Sabbir Ahmed

CMSC 411 - HW 05

DATE: November 28, 2017

---

1. The table below lists parameters for different direct-mapped cache designs.

	Cache Data Size	**Cache Block Size**
i)	64 kB	1 word
ii)	64 kB	2 words

(a) **Question** Calculate the total number of bits required for the cache listed in the table, assuming a 32-bit address.

Answer

Using:

$$2^n \times (\text{valid field size} + \text{tag size} + \text{block size})$$

where

$$\begin{aligned} \text{tag size} &= 32 - (n + m + 2) \\ n &= \log_2(\text{cache data size}) \\ m &= \log_2(\text{block size}) \\ \text{valid bits} &= 1 \end{aligned}$$

Total number of bits required for design i):

$$\begin{aligned} n &= \log_2(64\text{ kB}) = 16 \\ m &= \log_2(1) = 0 \\ \text{tag size} &= 32 - (16 + 0 + 2) = 12 \\ \text{valid bits} &= 1 \\ \text{total bits} &= 2^{16} \times (1 + 12 + 2) \\ &= 15 \times 2^{16} \text{ tag bits} \end{aligned}$$

Total number of bits required for design ii):

$$
\begin{aligned}
n &= \log_2(64\text{ kB}) = 16 \\
m &= \log_2(2) = 1 \\
\text{tag size} &= 32 - (16 + 1 + 2) = 13 \\
\text{valid bits} &= 1 \\
\text{total bits} &= 2^{16} \times (1 + 13 + 2) \\
&= 2^{20} \text{ tag bits}
\end{aligned}
$$

(b) **Question** What is the total number of bits if the cache is organized as a 4-way associative with one word blocks?

Answer

Since there are 16 (2^1) bytes per block, a 32-bit address yields 32 - 1 = 31 bits to be used for index and tag. The number of sets:

$$\text{number of sets} = \log_2(64\text{ kB}) = 16$$

Each degree of associativity decreases the number of sets by a factor of 2 and thus decreases the number of bits used to index the cache by 1 and increases the number of bits in the tag by 1.

$$\frac{64\text{ kB}}{4} = 16\text{ kB}$$

$$
\begin{aligned}
\text{total bits} &= (31 - 10) \times 1 \times 16\text{ kB} \\
&= 21 \times 16\text{ K tag bits}
\end{aligned}
$$

2. For a pipeline with a perfect CPI = 1 if no memory-access related stall, consider the following program and cache behaviors.

Data Reads Per 1000 Instructions	Data Writes Per 1000 Instructions	Instruction Cache Miss Rate	Data Cache Miss Rate	Block Size (Byte)
200	160	0.20%	2%	8

(a) **Question** For a write-through, write-allocate cache with sufficiently large write buffer (i.e., no buffer caused stalls), what's the minimum read and write bandwidths (measured by byte-per-cycle) needed to achieve a CPI of 2?

Answer

Let I be the number of instructions and W be the read/write bandwidth.

$$\begin{aligned}\text{Data cache read miss penalty} &= I \times \frac{200}{1000} \times \frac{2}{100} \times \left(\frac{8}{W} + 1\right) \\ &= 0.004I \times \left(\frac{8}{W} + 1\right)\end{aligned}$$

$$\begin{aligned}\text{Data cache write miss penalty} &= I \times \frac{160}{1000} \times \frac{2}{100} \times \left(\frac{8}{W} + 1\right) \\ &= 0.0032I \times \left(\frac{8}{W} + 1\right)\end{aligned}$$

$$\begin{aligned}\text{Instruction cache read miss penalty} &= I \times \frac{0.20}{100} \times \left(\frac{8}{W} + 1\right) \\ &= 0.002I \times \left(\frac{8}{W} + 1\right)\end{aligned}$$

For CPI = 2,

$$\begin{aligned}I \times 2 &= \text{Hit time + miss penalty} \\ &= I + I \times (0.004 + 0.0032 + 0.002) \times \left(\frac{8}{W} + 1\right) \\ &= 0.0092I \times \left(\frac{8}{W} + 1\right) \\ \Rightarrow I \times 2 &= 0.0092I \times \left(\frac{8}{W} + 1\right) \\ 2 &= 0.0092 \times \left(\frac{8}{W} + 1\right) \\ \therefore W &\approx 0.037 \text{ byte per cycle}\end{aligned}$$

(b) **Question** For a write-back, write-allocate cache, assuming 30% of replaced data cache blocks are dirty, what's the minimal read and write bandwidths needed for a CPI of 2?

Answer

Let I be the number of instructions and W be the read/write bandwidth.

$$\begin{aligned}\text{Data cache read miss penalty} &= I \times \frac{200}{1000} \times \frac{2}{100} \times \left(1 + \frac{30}{100}\right) \times \left(\frac{8}{W} + 1\right) \\ &= 0.0052I \times \left(\frac{8}{W} + 1\right)\end{aligned}$$

$$\begin{aligned}\text{Data cache write miss penalty} &= I \times \frac{160}{1000} \times \frac{2}{100} \times \left(1 + \frac{30}{100}\right) \times \left(\frac{8}{W} + 1\right) \\ &= 0.00416I \times \left(\frac{8}{W} + 1\right)\end{aligned}$$

$$\begin{aligned}\text{Instruction cache read miss penalty} &= I \times \frac{0.20}{100} \times \left(\frac{8}{W} + 1\right) \\ &= 0.002I \times \left(\frac{8}{W} + 1\right)\end{aligned}$$

For CPI = 2,

$$\begin{aligned}I \times 2 &= \text{Hit time + miss penalty} \\ &= I + I \times (0.0052 + 0.00416 + 0.002) \times \left(\frac{8}{W} + 1\right) \\ &= 0.01136I \times \left(\frac{8}{W} + 1\right) \\ \Rightarrow I \times 2 &= 0.01136I \times \left(\frac{8}{W} + 1\right) \\ 2 &= 0.01136 \times \left(\frac{8}{W} + 1\right) \\ \therefore W &\approx 0.046 \text{ byte per cycle}\end{aligned}$$

3. Using the sequences of 32-bit memory read references, given as word addresses in the following table:

6	214	175	214	6	84	65	174	64	105	85	215

For each of these read accesses, identify the binary address, the tag, the index, and whether it experiences a hit or a miss, for each of the following cache configurations. Assume the cache is initially empty.

