Department of Computer Science
PhD Qualifying Examination
Computer Architecture
Fall, 1998

1. (15 points)
   Describe the relative advantages and disadvantages of the following network topologies: crossbar, bus, ring, grid, and hypercube.

2. (40 points)
   This question is based on considering the execution of the following fragment of code. Assume the first operand in each instruction is the result (so the first instruction shown adds the contents of register `r1` to the contents of `r2` and puts the result in `r3`).

```
1.  add  r3,   r1,   r2
2.  add  r6,   r4,   r5
3.  add  r7,   r3,   r6
4.  add  r10,  r8,   r9
5.  add  r13,  r11,  r12
6.  add  r14,  r10,  r13
7.  add  r15,  r7,   r14
```

A. Draw a graph showing the data dependencies and antidependencies in the code. *Note: A **dependency** exists between two instructions if the first instruction produces a result that is used by the second instruction. An **antidependency** exists if the first instruction uses an input that will be corrupted if the second instruction finishes first.*

B. Suppose the code were executed on an implementation with a five-stage pipeline (fetch, decode/read registers, alu, memory operation, write register) and as much forwarding as possible between stages. How many cycles would it take to execute, from the fetch of the first instruction to writing the result of the last instruction into its register?

C. Now suppose it were executed on a superscalar implementation able to issue up to 8 instructions on a cycle (subject to dependencies, of course), but otherwise similar to that in Part B. Assume that if an instruction has to be delayed a cycle due to dependencies, all following instructions must also be delayed a cycle. Now how many cycles will it take?

D. If it were executed on an implementation like that in Part C, but with instruction reordering, how many cycles would it take?

3. (20 points)
   Suppose you had a bus-based multiple-processor computer using a snoopy invalidation-based cache coherence protocol. Initially, memory block M is not valid in any cache.

   Describe any changes to the state of the memory block in the different caches after each of the following transactions:

   A. Processor P1 reads M

   B. Processor P2 reads M

   C. Processor P1 writes M

   D. Processor P3 writes M

   E. A DMA device writes M

5. (25 points)
   In a word-addressable computer system, the latency of memory accesses is L and a word can be

transferred between memory and cache in time T (note: the latency L includes the time T required to transfer the first word of a block). A block of memory consists of B words. In order to reduce the penalty due to misses, we can adopt one of two strategies:

Early restart — The block is fetched in order and as soon as the required word arrives, it is sent to CPU.

Critical word first — The required word is fetched first and sent to CPU and the rest of the block is fetched after that.

Assume that data accesses are random and ignore branching effects in programs (i.e. assume that programs do not have branch instructions). How much better can the Critical word first scheme perform when compared to the Early restart scheme for

A. data misses?

B. instruction misses?