CSE 520 Computer Architecture II
Term: Spring 2026
Lead Instructor: **Prof. Michel A. Kinsy**

Problem Set 3

Posted April. 6th, 2026

http://ascslab.org/courses/cse520/index.html

General guidelines: Always state your assumptions and clearly explain your answers.

Part 1: Vector Processor and GPU Hybrid System

Problem 1

Albert Bitdiddle is looking at the following code, which multiplies two vectors that contain single-precision complex values:

```
For (i=0; i>300; i++){
        c_re[i] = a_re[i] * b_re[i] - a_im[i] * b_im[i];
        c_im[i] = a_re[i] * b_im[i] - a_im[i] * b_re[i];
}
```

Assume that the processor runs at 700 MHz and has a maximum vector length of 64. The load/store unit has a start-up overhead of 15 cycles; the multiply unit, 8 cycles; and the add/subtract unit, 5 cycles.

Question a. What is the arithmetic intensity of this kernel? Justify your answer.

Question b. Convert this loop into VMIPS assembly code using strip mining.

Question c. Assuming chaining and a single memory pipeline, how many chimes are required? How many clock cycles are required per complex result value, including start-up overhead?

Question d. If the vector sequence is chained, how many clock cycles are required per complex result value, including overhead?

Question e. Now assume that the processor has three memory pipelines and chaining. If there are no bank conflicts

Problem 2

In this problem, you will assist Albert Bitdiddle compare the performance of a vector processor with a hybrid system that contains a scalar processor and a GPU-based coprocessor. In the hybrid system, the host processor has superior scalar performance to the GPU, so in this case all scalar code is executed on the host processor while all vector code is executed on the GPU.

We will refer to the first system as the vector computer and the second system as the hybrid computer. Assume that your target application contains a vector kernel with an arithmetic intensity of 0.5 FLOPs per DRAM byte accessed; however, the application also has a scalar component which that must be performed before and after the kernel in order to prepare the input vectors and output vectors, respectively. For a sample dataset, the scalar portion of the code requires 400 ms

of execution time on both the vector processor and the host processor in the hybrid system. The kernel reads input vectors consisting of 200 MB of data and has output data consisting of 100 MB of data.

The vector processor has a peak memory bandwidth of 30 GB/sec and the GPU has a peak memory bandwidth of 150 GB/sec.

The hybrid system has an additional overhead that requires all input vectors to be transferred between the host memory and GPU local memory before and after the kernel is invoked.

The hybrid system has a direct memory access (DMA) bandwidth of 10 GB/sec and an average latency of 10 ms.

Assume that both the vector processor and GPU are performance bound by memory bandwidth. Compute the execution time required by both computers for this application

Part 2: Memory Technology

Problem 1

A potential drawback of SSDs is that the underlying flash memory can wear out. For example, one major manufacturer guarantees 1 petabyte (10^15 bytes) of random writes for their SSDs before they wear out. Given this assumption, estimate the lifetime (in years) of the SSD in Figure 1 for the following workloads:

Question a. Worst case for sequential writes: The SSD is written to continuously at a rate of 170 MB/s (the average sequential write throughput of the device).

Question b. Worst case for random writes: The SSD is written to continuously at a rate of 14 MB/s (the average random write throughput of the device).

Question c. Average case: The SSD is written to at a rate of 20 GB/day (the average daily write rate assumed by some computer manufacturers in their mobile computer workload simulations).

Reads		Writes	
Sequential read throughput	250 MB/s	Sequential write throughput	170 MB/s
Random read throughput	140 MB/s	Random write throughput	14 MB/s
Random read access time	30 μs	Random write access time	300 μs

Figure 1: Performance characteristics of a typical solid-state disk. Source: Intel X25-E SATA solid state drive product manual.

Part 3: Caching and Cache Structures

Problem 1

Caches are important to providing a high-performance memory hierarchy to processors. Below is a list of 32-bit memory address references, given as word addresses:

3, 180, 43, 2, 191, 88, 190, 14, 181, 44, 186, 253

Question a. For each of these references, identify the binary address, the tag, and the index given a direct-mapped cache with 16 one-word blocks. Also list if each reference is a hit or a miss, assuming the cache is initially empty.

Question b. For each of these references, identify the binary address, the tag, and the index given a direct-mapped cache with two-word blocks and a total size of 8 blocks. Also list if each reference is a hit or a miss, assuming the cache is initially empty.

Question c. You are asked to optimize a cache design for the given references. There are three direct-mapped cache designs possible, all with a total of 8 words of data: C1 has 1-word blocks, C2 has 2-word blocks, and C3 has 4-word blocks. In terms of miss rate, which cache design is the best? If the miss stall time is 25 cycles, and C1 has an access time of 2 cycles, C2 takes 3 cycles, and C3 takes 5 cycles, which is the best cache design?

Problem 2

There are many different design parameters that are important to a cache's overall performance. Below are listed parameters for a direct-mapped cache design.

- Cache Data Size: 32 KiB
- Cache Block Size: 2 words
- Cache Access Time: 1 cycle

Question a. Calculate the total number of bits required for the cache listed above, assuming a 32-bit address. Given that total size, and the total size of the closest direct-mapped cache with 16-word blocks of equal size or greater. Explain why the second cache, despite its larger data size, might provide slower performance than the first cache.

Question b. Generate a series of read requests that have a lower miss rate on a 2 KiB 2-way set associative cache than the cache listed above. Identify one possible solution that would make the cache listed have an equal or lower miss rate than the 2 KiB cache. Discuss the advantages and disadvantages of such a solution.

