Yousef Jarrar

CSE 401 – Homework 5

Dr. Gomez

5.3 For direct mapped cache design with a 32-bit address, the following bits of the address are used to access the cache

Tag	Index	Offset
31–10	9–5	4–0

1) what is the cache block size (in words)?

It is the basic unit for the cache storage that contains the data in the form of words and/or bites. To calculate the size of cache following formulas used:

Cache block size = $2^{OFFSETBITS}$

To calculate the cache block size for the direct mapped cache design with the 32-bit address and the problem the given data in tabular form shown below:

Tag	Index	Offset
31 – 10	9 – 5	4 – 0

Using the given information in the table above cache block size can be calculated as following:

the offset bits range is from 0 to 4. Hence, five bits for the offset.

Cache block size = $2^{offsetbits}$

= 2^5 bytes

Cache Block Size = 2^5 bytes

= 2^3 words

= 8 words

thus, the cache size of the block is eight words long.

2) how many entries as a cash have?

Using the same table from the last problem. To calculate the entries of cache for the direct mapped cache design with 32-bit address in the problem.

The index bits are from 5 – 9. Which means that the total bids for index is five

number of entries = $2^{\text{index bits}}$

= 2^5

= 32

there are 32 entries in the cached listed above

3) what is the ratio between total bits required for such a cache implementation over the data storage bits?

Consider the following bits of the address:

Tag	Index	Offset
31 – 10	9 – 5	4 – 0

The total number of bit and data storage bits formulas in a direct-mapped cache are,

Total bits= 2^{index} x (block size x tag size x valid field size)

Data storage bits – 2^{index} (block size)

The ratio b tween total bits and data storage bits for a cache implementation is

Ratio= Total bits / Data storage bits

Calculate the total number of bits:

$2^{\text{index}} = 2^5$

= 2^5

= 32 bytes

= 8 x 32 bits

= 256 bits

so the block sizes 256 bits

tag size equals two (31-10) bits

= 22 bits

since the tag size is 22 bids, block sizes 256 bits, and one bit is required for the valid field.

Total bids = 232 bits × (256 bits +22 bits +1 bit)

= 8928 bits

Calculate the data storage bits:

data storage bits = 2^{index} × (block size).

= 32 bits × (256 bits)

= 8192 bits

Ratio = 8928÷8001 92

approximated = 1.089

4) how many blocks are replaced?

The following table displays the replacement of the cache block that occurs based on the bits of addresses given below:

0	4	16	132	232	160	1024	30	140	3100	180	2180

The 16 MSB bits from 32-bits binary address of the above byte addressed cache reference is

0000 0000 0000 0000.

Decimal Address	**Binary Address (16 LSB bits of 32 bits)**	**Tag**	**Index**	**Hit/Miss**	**Replacement**
0	0000 0000 0000 0000	0	0	Miss	-
4	0000 0000 0000 0100	0	0	Hit	-
16	0000 0000 0001 0000	0	0	Hit	-
132	0000 0000 1000 0100	0	4	Miss	-
232	0000 0000 1110 1000	0	7	Miss	-

160	0000 0000 1010 0000	0	5	Miss	-
1024	0000 0100 0000 0000	1	0	Miss	Yes
30	0000 0000 0001 1110	0	0	Miss	Yes
140	0000 0000 1000 1100	0	4	Hit	-
3100	0000 1100 0001 1100	3	0	Miss	Yes
180	0000 0000 1011 0100	0	5	Hit	-
2180	0000 1000 1000 0100	2	4	Miss	Yes

5) how many entries does cache have?

Since, the offset is represented using lower 5 bits (4 – 0). So, the total number of bytes per block are 32 bytes.

The next five bits (9 – 5) are used for index. Therefore, total number of blocks are $2^5 = 32$.

To find the block number, the following formula can be used:

(Block Address) modulo (Number of blocks in the cache).

To compute the Block Address, the following formula can be used:

Byte Address / Bytes per block.

