ECE4700J Computer Architecture

Summer 2022

HW #4

Due: **2:59am (Beijing Time) July 4th, 2022**

Please submit a single ***PDF*** *file on Canvas*

Q1 (30%): Consider the following program and cache behaviors.

Data Reads per 1000 instructions	Date Write per 1000 Instructions	Instruction Cache Miss Rate	Data Cache Miss Rate	Block Size (bytes)
250	100	0.30%	2%	64

(1) (15%) Suppose a CPU with a write-through, write-allocate cache achieves a CPI of 2. What are the read and write bandwidths (measured by bytes per cycle) between RAM and the cache? (Assume each miss generates a request for one block.)

Sample Solution:

When the CPI is 2, there are, on average, 0.5 instruction accesses per cycle. 0.3% of these instruction accesses cause a cache miss (and subsequent memory request). Assuming each miss requests one block, instruction accesses generate an average of 0.5*.003*64 = 0.096 bytes/cycle of read traffic. Twenty-five percent of instructions generate a read request. Two percent of these generate a cache miss; thus, read misses generate an average of 0.5*0.25*0.02*64= 0.16 bytes/cycle of read traffic. Ten percent of instructions generate a write request. Two percent of these generate a cache miss. Because the cache is a write-through cache, only one word (8 bytes) must be written back to memory; but, every write is written through to memory (not just the cache misses). Thus, write misses generate an average of 0.5*0.1*8 = 0.4 bytes/cycle of write traffic. Because the cache is a write-allocate cache, a write miss also makes a read request to RAM. Thus, write misses require an average of 0.5*0.1*0.02*64 = 0.064 bytes/cycle of read traffic.

Hence: The total read bandwidth = 0.096 + 0.16 + 0.064 = 0.32 bytes/cycle, and the total write bandwidth is 0.4 bytes/cycle.

(2) (15%) For a write-back, write-allocate cache, assuming 30% of replaced data cache blocks are dirty, what are the read and write bandwidths needed for a CPI of 2?

Sample Solution:

The instruction and data read bandwidth requirement is the same as in (1). With a write-back cache, data are only written to memory on a cache miss. But, it is written on every cache miss (both read and write), because any line could have dirty data when evicted, even if the eviction is caused by a read request. Thus, the data write bandwidth requirement becomes 0.5*(0.25+0.1)*0.02*0.3*64 =0.0672 bytes/cycle.

Q2 (10%): Based on what you have learned in this course and in VE370, please describe on your own words at least 3 differences between **Cache** and **Virtual Memory**.

Sample Solution:

Reference:
https://www.javatpoint.com/cache-memory-vs-virtual-memory#:~:text=1.&text=Cache%20Memory%20is%20the%20high,memory%20in%20the%20computer%20system.

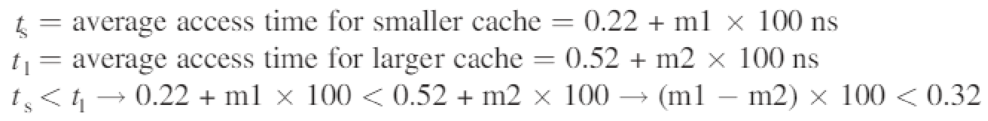
S. N.	Parameter Difference	Cache Memory	Virtual Memory
1.	**Definition**	Cache Memory is the high speed of computer memory that reduces the access time of files or documents from the main memory.	Virtual Memory is a logical unit of computer memory that increases the capacity of main memory by storing or executing programs of larger size than the main memory in the computer system.
2.	**Memory Unit**	Cache Memory is defined as a memory unit in a computer system.	Virtual Memory is not defined as a memory unit.
3.	**Size**	Its size is very small as compared to Virtual Memory.	Its size is very large as compared to the Cache Memory.
4.	**Speed**	It is a high-speed memory as compared to Virtual Memory.	It is not a high-speed memory as compared to the Cache Memory.