Question c. We saw in lecture the typical method to index a direct-mapped cache, specifically (Block address) modulo (Number of blocks in the cache). Assuming a 32-bit address and 1024 blocks in the cache, consider a different indexing function, specifically (Block address [31:27] XOR Block address [26:22]). Is it possible to use this to index a direct-mapped cache? If so, explain why and discuss any changes that might need to be made to the cache. If it is not possible, explain why.

Problem 3

In this exercise, we will look at the different ways capacity affects overall performance. In general, cache access time is proportional to capacity. Assume that main memory accesses take 70 ns and that memory accesses are 36% of all instructions. The following table shows data for ***Ll*** caches attached to each of two processors, Pl and P2.

	L1 Size	L1 Miss Rate	L1 Hit time
P1	2 KiB	8.0%	0.66ns
P2	4 KiB	6.0%	0.90ns

Question a: Assuming that the ***Ll*** hit time determines the cycle times for Pl and P2, what are their respective clock rates?

Question b: What is the Average Memory Access Time for Pl and P2?

Question c: Assuming a base CPI of 1.0 without any memory stalls, what is the total CPI for Pl and P2? Which processor is faster?

Problem 4

For a 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

Question a: What is the cache block size (in words)?

Question b: How many entries does the cache have?

Question c: What is the ratio between total bits required for such a cache implementation over the data storage bits?

Starting from power on, the following byte-addressed cache references are recorded.

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

Question d: How many blocks are replaced?

Questionef: What is the hit ratio?

Question f: List the final state of the cache, with each valid entry represented as a record of <index, tag, data>.

Problem 5

Albert has an L1 data cache, L2 cache, and main memory. The hit rates and hit times for each are:

- 80% hit rate, 2-cycle hit time to L1.
- 75% hit rate, 20-cycle hit time to L2.
- 100% hit rate, 200-cycle hit time to main memory.

Question a. What fraction of accesses is serviced from L2?

Question b. What fraction of accesses is serviced from main memory?

Problem 6

Question a. Suppose we have a memory and a direct-mapped cache with the following characteristics:

- Memory is byte addressable

- Memory addresses are 16 bits (i.e., the total memory size is 2^{16} = 65536 bytes)
- The cache has 8 rows (i.e., 8 cache lines)
- Each cache row (line) holds 16 bytes of data

In the spaces below, indicate how the 16 address bits are allocated to the offset, index, and tag parts of the address used to reference the cache:

_____ **offset bits**

_____ **tag bits**

_____ **index bits**

Below is a sequence of four binary memory addresses in the order they are used to reference memory. Assume that the cache is initially empty. For each reference, write down the tag and index bits and circle either hit or miss to indicate whether that reference is a hit or a miss.

Memory address	**Tag**	**Index**	**Hit / Miss (circle)**
0010 1101 1011 0011			**Hit Miss**
0000 0110 1111 1100			**Hit Miss**
0010 1101 1011 1000			**Hit Miss**
1010 1010 1010 1011			**Hit Miss**

Question b. With a 32-bit address, how many total **Tag** bits are required for a direct-mapped cache with 256 bytes of data and 16-byte blocks?

Question c. Give the block number to map byte address **0x0018** to, when we have a 16-block direct-mapped cache, with 16-byte blocks and Tag bits are the high order bits.

Question d. Suppose we have a 2-Way Set Associative cache with 256 bytes of data, 16-byte blocks, and 32-bit physical address, instead of a direct-mapped cache. Fill in the address breakdown table for this cache with the following **fields:** Index, Tag, and Byte offset and their **corresponding address bits**.

Address Bits	31 0
Field	Tag

Question e. During the design phase of his memory system, Ben generates a sequence of block addresses: 0x0200, 0x0018, 0x0200, 0x00B6, 0x0018. He runs them through a 2-way set associative, consisting of four 4-byte blocks. Please fill in the rest of table 1, to help Ben analyze his cache performance. Ben is using FIFO for his cache line replacement policy.

Table 1: 2-Way Set Associative

Address of memory accessed	**Hit or Miss**	**Content of cache blocks after reference**			
		Set 0	**Set 0**	**Set 1**	**Set 1**
0x0200	Miss	M[0x0200]			
0x0018					
0x0200					
0x00B6					
0x0018					

Question f. Ben has an L1 data cache, L2 cache, and main memory architecture. The hit rates and hit times for each are:
50% hit rate, 5 cycle hit time to L1.
70% hit rate, 15 cycle hit time to L2.
100% hit rate, 200 cycle hit time to main memory.

- What fraction of his accesses are serviced from L2? From main memory?
- What is the miss rate and miss time for the L2 cache?
- What is the miss rate and miss time for the L1 cache?
- If main memory is improved by 10%, what is the improvement in L1 miss time?
- Ben removes the L2 to add more L1. As a result, the new L1 hit rate is 75%. What is the improvement in L1 miss time?

Problem 7
Ben is learning virtual and physical addresses. For each configuration below, please help Ben identify the number of bits needed to specify: Virtual address, Physical address, Virtual page number, Physical page number, and Offset.

32-bit operating system, 4-KB pages, 1 GB of RAM

Virtual Address	**Physical Address**	**Virtual Page #**	**Physical Page #**	**Offset**

32-bit operating system, 16-KB pages, 2 GB of RAM

Virtual Address	Physical Address	Virtual Page #	Physical Page #	Offset

64-bit operating system, 16-KB pages, 16 GB of RAM

Virtual Address	Physical Address	Virtual Page #	Physical Page #	Offset

What are some advantages of using a larger page size?

What are some disadvantages of using a larger page size?