Hence, to find the block number using the byte address, the following formula can be used:

(Byte Address / Bytes per block) modulo (Number of blocks in the cache).

Using the above formula, each of the byte-addresses given in the question can be mapped as follows:

Byte-Address = 0,

$$\text{Block Number} = \left(\frac{0}{32}\right)\%32$$
$$= 0$$

Byte-Address = 4,

$$\text{Block Number} = \left(\frac{4}{32}\right)\%32$$
$$= 0$$

Byte-Address = 16,

$$\text{Block Number} = \left(\frac{16}{32}\right)\%32$$
$$= 0$$

Byte-Address = 132,

$$\text{Block Number} = \left(\frac{132}{32}\right)\%32$$
$$= 4\%32$$
$$= 4$$

For Byte-Address = 1024,

The number of blocks per tag is 32 and number of bytes per block is 32.

Hence, the number of bytes associated to a given tag value is 1024 (0 - 1023). So, for this address, the tag value will change.

$$\text{Tag} = \frac{\text{Byte Address}}{\text{Byte Address}}$$
$$= \frac{1024}{1024}$$
$$= 1$$

$$\text{Block number} = \left(\frac{1024}{32}\right)\%32$$
$$= 0$$

Byte-Address	**Tag**	**Corresponding Block Number**	**Hit/Miss**	**Reason**
0	0	0	Miss	Not in the cache
4	0	0	Hit	Already present
16	0	0	Hit	Already Present
132	0	4	Miss	Not in cache
232	0	7	Miss	Not in cache

160	0	5	Miss	Not in cache
1024	1	0	Miss	Change of tag
30	0	0	Miss	Again, change of tag
140	0	4	Hit	Already present
3100	3	0	Miss	Change of tag
180	0	5	Hit	Already present
2180	2	4	Miss	Change of tag

From the above table, the total number of hits is 4 and total number of cache access is 12.

$$\text{Hit ratio} = \frac{\text{Number of hits}}{\text{Number of cache access}}$$
$$= \frac{4}{12} \times 100$$
$$= 33.34\%$$

5.5.6) For constant miss latency, what is the optimal block size?

Consider the following information:

Average references per instruction for a 1-CPI system = 1.35

The miss latency is a constant.

The following are the miss rates for different block sizes:

For block size 8, miss rate is 4%=0.04,

For block size 16, miss rate is 3% =0.03,

For block size 32, miss rate is 2%=0.02,

For block size 64, miss rate is 1.5%=0.015,

For block size 128, miss rate is 1%=0.01.

Assume that the constant miss latency be C.

The size of cache block affects both the miss latency and miss rate.

The average latency for the block size (B) is calculated using the following formula:

$$Average\ \ latency = miss\ rate \times miss\ \ latency$$

The average latency for the block size (B) of 8 can be calculated as:

$$\begin{aligned} Average\ \ latency &= miss\ rate \times miss\ \ latency \\ &= 0.04 \times C \\ &= 0.04C \end{aligned}$$

The average latency for the block size (B) of 16 can be calculated as:

$$\begin{aligned} Average\ \ latency &= miss\ rate \times miss\ \ latency \\ &= 0.03 \times C \\ &= 0.03C \end{aligned}$$

The average latency for the block size (B) of 32 can be calculated as:

$$\begin{aligned} Average\ \ latency &= miss\ rate \times miss\ \ latency \\ &= 0.02 \times C \\ &= 0.02C \end{aligned}$$

The average latency for the block size (B) of 64 can be calculated as:

$$\begin{aligned} Average\ \ latency &= miss\ \ rate \times miss\ \ latency \\ &= 0.015 \times C \\ &= 0.015C \end{aligned}$$

The average latency for the block size (B) of 128 can be calculated as:

$$\begin{aligned} Average\ \ latency &= miss\ \ rate \times miss\ \ latency \\ &= 0.01 \times C \\ &= 0.01C \end{aligned}$$

The minimum latency is 0.01C for the block size 128.

Thus, the optimal block size for the constant miss latency is 128.