Department of Electrical and Computer Engineering
The University of Texas at Austin
EE 460N/382N.1, Spring 2018
Instructor: Dr. Mattan Erez
TAs: Steven Flolid, Armand Behroozi, Abraham Gonzalez

Problem Set 4 Solutions

1. A byte-addressable computer has a small data cache capable of holding eight 32-bit words. Each cache block consists of one 32-bit word. When a given program is executed, the processor reads data from the following sequence of hex addresses:

0x200, 0x204, 0x208, 0x20C, 0x2F4, 0x2F0, 0x200, x204, x218, 0x21C, 0x24C, 0x2F4

This pattern is repeated four times.

Show the contents of the cache at the end of each of the four iterations of this loop, if a direct-mapped cache is used. Compute the hit rate for this example. Assume that the cache is initially empty.

The address is divided into 3 portions:

- address[1:0] (2 bits) for the byte in the block
- address [4:2] (3 bits) for the cache index
- address[11:5] (7 bits) for the tag

The contents of the cache at the end of each pass are the same. They are shown below:

Valid	Tag	Data (addresses are written inside each byte)			
1	0010 000	203	202	201	200
1	0010 000	207	206	205	204
1	0010 000	20B	20A	209	208
1	0010 010	24F	24E	24D	24C
1	0010 111	2F3	2F2	2F1	2F0
1	0010 111	2F7	2F6	2F5	2F4
1	0010 000	21B	21A	219	218
1	0010 000	21F	21E	21D	21C

The hit/miss information for each pass is shown below:

Reference	200	204	208	20C	2F4	2F0	200	204	218	21C	24C	2F4
Pass 1:	M	M	M	M	M	M	H	H	M	M	M	H
Pass 2:	H	H	H	M	H	H	H	H	H	H	M	H
Pass 3:	H	H	H	M	H	H	H	H	H	H	M	H
Pass 4:	H	H	H	M	H	H	H	H	H	H	M	H

Hit rate is 33/48

2. An LC-3b system ships with a two-way set associative, writeback cache, with perfect LRU replacement (LRU replacement requires 1 extra bit per set in the tag store to indicate which of the two blocks in that set was used most recently and which was used earlier; a writeback cache needs another additional tag bit per cache location). The tag store requires a total of 4352 bits of storage (the tag array is the information in the cache that doesn't store data and is used to determine hit or miss). How many bytes are stored by each cache location (i.e., what is the block size of the cache)?
Please show all your work.
Hints: recall that:
(1) number of sets must be a power of 2,
(2) number of bytes per block must be a power of 2,
(3) all numbers you use should be integers,
(4) you can "search" for an answer if you do it intelligently, and
(5) you don't need a calculator.

The size of the tag store is $2^{12} + 2^8$. We know that the size of the tag store can be given as the product of number of sets and number of bits per set.
We also know that address space is 16-bits. Hence, tag + index + offset = 16
In order to find offset, we need to find index and tag. The following bits are necessary for each set of the tag store:

- 1 bit for LRU (remember: you only need one bit per set for a 2-way associative cache)
- 2 valid bits
- 2 dirty bits
- 2 × tag bits

Total number of bits per set is 5 + 2 × tag. An important conclusion that can be drawn is that number of bits per set will always be an odd number.
We will also use the fact that number of sets is always a power of 2 (since it is indexed using an integer number of bits). Given the size of the tag store, index has to be a number less than 8 (the size of the tag store is indivisible by 2^9 or greater).
The only value of index that fits both criterion is 8. Therefore:
Number of sets = $2^8 = 256$

Bits per row = 4352 ÷ 256 = 17
17 = 5 + 2 × tag ⇒ tag = 6
6 + 8 + offset = 16 ⇒ offset = 2 bits
Hence, the cache block size is **4 bytes**

3. In Lecture 9 Slide 9, Mattan went over a timing model for memory based on if there was a single cache or not. Similar to this problem create a timing model for a 2 level cache ($1 and $2) hierarchy. The following should be used to represent the various variables.

L_{mem} : Latency for a memory access
N_{mem} : Number of instructions that use memory
$N_{non\text{-}mem}$: Number of instructions that do not use memory
$L_{\$1}$, $L_{\$2}$: Latency for a $1 or $2 memory access
$M_{\$1}$, $M_{\$2}$: Miss rate for $1 or $2

Time with 2 $ hierarchy = $N_{non\text{-}mem} + N_{mem} + (N_{mem} * (L_{\$1} + (M_{\$1} * (L_{\$2} + (M_{\$2} * L_{mem})))))$