2011/Spring Computer Architecture PhD Qualifying Exam

Student ID _____ Name _____

1. Bob is designing a processor with 5-stage pipeline : Fetch (IF) → Decode (DE) → Execution (EX) → Memory (MEM) → Writeback (WB). The following table shows the latency for each stages

IF	DE	EX	MEM	WB
50ps	50ps	100ps	200ps	100ps

However, Fred is not very happy with the performance of Bob's processor and decides to make some changes. For the following changes, please state whether it will improve performance, no impact on performance, or whether it will not work. Please ***explain*** your answer. (Each part is a **different** change – i.e., when answering part (B), you can assume the baseline 5-stage pipeline and ignore (A).)

   A. Fred notices that IF and DE stages can be combined into a single stage and proposes a 4-stage pipeline.
   B. Fred notices that most instructions (i.e., ALU instructions) do not access the memory. As a result, he proposes to remove the MEM stage for all ALU instructions and make those instructions 4 stages, instead of 5 stages.
   C. Fred notices that MEM stage is the longest stage and decides to partition the stage into two stages MEM1 and MEM2.

2. Bob is now focusing on trying to improve the power efficiency of this new processor and is having a hard time. Fred suggests that you can increase the power efficiency by exploiting parallelism and have multiple functional units. However, Bob argues that Fred's suggestion is incorrect since multiple functional units can increase the power consumption. Who do you think is correct, Bob or Fred? Why? Please explain.

3. Please answer the following question on the design of a cache.
   A. Assume the cache size and associativity remain constant. What happens to the size of the tag array when the line size is doubled?
   B. Assume the line size and associativity remain constant. What happens to the compulsory misses when the cache capacity of a direct-mapped is doubled?

4. Virtual address & Cache organization:
   A. What is the benefit of having a virtually-indexed virtually-tagged cache structure?
   B. Fred designs a one level 4-way set associative cache that is virtually indexed and virtually tagged. He realizes that such a cache suffers from a homonym aliasing problem. The homonym problem happens when two processes use the same virtual address to access different physical locations. Can you help Fred out solve this homonym problem?
   C. Another problem with virtually indexed and virtually tagged caches is called the synonym problem. The synonym problem happens when distinct virtual addresses refer to the same physical location. Does your solution in (B) help with the synonym problem as well?

5. Coherence:
   (a) Snoop-based coherence are attractive as it provides fast cache-to-cache transfer and does not require the complexity of maintaining a directory. However, its scalability is limited. What aspect of snoop-based coherence prevents its from being scalable?
   (b) To overcome this limitation, many researchers have proposed "snoop filtering" technique. Can you try to describe what this technique does and how they overcome the scalability problems of (a)?