CS 252: Spring 00—Quiz 1

Due Monday March 13 at 10am. Either at Reception of 231 Cory or by e-mail to jan@eecs.berkeley.edu

FOR ALL YOUR ANSWERS, BE CONCISE AND TO THE POINT. VERBOSITY DOES NOT GET YOU ANY EXTRA CREDIT.

1. Explain the utility of each of the following DSP architectural features. Demonstrate with the aid of a simple specific example (described in pseudo-code).

- Circular buffers
- Zero-wait loop
- Separate instruction and data memories
- Auto-increment address generation units

2. Name 5 significant differences between general purpose and embedded processors. Give for each of these one example on how this impacts the architectural design process.

3. Consider the following code segment, which is a typical representative of an application domain. Determine which architectural features will most impact the performance, and make an estimate of obtained savings (use some reasonable execution time estimates).

```c
if (a) {
    if (b) {
        if (c) {x=3; } else {x=2;}
    }
    else {
        if (c) {x=2}; else {x=1;}
    }
}
else {
    if (b) {
        if (c) {x=2; } else {x=1;}
    }
    else {
        if (c) {x=1}; else {x=0;}
    }
}
```
4. Give an example of each of the following. Be detailed and explain why.
   a. An application where a VLIW is superior in performance to a superscalar.
   b. An example where a VLIW is superior in performance to a vector processor.
   c. An example where a superscalar is superior in performance to a VLIW.
   d. An example where a vector processor is superior in performance to a VLIW.

5. Describe very concisely the advantages of Tomasulo’s method for dynamic scheduling over the CDC 6600 Scoreboard. Are there any disadvantages relative to the scoreboard?

6. Enumerate 4 reasons why reconfigurable computing can lead to lower energy solutions for programmable embedded applications. Determine also 2 conditions that these application programs have to meet to exploit these opportunities.

7. Consider the following configurable network topologies: bus, crossbar, mesh, and binary tree. Assume that we are trying to connect \( n \) elements to each other. Rank order each of them with respect to the following properties (best first):
   - overall bandwidth
   - maximum latency
   - area (or cost)
   - energy dissipation per transfer

   Use simple models and assume that connection requirements are equally spread amongst all of the elements (i.e. there is no inherent locality in the connections).

   GOOD LUCK!