**Lec 2 Introduction**

Q1. Why does memory access time increases with the size of the memory for any type of the memory (cache, main memory, ...)?

Q2. It is possible to have speedup better than P where P is the number of available processors.

1. True
2. False

Q3. In vector processors, does one instruction perform many operations?

1. Yes
2. No

Q4. Why is it difficult to further increase the number of pipelining stages in the processors?

Q5. How is it possible to prevent against transient faults in memories?

**Caches and Parallel programming models**

Q1. How many bits per cache set are needed to support LRU replacement policy for 16-way set-associative caches? (here I mean the total number of bits needed for LRU for one set in all 16 lines)

1. 4
2. 16
3. 64
4. 256

Q2. What is prefetch degree?

1. The number of blocks that are prefetched
2. The number of hardware units for prefetching
3. The number of instruction caches

Q3. We need to apply mutual exclusion when:

1. Two processes read from the same memory location
2. Two processes write to the same memory location
3. In both 1. and 2.

**Caches and shared memory model**

Q1. What is the reason for using Barrier in parallel programming?

1. To provide mutual exclusion mechanism
2. To provide a point for synchronization for multiple threads
3. To lock the shared variable

Q2. The sequence of 10 byte accesses for a direct-mapped cache with 4 blocks of 4 bytes is shown below. Accesses number 8, 9 and 10 will result in the following sequence of hits/misses:

|  |  |  |
| --- | --- | --- |
|  | **Address (binary)** | **Hit/Miss** |
| 1. | 110001 | Miss |
| 2. | 100111 |  |
| 3. | 001111 |  |
| 4. | 001100 |  |
| 5. | 010001 |  |
| 6. | 110010 |  |
| 7. | 100101 |  |
| 8. | 001110 |  |
| 9. | 100001 |  |
| 10. | 110101 |  |

1. M, M, M
2. M, H, M
3. H, M, M
4. H, M, H

**Message passing and bus-based shared memory systems**

Q1. One of the architectures of bus-based shared memory systems relies on shared L2 caches. What is the reason for having shared L2 caches?

Q2. In the following program assume that SEND takes 500 cycles to finish and unrelated computation takes 600 cycles. How long does it take to execute this program? Assume that both threads can send their messages at the same time?

**CODE FOR THREAD T1:                    CODE FOR THREAD T2:**

  A = 10;                                                        B = 5;

  ASEND(&A,sizeof(A),T2,SEND\_A);            ASEND(&B,sizeof(B),T1,SEND\_B);

   <Unrelated computation;>                            <Unrelated computation;>

  SRECV(&B,sizeof(B),T2,SEND\_B);              SRECV(&A,sizeof(B),T1,SEND\_A);

1. 500
2. 600
3. 1100

Q3. Is OpenMP language or Application Programming Interface (API)? If it is API, what languages is it specified for? Is it for shared memory or for message passing systems?

1. Programming Language for shared memory systems
2. API for C and C++ for message passing systems
3. API for C, C++ and Fortran for shared memory systems

**Quiz\_GPU**

Q1. Each kernel instance is a \_\_\_\_\_\_\_\_\_\_ or thread

1. work item
2. work group
3. wave front

Q2. What are the main benefits of OpenCL vs. CUDA? Select all that apply.

1. OpenCL is made for heterogeneous computing platforms.
2. OpenCL is hardware independent.
3. OpenCL is usually executing faster than CUDA

Q3. How does the GPU hide memory latency?

**Snooping cache coherence implementation**

Q1. What is the purpose of wired-NOR bus line for the MESI protocol?

Q2. What are the information that have to be exchanged between two caches (L1 and L2) for the snooping protocol with multilevel caches? Assume that the caches are inclusive. Describe information exchange both in L1->L2 and L2->L1 directions.

Q3. Disadvantages of the MESI protocol in comparison with MSI protocol are (list all that apply):

1. additional bus lines
2. more complex state machine
3. waiting for all processors to set the shared line before it is possible to change state after the read miss
4. MESI works worse than MSI when the program runs only on one processor.