5.	**Operation**	Generally, it stores frequently used data in the cache memory to reduce the access time of files.	The virtual memory keeps those data or programs that may not completely be placed in the main memory.
6.	**Management**	Cache Memory is controlled by the hardware of a system.	Whereas the virtual memory is control by the Operating System (OS).
7.	**Mapping**	It does not require a mapping structure to access the files in Cache Memory.	It requires a mapping structure to map the virtual address with a physical address.

Q3 (20%): Whereas larger caches have lower miss rates, they also tend to have longer hit times. Assume a direct-mapped 8 KB cache has 0.22 ns hit time and miss rate m1; also assume a 4-way associative 64 KB cache has 0.52 ns hit time and a miss rate m2.

(1) (10%) If the miss penalty is 100 ns, when would it be advantageous to use the smaller cache to reduce the overall memory access time?

Sample Solution:



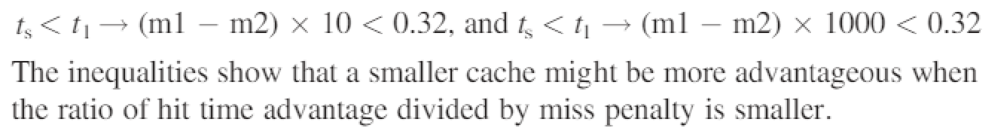
t_s = average access time for smaller cache = $0.22 + \mathrm{m1} \times 100$ ns

t_l = average access time for larger cache = $0.52 + \mathrm{m2} \times 100$ ns

$t_s < t_l \rightarrow 0.22 + \mathrm{m1} \times 100 < 0.52 + \mathrm{m2} \times 100 \rightarrow (\mathrm{m1} - \mathrm{m2}) \times 100 < 0.32$

(2) (10%) Repeat part (1) for miss penalties of 10 and 1000 cycles. Conclude when it might be advantageous to use a smaller cache.

Sample Solution:

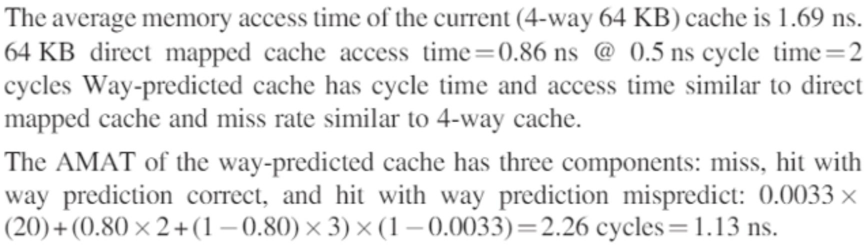


$t_s < t_l \rightarrow (\mathrm{m1} - \mathrm{m2}) \times 10 < 0.32$, and $t_s < t_l \rightarrow (\mathrm{m1} - \mathrm{m2}) \times 1000 < 0.32$

The inequalities show that a smaller cache might be more advantageous when the ratio of hit time advantage divided by miss penalty is smaller.

Q4 (10%): You are investigating the possible benefits of a way predicting L1 cache. Assume that a 64 KB four-way set associative single-banked L1 data cache is the cycle time limiter. This cache has an AMAT of 1.69ns. The clock cycle time of the system is 0.5ns. For an alternative cache organization, you are considering a **way-predicted cache** modeled as **a 64 KB direct mapped cache** with 80% prediction accuracy. The 64KB direct mapped cache has an access time of 0.86ns, and a miss rate of 0.33% along with miss penalty of 20 cycles. Unless stated otherwise, assume that a mispredicted way access that hits in the cache takes one more cycle. What is the average memory access time of the way-predicted cache?

Sample Solution:



The average memory access time of the current (4-way 64 KB) cache is 1.69 ns. 64 KB direct mapped cache access time = 0.86 ns @ 0.5 ns cycle time = 2 cycles Way-predicted cache has cycle time and access time similar to direct mapped cache and miss rate similar to 4-way cache.

The AMAT of the way-predicted cache has three components: miss, hit with way prediction correct, and hit with way prediction mispredict: $0.0033 \times (20) + (0.80 \times 2 + (1 - 0.80) \times 3) \times (1 - 0.0033) = 2.26$ cycles = 1.13 ns.

