Homework Assignment #5
SAMPLE SOLUTION

Q1. (15 points) Please briefly explain the difference among symmetric multiprocessors (SMP) architecture, distributed shared memory (DSM) multiprocessors architecture, and distributed-memory multicomputers architecture.

Answer: Symmetric multiprocessors (SMP) architecture shares a single memory with uniform memory access latency.

Distributed shared memory (DSM) multiprocessors architecture has memory distributed among processors, but still preserves a shared memory view (which means each processor can address any variable in any memory directly) and has non-uniform memory access latency (accessing variables in local memory is faster than accessing variables in remote memory).

Distributed-memory multicomputers architecture has memory distributed among processors, and does not provide a shared memory view, which means a processor can only address variables in its local memory and cannot address a variable in another processor's memory directly.

Q2. (10 points) What is false sharing? Please give an example to illustrate it.

Answer: False sharing occurs in coherence protocols, e.g., in an invalidation-based coherence protocol. It occurs because the cache keeps a copy of data in the unit of cache line (or cache block), which is often multiple words.

For instance, as shown in the below diagram, assume a cache line is 16 bytes (4 words), which holds variables a, b, c, and d, with 4-bytes (one word) each. Assume P0 (processor 0) writes to variable a, and then P1 (processor 1) reads variable b. They are not "true" sharing, because P0 and P1 access different variables. However, since data is cached in the unit of cache line, P0 generates an invalidation and invalidates P1's cached copy of the entire cache line, which causes a coherence miss for P1. Such a scenario is called false sharing and the coherence miss generated is called false sharing miss. In such false sharing miss, the cache line is shared, but no word in the cache line is actually shared (thus "false" sharing), and the miss would not occur if the block size were a single word.

a	b	c	d

One cache line (16 bytes) with 4 words, a, b, c, and d.

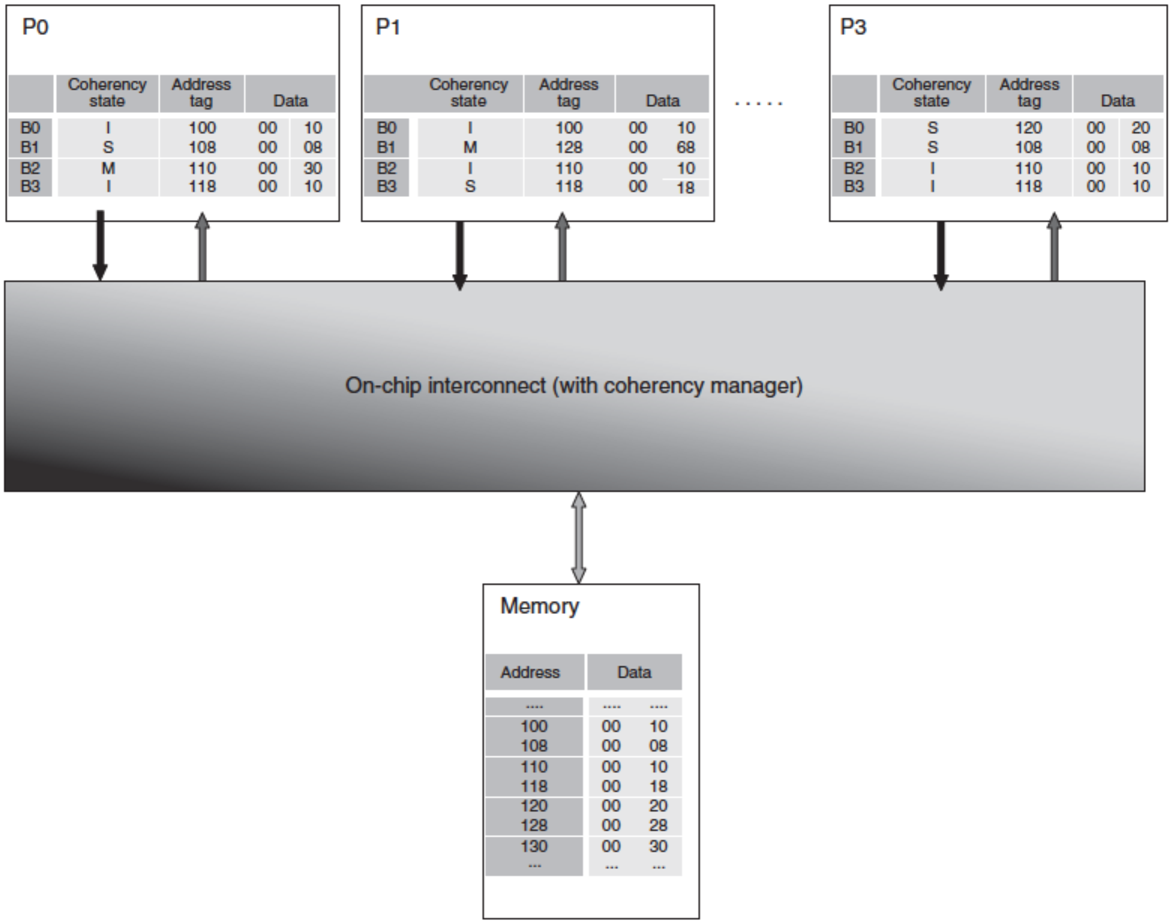
Q3. **(20 points)** The simple, multicore multiprocessor illustrated in the below figure represents a commonly implemented symmetric shared-memory architecture. Each processor has a single, private cache with coherence maintained using the snooping coherence invalidate protocol. Each cache is direct-mapped, with four blocks each holding two words. To simplify the illustration, the cache-address tag contains the full address, and each word shows only two hex characters, with the least significant word on the right. The coherence states are denoted M, S, and I (Modified or Dirty, Shared, and Invalid).

