FINAL REVIEW

Mahdi Nazm Bojnordi

Assistant Professor

School of Computing

University of Utah

Memory Hierarchy Design

- The following C code implements the necessary dot product operation for a Sparse Matrix-Vector (SpMV) multiplication kernel. Assume that j is allocated in the register file only. Identify the existing temporal and spatial locality in the code.

```
double sum = 0.0;
for(j = 0; j<1000; j++) {
    sum += values[j] * vector[indices[j]];
}
```

Cache Replacement Policies

- Consider a computer system using a 2-way set associative level-1 cache, where blocks A, B, and C of a particular application are mapped to a single set.
  - Compute the miss rate of an IDEAL replacement policy for an access pattern comprising A, B, B, A, B, C, C, C, B, B, A, A, B.
  - Design a memory access pattern comprising 10 read requests to A, B, and C that results in a lower miss rate for the Most Recently Used (MRU) block replacement policy than the Least Recently Used (MRU) policy.

Cache Replacement Policies

- Consider a computer system using a 2-way set associative level-1 cache, where blocks A, B, and C of a particular application are mapped to a single set.
  - Compute the miss rate of an IDEAL replacement policy for an access pattern comprising A, B, B, A, B, C, C, C, B, B, A, A, B.

    A, B, B, A, B, C, C, C, B, B, A, A, B

  - Design a memory access pattern comprising 10 read requests to A, B, and C that results in a lower miss rate for the Most Recently Used (MRU) block replacement policy than the Least Recently Used (MRU) policy.

    LRU: A, B, C, A, B, C, A, B, C, A

    MRU: A, B, C, A, B, C, A, B, C, A

Average Memory Access Time

- Consider running an application on a computer system comprising a memory hierarchy with the following components.
  - 32KB L1: hit time: 1 cycles; miss rate: 5%
  - 256KB L2: hit time: 12 cycles; miss rate: 20%
  - 8MB L3: hit time: 25 cycles; miss rate: 40%
  - 8GB Memory: average access time: 400 cycles
- Compute the average memory access time for this application.

Average Memory Access Time

- Consider running an application on a computer system comprising a memory hierarchy with the following components.
  - 32KB L1: hit time: 1 cycles; miss rate: 5%
  - 256KB L2: hit time: 12 cycles; miss rate: 20%
  - 8MB L3: hit time: 25 cycles; miss rate: 40%
  - 8GB Memory: average access time: 400 cycles
- Compute the average memory access time for this application.

AMAT = 1 + 0.05 x (12 + 0.2 x (25 + 0.4 x 400))

Virtual Memory

- A virtually indexed, physically tagged (VIPT) L1 cache is used to hide the TLB access time per every memory request. Assume that the virtual address space is 4GB partitioned into 4KB pages; and L1 is a direct mapped cache storing 64B data blocks. Explain the issues and solutions for each of the following cases:
  - The size of L1 is 4KB.
  - The size of L1 is 64KB.

DRAM Address Mapping

- Consider the following sequence of memory addresses seen by a 1MB DRAM subsystem with 1KB rows:
  - 00000, 03480, 05480, 09480, 05080, 05780, 05180.
- All of the addresses are in hexadecimal. The DRAM subsystem comprises one channel, one rank, and sixteen banks with a single-queue, in-order command scheduler. The following address mapping schemes are possible at the memory controller.
  - Bank interleaved: channel:rank:bank:row:column
  - Row interleaved: row:channel:rank:bank:column
- Assuming that all of the banks are initially precharged (i.e., empty row buffers); which one of these address mapping schemes performs better?

DRAM Address Mapping

- Bank interleaved
  - channel(0):rank(0):bank(4):row(6):column(10)

	bank	row
00000	0	00
03480	0	0D
05480	0	15
09480	0	25
05080	0	14
05780	0	15
05180	0	14

DRAM Address Mapping

- Bank interleaved
  - row(6):channel(0):rank(0):bank(4):column(10)

	bank	row
00000	0	00
03480	D	00
05480	5	01
09480	5	02
05080	4	01
05780	5	01
05180	4	01

Chip Multiprocessing

- Consider an application running on a single-core processor that operates at 4GHz; the execution time is 10s; and the dynamic power consumption is equal to 40W. Assume that the processor consumes zero static power and 70% of the application can be executed fully in parallel on a four-core processor that operates at 2GHz. Assuming that the power consumption scales proportionally to frequency, what is the amount of energy consumed by the parallel application?

Chip Multiprocessing

- Consider an application running on a single-core processor that operates at 4GHz; the execution time is 10s; and the dynamic power consumption is equal to 40W. Assume that the processor consumes zero static power and 70% of the application can be executed fully in parallel on a four-core processor that operates at 2GHz. Assuming that the power consumption scales proportionally to frequency, what is the amount of energy consumed by the parallel application?

	Single Core	Quad Core
Frequency	4GHz	2GHz
Core Power	40W	20W
Time	10s	(2x3+2x7/4)s = 9.5s
Energy	400J	20x(9.5+3x3.5) = 400J

Cache Coherence

- Consider a four-core system comprising a single level of cache and a shared main memory. A Modified-Shared-Invalid (MSI) coherence protocol is implemented for the cache units; A is a shared variable accessed by the cores. Assuming that A is initially stored in the main memory, track and report the state of A in every cache unit for the following sequence of memory accesses.

  - P1: store A
  - P3: store A
  - P2: load A
  - P4: load A
  - P1: store A

Cache Coherence

- Consider a four-core system comprising a single level of cache and a shared main memory. A Modified-Shared-Invalid (MSI) coherence protocol is implemented for the cache units; A is a shared variable accessed by the cores. Assuming that A is initially stored in the main memory, track and report the state of A in every cache unit for the following sequence of memory accesses.

	P1	P2	P3	P4
P1: store A	M	I	I	I
P3: store A	I	I	M	I
P2: load A	I	S	S	I
P4: load A	I	S	S	S
P1: store A	M	I	I	I