



# CS 2200 Spring 2006 Final Section B - SOLN

Name: \_\_\_\_\_ Kishore \_\_\_\_\_ GT Number: gt \_\_\_\_\_

## Network protocols

2. (15 points, 20 mins)

You are given the following:

Message size	=	1900 Kbits
Header size per packet	=	1000 bits
Packet size	=	20 Kbits
Bandwidth on the wire	=	400,000 bits/sec
Time of flight	=	2 secs
Window size	=	10
Sender overhead	=	0
Receiver overhead	=	0
Size of ACK message	=	negligible (take it as 0)

- (a) (5 points) Assuming a 10% packet errors on DATA packets (no errors on ACK packets, and no lost packets), how many total DATA packets are transmitted by the sender to accomplish the above message delivery?

$$\text{PKT size} = \text{header size} + \text{payload} \quad (-3 \text{ if header not accounted})$$

$$20000 = 1000 + \text{payload}$$

$$\text{Payload in a packet} = 19000 \text{ bits}$$

$$\text{Number of packets needed to send the message} = 1900000 / 19000 = 100$$

$$\text{With 10\% packet loss the total number of DATA packets} = 100 + 10 + 1 = 111$$

- (b) (10 points) Assuming an error free network, compute the total time to accomplish the above message delivery. You must show your work (such as timing diagram) to get partial credit.

Time per packet:

$$\text{Sender overhead} + \text{packet time} + \text{time of flight} + \text{receive overhead}$$

$$= 0 + 20000/400000 \text{ secs} + 2 \text{ secs} + 0$$

Ack for this packet takes 2 secs after reception on the receiving end.

Timing diagram:



$$\leftarrow \text{-----} 4.05 \text{-----} \rightarrow$$

In one 4.05 secs duty cycle, the sender sends 10 packets and then waits for an ACK packet before sending the 11<sup>th</sup> packet.

To send 100 packets, we need 10 such duty cycles. So the time to send all 100 DATA packets = 4.05 \* 10 = 40.5 secs

However, (window-size - 1) = 9 ACKs are yet to arrive when all the data packets have been sent.

(-3 if ACKS not accounted)

$$\text{Time to receive these ACK packets} = 0.05 * 9 = 0.45 \text{ secs}$$

So,

$$\text{total time to accomplish the above message delivery} = 40.5 + 0.45 = 40.95 \text{ secs}$$

# CS 2200 Spring 2006 Final Section B - SOLN

Name: \_\_\_\_\_ Kishore \_\_\_\_\_ GT Number: gt \_\_\_\_\_

## File system (i-nodes)

3. (15 points, 20 mins)

- (a) (10 points) Given the following commands pictorially show the i-nodes and their contents. You can fabricate disk block addresses for the i-nodes to make the pictures simpler. Where relevant show the reference count for the i-nodes.

```
touch /tmp/foo
mkdir /tmp/bar
mkdir /tmp/bar/gag
ln /tmp/foo /tmp/bar/foo2
ln -s /tmp/foo /tmp/bar/foo
ln /tmp/foo /tmp/bar/gag/foo
ln -s /tmp/bar /tmp/bar/gag/bar
ln -s /tmp /tmp/bar/gag/tmp
```

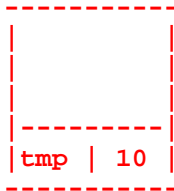
Note:

mkdir creates a directory; touch creates a zero

byte file; ln is link command (-s denotes symbolic link).

Assume that the above files and directories are the only ones in the file system.

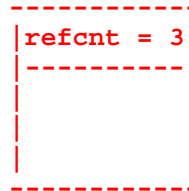
address 0  
i-node for /



address 10  
i-node for /tmp



address 100  
i-node for /tmp/foo



(-2 for each incorrect i-node)

address 20  
i-node for  
/tmp/bar



address 30  
i-node for  
/tmp/bar/gag



# CS 2200 Spring 2006 Final Section B - SOLN

Name: \_\_\_\_\_ Kishore \_\_\_\_\_ GT Number: gt \_\_\_\_\_

- (b) (5 points) If we now execute the command:  
    rm /tmp/bar/foo  
Show the new contents of the i-nodes. (SHOW ONLY THE AFFECTED I-NODES).