Assume the initial cache and memory state as illustrated in Figure 1. Assume a sequence of one or more CPU operations of the form:

```
P#: <op> <address> [<value>]
```

where P# designates the CPU (e.g., P0), <op> is the CPU operation (e.g., read or write), <address> denotes the memory address, and <value> indicates the new word to be assigned on a write operation. Treat each action below as independently applied to the initial state as given in this figure. What is the resulting state (i.e., coherence state, tags, and data) of the caches and memory after the given action? Show only the blocks that change; for example, P0.B0: (I, 120, 00 01) indicates that CPU P0 's block B0 has the final state of I, tag of 120, and data words 00 and 01. Also, what value is returned by each read operation?

a. P0: read 120 (note address 120 hashed to block B0 for all processors).
b. P0: write 120 <-- 80
c. P3: write 120 <-- 80
d. P1: read 110
e. P0: write 108 <-- 48



P0

	Coherency state	Address tag	Data	
B0	I	100	00	10
B1	S	108	00	08
B2	M	110	00	30
B3	I	118	00	10

P1

	Coherency state	Address tag	Data	
B0	I	100	00	10
B1	M	128	00	68
B2	I	110	00	10
B3	S	118	00	18

P3

	Coherency state	Address tag	Data	
B0	S	120	00	20
B1	S	108	00	08
B2	I	110	00	10
B3	I	118	00	10

Memory

Address	Data	
....	....	....
100	00	10
108	00	08
110	00	10
118	00	18
120	00	20
128	00	28
130	00	30
...	...	...

Figure 1. Multicore multiprocessor.

Answer:

a. P0.B0: (S, 120, 0020), returns 0020
b. P0.B0: (M, 120, 0080)
   P3.B0: (I, 120, 0020)
c. P3.B0: (M, 120, 0080)
   P0.B0: (I, 120, 0080)
d. P1.B2: (S, 110, 0030) returns 0030
   P0.B2: (S, 110, 0030)
e. P0.B1: (M, 108, 0048)
   P3.B1: (I, 108, 0008)

Q4. (10 points) Please name one advantage of the directory based cache coherence systems over the snoopy cache systems.

Answer: In snoopy cache systems, each coherence operation is sent to all processors. One advantage of the directory based cache coherence systems over the snoopy cache systems is that coherence requests are only sent to those processors that need to be notified, since the directory maintains a presence vector (which processor has a copy of it) for each data item (cache line) along with its state.

Q5. (10 points) What are the two main types of hardware threading? Please briefly explain the main difference between them.

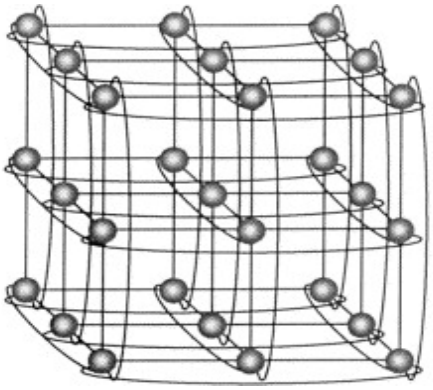
Answer: The two types are "temporal multithreading" and "simultaneous multithreading". The main difference being that in temporal multithreading only a single thread has access to the pipeline resources at a given time. Simultaneous multithreading permits multiple threads to access the pipeline concurrently.