(a) **Question** A direct-mapped cache with 16 one-word blocks.

Answer

$\because$ 8 blocks, 1 word per block, 4 bytes per word,

$\therefore$ $\log_2(8)$ = 3-bit offset, 3-bit index and (16-3-3) = 10 bits tag

Table 1: A Direct-mapped Cache With 16 One-word Blocks

Memory	Binary	Tag	Index	Hit / Miss
6	0000000000000110	0000000000	000	miss
214	0000000011010110	0000000011	010	miss
175	0000000010101111	0000000010	101	miss
214	0000000011010110	0000000011	010	hit
6	0000000000000110	0000000000	000	hit
84	0000000001010100	0000000001	010	miss
65	0000000001000001	0000000001	000	miss
174	0000000010101110	0000000010	101	miss
64	0000000001000000	0000000001	000	miss
105	0000000001101001	0000000001	101	miss
85	0000000001010101	0000000001	010	miss
215	0000000011010111	0000000011	010	miss

(b) **Question** A direct-mapped cache with two-word blocks and a total size of 8 blocks.

Answer

$\because$ 8 blocks, 2 words per block, 4 bytes per word,

$\therefore$ $\log_2(8)$ = 3-bit offset, 3-bit index and (16-3-3) = 10 bits tag

Table 2: A Direct-mapped Cache With Two-word Blocks And A Total Size Of 8 Blocks

Memory	Binary	Tag	Index	Hit / Miss
6	0000000000000110	0000000000	000	miss
214	0000000011010110	0000000011	010	miss
175	0000000010101111	0000000010	101	miss
214	0000000011010110	0000000011	010	hit
6	0000000000000110	0000000000	000	hit
84	0000000001010100	0000000001	010	miss
65	0000000001000001	0000000001	000	miss
174	0000000010101110	0000000010	101	hit
64	0000000001000000	0000000001	000	hit
105	0000000001101001	0000000001	101	miss
85	0000000001010101	0000000001	010	hit
215	0000000011010111	0000000011	010	miss

(c) **Question** A fully associative cache with two-word blocks and a total size of 8 words. Use LRU replacement.

Answer

$\because$ 8 blocks, 2 words per block, 4 tags

$\therefore$ $\log_2(8)$ = 3-bit offset, 0-bit index and (16-0-3) = 13 bits tag

Table 3: A Fully Associative Cache With Two-word Blocks And A Total Size Of 8 Words With LRU Replacement. The Integers Next to The Tags Indicate the Age of the Content

Memory	**00000000 + Binary**	**Tag 1**		**Tag 2**		**Tag 3**		**Tag 4**		**Hit / Miss**
6	00000110	00000	1							miss
214	11010110	00000	2	11010	1					miss
175	10101111	00000	3	11010	2	10101	1			miss
214	11010110	00000	4	11010	1	10101	2			hit
6	00000110	00000	1	11010	2	10101	3			hit
84	01010100	00000	2	11010	3	10101	4	01010	1	miss
65	01000001	00000	3	11010	4	01000	1	01010	2	miss
174	10101110	00000	4	11010	5	10101	1	01010	3	miss
64	01000000	00000	5	11010	6	01000	1	01010	4	miss
105	01101001	00000	6	01101	1	01000	2	01010	5	miss
85	01010101	00000	7	01101	2	01000	3	01010	1	hit
215	11010111	11010	1	01101	3	01000	4	01010	2	miss

(d) **Question** A 2-way set associative cache with one-word block size and total size of 8 words, while applying LRU replacement policy.

Answer

$\because$ 8 blocks, 1 word per block, 4 bytes per word, 2 way- set,

$\because$ The number of sets $= \frac{8 \text{ words}}{2 \text{ blocks per set}}$ = 4 sets

$\therefore \log_2(8)$ = 3-bit offset, 2-bit index and (16-2-3) = 11 bits tag

Table 4: A 2-way Set Associative Cache With One-word Block Size And Total Size Of 8 Words With LRU Replacement Policy

Memory	Binary	Tag	Index	Hit / Miss
6	0000000000000110	0000000000	00	miss
214	0000000011010110	0000000011	10	miss
175	0000000010101111	0000000010	01	miss
214	0000000011010110	0000000011	10	hit
6	0000000000000110	0000000000	00	hit
84	0000000001010100	0000000001	10	miss
65	0000000001000001	0000000001	00	miss
174	0000000010101110	0000000010	01	miss
64	0000000001000000	0000000001	00	miss
105	0000000001101001	0000000001	01	miss
85	0000000001010101	0000000001	10	miss
215	0000000011010111	0000000011	10	miss

Table 5: Contents of the Cache After Each Addition

00	01	10	11
6			
6		214	
6	175	214	
6	175	214, 84	
6, 65	175	214, 84	
65, 64	174	214, 84	
65, 64	174, 105	214, 84	
65, 64	174, 105	84, 85	
65, 64	174, 105	85, 215	