**Directory based protocols**

Q1. The full map (or baseline directory protocol) directory can cause significant memory overhead. For a fixed size of the main memory, the overhead of the directory will increase if we increase the size of cache blocks.

1. True
2. False

Q2. Consider a multiprocessing system with 8 processors that have their local caches and they are connected to the main memory.

Assume that Full Map Directory cache coherence protocol with Centralized Directory Invalidate is implemented. Assume that directory for address X contained all 0s at the beginning. Fill the following table for the following sequence of instructions:

|  |  |  |
| --- | --- | --- |
| Time instant | Operation | Content of the directory for X |
| 1 | Processor 0 – read X | [1] |
| 2 | Processor 5 – read X | [2] |
| 3 | Processor 0 – writes to X | [3] |

Content of the directory include only presence bits. LSB corresponds to the cache of processor 0 .

*Centralized Directory Invalidate protocol description:*

*Invalidating signals and a pointer to the requesting processor are forwarded to all processors that have a copy of the block. Each invalidated cache sends an acknowledgment to the requesting processor. After the invalidation is complete, only the writing processor will have a cache with a copy of the block.*

**Virtualization and Networks**

Q1. What is the maximum degree of the switches in 4x4 mesh network?

Q2. Why is hypervisor type 2 less efficient than type 1?

Q3. What are events in Xen?

1. Virtual communication channels
2. Virtual interrupts
3. Calls to virtual grant tables

**Interconnection network delays**

Q1. Consider 4x4 mesh network. Switching strategy is store-and-forward. The network clock frequency is 1 GHz and the phit size is 8 bits. The routing address occupies 2 phits (header of the packet), a payload of a packet is 128 bits. Assume that the sender and receiver overheads are zero and that wire delay is also zero. End-to-end network latency for sending a packet from the lower left to the upper right corner in nanoseconds is [1]. The minimum size buffer in each switch in bytes is [2].

**Interconnection network topologies**

Q1. Consider simple comparison between 16x16 Omega network and 16x16 crossbar network. While the crossbar uses cross points, the Omega network is using 2x2 switching elements (SE). Assume that the cost of the SE is four times that of a cross point. How many times is crossbar network more expensive than Omega network if we assume that the cost of the Omega network is determined only by its switching elements and the cost of the crossbar network is determined only by its cross points.

Q2. How many processor nodes Np and how many switches Ns are there in a 4-ary tree (4-ary means that each intermediate node has 4 children) which height is 3 (3 levels of links and 4 levels of nodes)?

1. Np=64

Ns=63

1. Np=64

Ns=21

1. Np=8

Ns=7

1. Np=64

Ns=17

**Networks and Synchronization**

Q1. If reducing the amount of memories and buffers in a system-on-a-chip is a main design goal, which switching technique would you use?

1. circuit switching
2. virtual cut through
3. wormhole

Q2. What techniques/transactions in bus-based networks can be used where fast and slow devices need to communicate?

1. Pipelined
2. Split-transactions
3. Burst-mode

Q3. Barrier is used to provide

1. Mutual exclusion
2. global synchronization among all threads
3. point-to-point synchronization

Q4. Diameter of 4x4 2D mesh, where each node is also connected to additional 3 nodes in the form of a linear array, is:

**Synchronization**

Q1. In case the Store Conditional (SC) instruction fails, bus will not be accessed. How is that achieved? Describe what component needs to be added and what actions need to be performed in order for LL-SC to work.

Q2. Test\_and\_test&set causes larger number of invalidations in MSI invalidate protocol than test&set.

1. True
2. False

**Multithreading**

Q1. What is the common switching delay in RISC-based blocked multithreading processors?

1. Processor switches threads every cycle.
2. The delay corresponds to pipeline depth.
3. The delay corresponds to L1 cache access latency.

Q2. Select resources that need to be replicated to support blocking multithreading implementation.

1. Cache
2. ALU
3. Register file
4. Program counter

**Multithreading1**

Q1. Will context switching take longer for the OoO superscalar processor then in-order in case of blocked multithreading implementation. Why?

Q2. SMT increases the processor area approximately by

1. 100%
2. 50%
3. 5%