Q6. (10 points) What is a "pipeline flush" and when does it occur?

Answer: A pipeline flush occurs when during a context switch between threads. The state of any executing instructions in the pipeline is cleared in order to avoid poisoning the execution state for an adjacent thread.

Q7. (10 points) Please draw a diagram to show what the 3-D torus interconnect topology with 27 nodes looks like.

Answer: Please see below.



Q8. (15 points) A ***mesh of trees*** is a network that imposes a tree interconnection on a grid of processing nodes. A $\sqrt{p} \times \sqrt{p}$ mesh of trees is constructed as follows. Starting with a $\sqrt{p} \times \sqrt{p}$ grid, a complete binary tree is imposed on each row of the grid. Then a complete binary tree is imposed on each column of the grid. The below figure illustrates the construction of a 4 x 4 mesh of trees. Assume that the nodes at intermediate levels are switching nodes. Please determine the bisection width, diameter, and total number of switching nodes in a $\sqrt{p} \times \sqrt{p}$ mesh. Dividing in to two equal parts

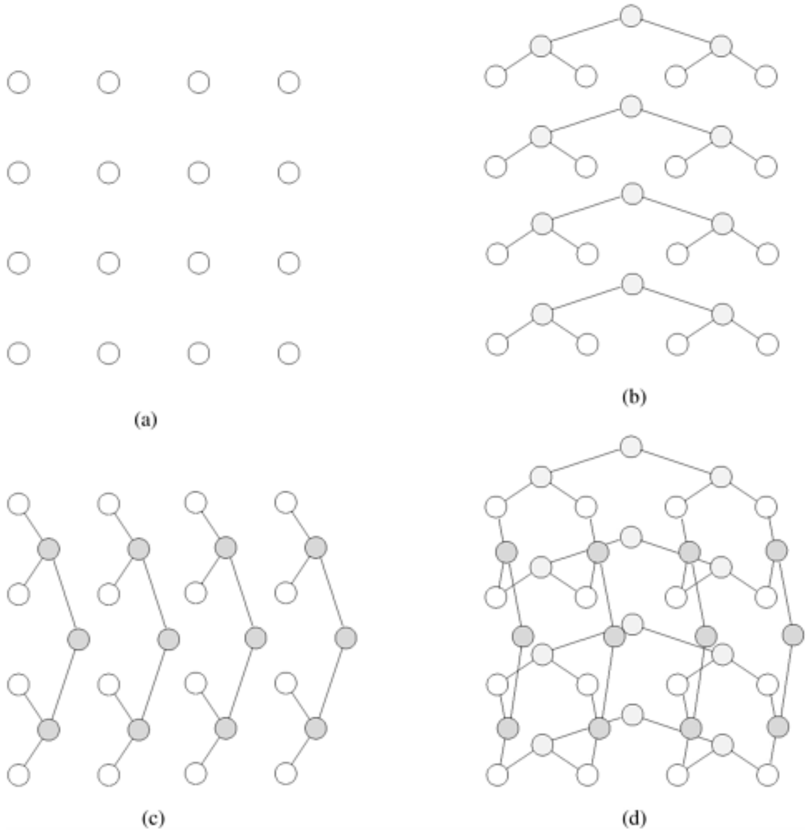


Figure 2. The construction of a 4 x 4 mesh of trees: (a) a 4 x 4 grid, (b) complete binary trees

imposed over each row (nodes added are switching nodes), (c) complete binary trees imposed over each column (nodes added are switching nodes), and (d) the complete 4 x 4 mesh of trees.

diving into two equal parts

Answer: Partitioning the mesh into two equal parts of *p/2* processors each would leave at least $\sqrt{p}$ communication links between the partitions. Therefore, the bisection width of a mesh of trees is $\sqrt{p}$.

Largest path

The processors at the two extremities of the mesh of trees require the largest number of communication links to communicate. This is given by 2log ($\sqrt{p}$) + 2log ($\sqrt{p}$), or 2 log p.

A complete binary tree is imposed on each row and each column of the mesh of trees. There are $2\sqrt{p}$ such rows and columns. Each such tree has $\sqrt{p} - 1$ switches. Therefore, the total number of switches is given by $2\sqrt{p}(\sqrt{p} - 1)$, or $2(p-\sqrt{p})$.

THE END.

switch nodes = number of nodes in tree which are not procssing nodes(last nodes).

4-1 = 3 for each row.