	0	0	0	1	0	1	1	1	1	0	1

Data read : 10000111

Error code read : 1101

Generating error code :

$$C_1 = D_1 \oplus D_2 \oplus D_4 \oplus D_5 \oplus D_7 \\ = 1 \oplus 1 \oplus 0 \oplus 0 \oplus 0 = 0$$

$$C_2 = D_1 \oplus D_3 \oplus D_4 \oplus D_6 \oplus D_7 \\ = 1 \oplus 1 \oplus 0 \oplus 0 \oplus 0 = 0$$

$$C_3 = D_2 \oplus D_3 \oplus D_4 \oplus D_8 \\ = 1 \oplus 1 \oplus 0 \oplus 1 = 1$$

$$C_4 = D_5 \oplus D_6 \oplus D_7 \oplus D_8 \\ = 0 \oplus 0 \oplus 0 \oplus 1 = 1$$

Error code = 1100

$$\begin{array}{r} \text{XOR} & 1101 \\ & 1100 \\ \hline & 0001 \end{array}$$

Error is in 1st position
Here no data is error but error
in parity bit.

Answer to the question no - 4

(a) Given,

$$-0.0000125$$

Hence,

$$(0.0000125)_{10} = (0.00000000000000001101)_2$$

to make it normalised form.

$$= 1.101 \times 2^{-17}$$

$$\text{Now, exponent } = -17 + 127 = \del{114} 110$$

$$\del{114} \text{ in Binary} = 10010000$$

$$110 \text{ in Binary} = 0110110$$

$$\text{significant} = (10100\ldots)_2$$

sign bit = 1 as the decimal number

IEEE - 32 bit format is negative.

1	01101110	1010000000000000000000000
sign bit (1 bit)	Exponent (8 bit)	significant (23 bit)

(b)

Hence, sign bit = 1 so number is negative

$$(00001110)_2 = (14)$$

$$(00001110)_2 = (14)_{10}$$

$$\text{So, } 14 - 127 = -113 \quad (\text{exponent})$$

$$\text{significant} = (1.00010010001000\dots)_2$$

$$= (1.07080078125)_{10}$$

~~Answer~~

The decimal form,

$$-1.07080078125 \times 10^{-13}$$

$$= -1.031143645 \times 10^{-34}$$

(Answer)