Only i-node for /tmp/bar changes as shown below:

(-3 if mostly incorrect)

```
address 20
i-node for
/tmp/bar
-----
|                                     |
|-----|
|foo2|100|
|-----|
|  gag| 30 |
|-----|
```

## File system (user level)

4. (5 points, 5 mins)

Given the following:

Number of cylinders on the disk	=	10,000
Number of platters	=	10
Number of surfaces per platter	=	2
Number of sectors per track	=	128
Number of bytes per sector	=	256
Disk allocation policy	=	contiguous cylinders

- (a) How many cylinders should be allocated to store a file of size 3 Mbyte?

Capacity in 1 cylinder =  $10 * 2 * 128 * 256 = 10 * 2^{16}$  bytes

Number of cylinders to host a 3 MB file =  $\text{CEIL}((3 * 2^{20}) / (10 * 2^{16})) = 5$   
(-3 If incorrect)

- (b) How much is the internal fragmentation caused by this allocation?

Internal fragmentation = 5 cylinders - 3 MB  
=  $3276800 - 3145728 = 131072$  bytes

(-2 if incorrect)

# CS 2200 Spring 2006 Final Section B - SOLN

Name: \_\_\_\_\_ Kishore \_\_\_\_\_ GT Number: gt \_\_\_\_\_

## Disk scheduling

5. (9 points, 10 mins)

Given the following:

Total number of cylinders in the disk	=	200
Current head position	=	cylinder 23
Current requests in order of arrival	=	20, 17, 55, 35, 25, 78, 99

(a) Show the schedule for C-LOOK for the above requests

**25, 35, 55, 78, 99, 17, 20**

**(-3 if incorrect)**

(b) Show the schedule for SSTF

**25, 20, 17, 35, 55, 78, 99**

**(-3 if incorrect)**

(c) Show the schedule for LOOK

**25, 35, 55, 78, 99, 20, 17**

**(-3 if incorrect)**

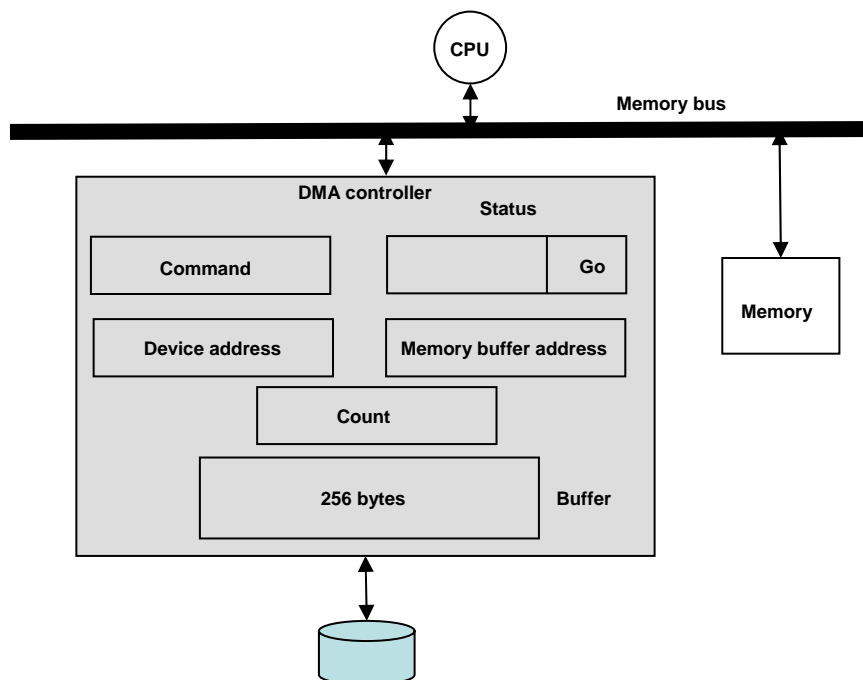
# CS 2200 Spring 2006 Final Section B - SOLN

Name: \_\_\_\_\_ Kishore \_\_\_\_\_ GT Number: gt \_\_\_\_\_

I/O

6. (5 points, 10 mins)

Given the following device controller:



Write the sequence of instructions (pseudo instructions will suffice, need not be actual assembly instructions) that the CPU has to execute to move  $N$  bytes of data starting from device address  $D$  to memory address starting at  $M$ .

1. Store  $N$  into the Count register.
2. Store  $M$  into the Memory buffer address register.
3. Store  $D$  into the Device address register.
4. Store READ from the device command into the Command register.
5. Set the Go bit in the Status register.

