

16.482 / 16.561: Computer Architecture and Design

Fall 2014

Homework #9

Due **Thursday, 11/20/14**

Notes:

- While typed submissions are preferred, handwritten submissions are acceptable.
- Any electronic submission must be in a single file. Archive files will not be accepted.
- Electronic submissions should be e-mailed to Dr. Geiger at Michael.Geiger@uml.edu.
- This assignment is worth a total of 100 points.

1. **RAID** (40 points) You are working with a 5-disk RAID array that contains a total of 15 sectors; the exact sector configuration depends on the RAID level used. In all cases, twelve of the fifteen sectors (S0-S11) will hold data, while the remaining three sectors (P0-P2) hold parity information. Large reads and writes (reads/writes that access an entire stripe in the array) take 300 ms, small reads (reads involving only a single disk) take 150 ms, and small writes (writes involving 1 data disk + 1 parity disk) take 200 ms.

Given the following sequence of sector reads and writes, determine the time required if the array is configured with RAID 3, RAID 4, and RAID 5. Assume the following:

- Requests are queued in such a manner that two consecutive operations may proceed simultaneously if they do not share any disk within the array.
- If two disks, D_x and D_y , are in use, and the access to D_x finishes before the access to D_y , a new operation may start immediately assuming it does not involve D_y .
- Multiple accesses to the same stripe may overlap if they don't use the same disk.

1. read S0
2. write S5
3. write S8
4. read S9
5. read S3
6. write S7
7. read S11
8. write S1

In each case, show the organization of the array to support your answer.

1. Snooping protocols (30 points) You are given a four-processor system that uses a write-invalidate, snooping coherence protocol. Each direct-mapped, write-back cache has four lines, each of which holds eight bytes; in the diagram below, only the least-significant byte of each word is shown. The cache states are I (invalid), S (shared), and M (modified/exclusive). The caches and memory have the following initial state:

P0				
	State	Tag	Data	
B0	I	0x100	01	23
B1	S	0x108	00	88
B2	M	0x110	00	30
B3	I	0x118	00	10

P1				
	State	Tag	Data	
B0	I	0x100	01	23
B1	M	0x128	AB	CD
B2	S	0x130	14	12
B3	S	0x118	14	92

P2				
	State	Tag	Data	
B0	S	0x120	13	31
B1	S	0x108	00	88
B2	I	0x130	51	55
B3	I	0x138	01	38

P3				
	State	Tag	Data	
B0	S	0x120	13	31
B1	S	0x108	00	88
B2	I	0x110	00	30
B3	S	0x118	14	92

Memory		
Address	Data	
0x100	00	00
0x108	00	88
0x110	20	08
0x118	14	92
0x120	13	31
0x128	FF	FE
0x130	14	12
0x138	AB	BA

For each of the transactions listed below, list all cache blocks modified and their final state, as well as all memory blocks modified and their final state. Assume each set of transactions starts with the same initial state—in other words, your answer to part (b) does not depend on your answer to part (a). However, you should track the state transitions of each block throughout the problem.

- | | |
|--|--|
| <p>a. P1: read 0x128
 P2: read 0x128
 P3: read 0x128</p> | <p>c. P3: write 0x128 ← 45
 P0: write 0x128 ← 67
 P3: read 0x12C</p> |
| <p>b. P2: write 0x110 ← 14
 P2: write 0x114 ← DF
 P2: write 0x130 ← 20</p> | |

2. Directory protocols (30 points) Say we have a four-processor system that uses a write-invalidate, directory coherence protocol. The system contains a total of 8 memory blocks, as shown in the initial directory state below:

Block #	P0	P1	P2	P3	Dirty
0	1	1	0	0	0
1	0	0	1	0	1
2	1	1	1	1	0
3	0	1	1	0	0
4	0	0	0	1	1
5	0	0	0	0	0
6	0	1	0	0	0
7	1	0	0	0	1
8	0	0	0	0	0

- a. (5 points) What state is each block in?
- b. (25 points) What messages are sent between the nodes and directory for each of the following transactions to ensure the processor has the most up-to-date block copy, the appropriate invalidations are made, and the directory holds the appropriate state?
- i. P1: write block 0
 - ii. P2: read block 7
 - iii. P3: write block 4
 - iv. P0: write block 2
 - v. P1: read block 6