Q5 (10%): Use your own words to explain the advantage and disadvantage does a Harvard L1 cache have over a unified L1 cache?

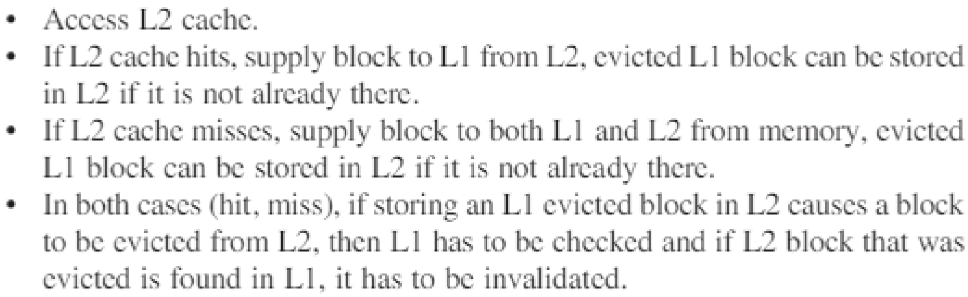
Sample Solution:

A Harvard (split) cache permits the processor to access both an instruction word and a data word in a single cache memory cycle. This can significantly increase the performance of the processor. The only real drawback of a Harvard cache is that the processor needs two memory busses (two complete sets of memory address and data lines), one to each of the caches. However, when the Harvard cache is implemented within the processor chip, the separate memory busses never have to escape the chip, being implemented completely in the metal interconnections on the chip.

Q6 (20%): Consider a two-level memory hierarchy made of L1 and L2 data caches. Assume that both caches use write-back policy on write hit and both have the same block size. List the actions taken in response to the following events:
(1) (10%) An L1 cache miss when the caches are organized in an inclusive hierarchy:

Example Answer: When there is L1 cache miss, it will first access L2 cache, then if hit, XXX, if miss XXX. In both hit and miss, if storing an L1 evicted block in L2 causes a block to be evicted from L2, then XXX.

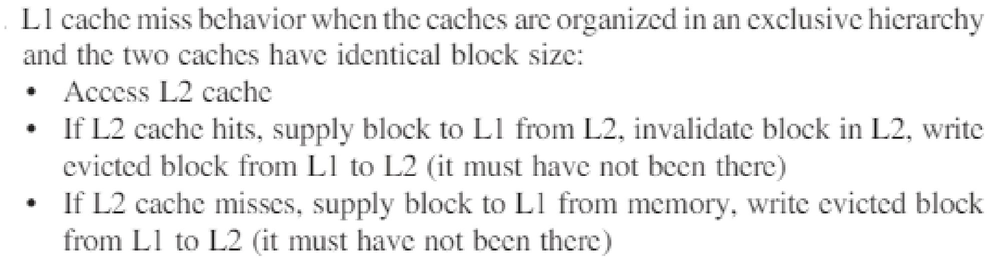
Sample Solution:



- Access L2 cache.
- If L2 cache hits, supply block to L1 from L2, evicted L1 block can be stored in L2 if it is not already there.
- If L2 cache misses, supply block to both L1 and L2 from memory, evicted L1 block can be stored in L2 if it is not already there.
- In both cases (hit, miss), if storing an L1 evicted block in L2 causes a block to be evicted from L2, then L1 has to be checked and if L2 block that was evicted is found in L1, it has to be invalidated.

(2) (10%) An L1 cache miss when the caches are organized in an exclusive hierarchy. The style of the answers should be similar to (1).

Sample Solution:



L1 cache miss behavior when the caches are organized in an exclusive hierarchy and the two caches have identical block size:

- Access L2 cache
- If L2 cache hits, supply block to L1 from L2, invalidate block in L2, write evicted block from L1 to L2 (it must have not been there)
- If L2 cache misses, supply block to L1 from memory, write evicted block from L1 to L2 (it must have not been there)