(-1 for each missed or misplaced step)

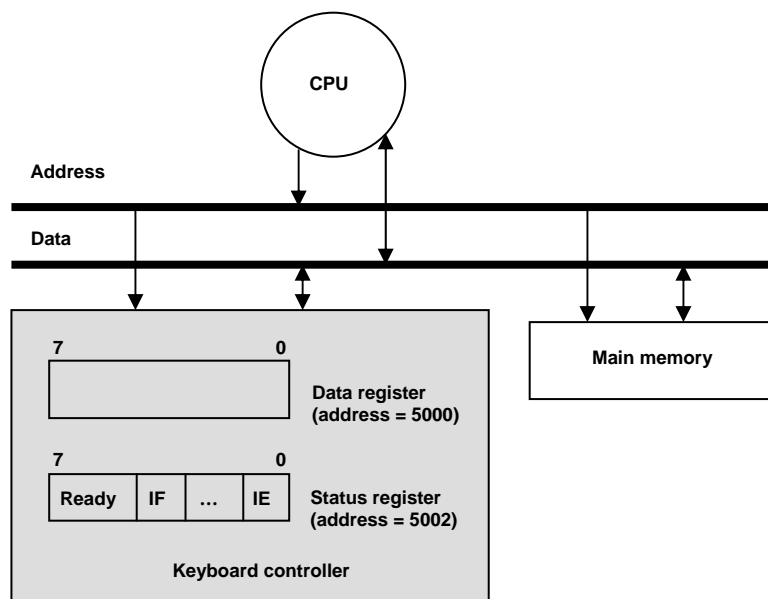
# CS 2200 Spring 2006 Final Section B - SOLN

Name: \_\_\_\_\_ Kishore \_\_\_\_\_ GT Number: gt \_\_\_\_\_

I/O

7. (5 points, 10 mins)

Give the following device controller:



Write the sequence of instructions (pseudo instructions will suffice, need not be actual assembly instructions) that the CPU has to execute to read a character from the device and store to some memory location M.

1. Check the ready bit.
2. If not set go to step 1.
3. Read the contents of the data register into a CPU register.
4. Store the character (from CPU register) into memory.

(-1 for each missed or misplaced step)

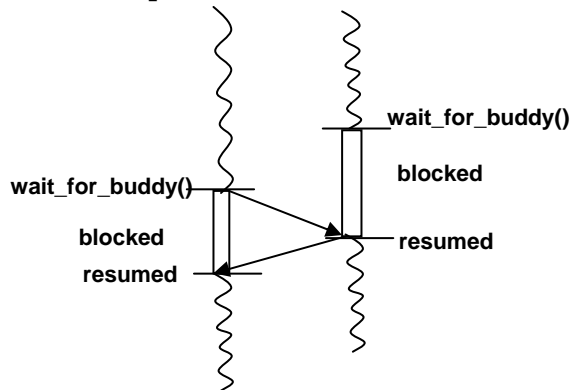
# CS 2200 Spring 2006 Final Section B - SOLN

Name: \_\_\_\_\_ Kishore \_\_\_\_\_ GT Number: gt \_\_\_\_\_

## Threads Rendezvous

8. (10 points, 20 mins)

The function `wait_for_buddy()` is used by EXACTLY 2 threads to rendezvous with each other as shown in the figure below. The order of arrival of the two thread should be immaterial. At the indicate places in the function, insert code that will accomplish the intended rendezvous.



```
boolean buddy_waiting = FALSE;
mutex_lock mtx; /* assume this has been initialized properly */
cond_var cond; /* assume this has been initialized properly */
```

```
wait_for_buddy()
{
    lock(mtx);

    if (buddy_waiting == FALSE) {
        buddy_waiting = TRUE;
        wait (cond, mtx);
        signal(cond);
    }
    else {
        buddy_waiting = FALSE;
        signal (cond);
        wait (cond, mtx);
    }

    unlock (mtx);
}
```

(-2 for each missed or misplaced fix)

# CS 2200 Spring 2006 Final Section B - SOLN

Name: \_\_\_\_\_ Kishore \_\_\_\_\_ GT Number: gt \_\_\_\_\_

## Multiprocessor Cache Coherence

9. (10 points, 10 mins)

Given the following details about an SMP (symmetric multiprocessor):

Cache coherence protocol: **write-invalidate**  
 Cache to memory policy: **write-back**

Initially:

The caches are empty

Memory locations:

**A contains 10**

**B contains 5**

Consider the following timeline of memory accesses from processors P1, P2, and P3.

