2015/Fall CS510 Computer Architecture PhD Qualifying Exam

Student ID _____ Name _____

1. [10pts] Suppose the width of physical address of a system is 30 bits (the maximum physical address space = 2^30B). Calculate the size of storage (in Bits) for the tags for the following cache configurations. Each tag needs extra 2 bits for valid and modified bits. Ignore extra bits for LRU replacement for set-associative caches. You must show how you calculate the final answer (possibly with figures)

   A. 32KB direct-mapped cache, block size = 32B
   B. 64KB 4-way associative cache, block size = 32B
   C. 1KB fully associative cache, block size = 128B

2. [10pts] Cache Replacement Policy.
   A. Explain the LRU replacement policy for associative caches.
   B. Show an example sequence of memory accesses which always miss a 4-way associative cache. Discuss what replacement policy can minimize cache misses for such a cache.

3. [5pts, Multi-threading] Discuss the differences between HW multi-threading and multi-cores. Discuss three components which must be replicated to support multiple threads, and other three components which can be shared in multi-threading.

4. [5pts, SIMD vs VLIW] Discuss the differences in SIMD and VLIW. Explain how the two approaches can improve system performance.

5. [20pts] Dependency and scheduling

```
X0:  lw   r6, 0(r10)
X1:  add r1, r2, r3
X2:  or   r4, r1, r2
X3:  sub r2, r5, r6
X4:  sw   r2, 0(r6)
X5:  and r2, r5, r7
X6:  lw   r4, 0(r2)
```

   A. Find all possible dependencies

   B. What is the minimum number of cycles to run the instructions with an ideal out-of-order processor? The execution of an instruction takes one cycle, and the unlimited number of instructions can be executed in parallel, if the instructions do not have dependencies. However, register renaming is not supported.

   C. What is the minimum number of cycles if unlimited register renaming is supported? Use the same assumption as 5-B.

   D. With out-of-order execution capability, X6 can be ready to be executed while X4 is not ready for execution. Explain when an incorrect out-of-order execution can occur regarding the two instructions.

6. [10pts] The following code fragment is the lock acquire part of a spin lock implementation. It spins until a variable (pointed by R1) becomes zero, and if the variable is zero, it is locked by setting it to 1.

```
          li R2,#1
lockit:   lw R3,0(R1)
          bnez R3,lockit
          sw R2,0(R1)
```

A. Explain why the code fragment does not implement a spin lock correctly. Describe a scenario when the above implementation can allow multiple threads to enter the critical section simultaneously.

B. How will you rewrite the code, if a new instruction swap. “swap Rx, 0(Ry)” read the content of memory[Ry] into Rx, and store the content of Rx to the memory location atomically.

6. [20pts] Multi-processors

A. Discuss the pros and cons of updated-based coherence protocols compared to invalidation-based protocols. You may explain why invalidation-based protocols are dominantly used in commercial multi-processors, and also discuss on what conditions updated-based protocols can excel invalidation-based ones.

B. In multi-processors with invalidation-based coherence protocols, a cache block can be invalidated by another core, which is about to update the cache block. In such a system, is it possible that a cache can hold a correct value for a word (4B), even if the cache block containing the word has recently been invalided?

C. Discuss the pros and cons of shared L2 caches, compare to private L2 caches. Assume that the total cache capacity on a chip does not change. Explain why the current commercial multi-cores commonly have shared L3 caches, instead of shared L2 caches.

7. [20pts] The complete following table to implement a simple MSI protocol for snoop-based coherence

State	CPU		Snoop transition (by Bus command)	
	Command	Next state (Action)	Read	Next state (Action)
Invalid	Read	Shared (BusRead)	BusRead	Invalid
	Write	Modified (BusRfo)	BusRfo	Invalid
			BusUp	Invalid
Shared	Read	1)_____	BusRead	2)_____
	Write	3)_____ (_____)	BusRfo	Invalid
			BusUp	Invalid
Modified	Read	4)_____	BusRead	5)_____ (WriteBack)
	Write	Modified	BusRfo	Invalid (WriteBack)
			BusUp	Error

A. Complete the MSI protocol.

B. Explain why "Modified" state cannot receive "BusUp" command. ("Error" state in the table)

C. Extend the protocol to add Exclusive state. In Exclusive state, the cache block is clean (not modified), but no other caches have a copy of the cache block. Explain why the addition of exclusive state is beneficial by describing a scenario. You may specify extra conditions, if necessary.

State	CPU		Snoop transition (by Bus command)	
	Command	Next state (Action)	Read	Next state and Action
Invalid	Read	1)_____	BusRead	Invalid
	Write	Modified (BusRfo)	BusRfo	Invalid
			BusUp	Invalid
Shared	Read	[same as MSI]	BusRead	[same as MSI]
	Write	[same as MSI]	BusRfo	Invalid
			BusUp	Invalid
Modified	Read	[same as MSI]	BusRead	[same as MSI] (WriteBack)
	Write	Modified	BusRfo	Invalid (WriteBack)
			BusUp	Error
Exclusive	Read	2)_____	BusRead	3)_____
	Write	4)_____	BusRfo	5)_____
			BusUp	Error