Time (in increasing order)	Processor P1	Processor P2	Processor P3
T1	Load A		
T2		Load A	
T3			Load A
T4		Store #40, A	
T5	Store #30, B		

Fill the table below. We have started it off for you at time T1.

(**I indicates the cache location is invalid. NP indicates not present**)

Time	Variables	Cache of P1	Cache of P2	Cache of P3	Memory
T1	A	10	NP	NP	10
T2	<b>A</b>	<b>10</b>	<b>10</b>	<b>NP</b>	<b>10</b>
T3	<b>A</b>	<b>10</b>	<b>10</b>	<b>10</b>	<b>10</b>
T4	<b>A</b>	<b>I</b>	<b>40</b>	<b>I</b>	<b>10</b>
T5	<b>A</b>	<b>I</b>	<b>40</b>	<b>I</b>	<b>10</b>
	<b>B</b>	<b>30</b>	<b>NP</b>	<b>NP</b>	<b>5</b>

**(-2 for each incorrect row)**

# CS 2200 Spring 2006 Final Section B - SOLN

Name: \_\_\_\_\_ Kishore \_\_\_\_\_ GT Number: gt \_\_\_\_\_

10. (5 points, 5 mins)

39

0

Given the following:

Virtual address                      40 bits

Physical address                      24 bits

23

0

Page size                              8 K Bytes

A direct mapped TLB with 256 entries

Answer the following:

(a) The number of tag bits per TLB entry is \_\_\_\_\_ **19** \_\_\_\_\_  
(show your work here)

**With 8 KByte page (13 bits) the VPN is 27 bits.**

**(-3 if incorrect)**

**The TLB needs 8 bits to address (256 entries).**

**The number of tag bits =  $27 - 8 = 19$  bits**

(b) The number of bits per TLB entry to store page frame address is   **11**    
(show your work here)

**With 8 KByte page (13 bits) the PFN is  $24 - 13 = 11$  bits**

**(-2 if incorrect)**

# CS 2200 Spring 2006 Final Section B - SOLN

Name: \_\_\_\_\_ Kishore \_\_\_\_\_ GT Number: gt \_\_\_\_\_

11. (10 points, 15 mins)

Given the following:

Total number of blocks in a 2-way set associative cache: 8

	C1	C2
0		
1		
2		
3		

Given the following memory accesses shown in the table below, indicate the cache which will host the memory location and the specific cache index where it will be hosted, and in case of a miss the type of miss (cold/compulsory, capacity, conflict). To help you along we have filled out the first couple of memory accesses.

Memory location	C1	C2	Hit/miss	Type of miss
0	Index = 0		Miss	Cold/compulsory
16		Index = 0	Miss	Cold/compulsory
1	Index = 1		Miss	Cold/compulsory
2	Index = 2		Miss	-do-
0	Index = 0		Hit	
8		Index = 0	Miss	Conflict
0	Index = 0		Hit	
8		Index = 0	Hit	
16	Index = 0		Miss	Conflict
0		Index = 0	Miss	Conflict
1	Index = 1		Hit	
2	Index = 2		Miss	Cold/compulsory
3	Index = 3		Miss	Cold/compulsory
4	Index = 0		Miss	Cold/compulsory
5		Index = 1	Miss	Cold/compulsory
6		Index = 2	Miss	Cold/compulsory
7		Index = 3	Miss	Cold/compulsory
16	Index = 0		Miss	Capacity

(-0.5 for each incorrect item)

# CS 2200 Spring 2006 Final Section B - SOLN

Name: \_\_\_\_\_ Kishore \_\_\_\_\_ GT Number: gt \_\_\_\_\_

12. (10 points, 10 mins)

(a) (5 points)

$I_1: R1 \leftarrow R2 + R3$   
 $I_2: R4 \leftarrow R1 + R5$



If  $I_2$  is immediately following  $I_1$  in the pipeline with no forwarding it will result in

\_\_\_\_\_ Zero bubbles

\_\_\_\_\_ One bubble

\_\_\_\_\_ Two bubble

**X** Three bubbles

\_\_\_\_\_ Four bubbles

(-5 if incorrect for part a and b each)

(b) (5 points)

Branch target buffer is

\_\_\_\_\_ An area of memory reserved for branch instructions

**X** A hardware device that keeps the outcome and target addresses of recent branches encountered during the program execution

\_\_\_\_\_ A hardware device that is pre-loaded before the program starts with the expected outcome and the target addresses of the branches in the program

\_\_\_\_\_ An extra stage in the pipeline for efficient handling of control hazards