# Exam 4 May 2013, questions

Computer Architecture (The University of Edinburgh)

UNIVERSITY OF EDINBURGH

COLLEGE OF SCIENCE AND ENGINEERING

SCHOOL OF INFORMATICS

# INFR09009 COMPUTER ARCHITECTURE

**Saturday 4th May 2013**

**09:30 to 11:30**

## INSTRUCTIONS TO CANDIDATES

**Answer any TWO questions.**

**All questions carry equal weight.**

**CALCULATORS MAY NOT BE USED IN THIS EXAMINATION**

Year 3 Courses

Convener: K. Kalorkoti
External Examiners: K. Eder, T. Field

THIS EXAMINATION WILL BE MARKED ANONYMOUSLY

1. (a) On a particular workload, 50% of the overall execution time is spent performing floating point additions. What is the maximum speedup that can be achieved by an enhancement that speeds up floating point additions? [*2 marks*]

   (b) How fast should the enhancement speed up floating point additions in order for the workload to run 1.33 times faster? [*2 marks*]

   (c) Given an out-of-order processor, explain qualitatively how each of the following design changes would affect: (i) the number of instructions executed (IC), (ii) the cycles per instruction (CPI) and (iii) the clock period.

   - Changing process technology from $65nm$ process to $32nm$ process
   - Adding another optimisation to the compiler
   - Doubling the number of reservation stations
   - Doubling the number of architectural registers (those that are visible to the compiler)

   [*12 marks*]

   (d) The three major methods for specifying branch conditions are: (a) condition code (b) condition register and (c) compare and branch. Discuss their relative advantages and disadvantages. [*6 marks*]

   (e) Explain how the computer architect can help the compiler writer while designing an instruction set. [*3 marks*]

2. (a) Identify the three commonly-used categories of cache miss (the 3 Cs), and for each category explain the cause of the miss. [*4 marks*]

   (b) What is prefetching? Explain how it might affect each of the above three categories of cache misses? [*3 marks*]

   (c) Consider a computer system with a first-level data cache with the following characteristics: size: 16KBytes; associativity: direct-mapped; line size: 64Bytes; addressing: physical.

   The system has a separate instruction cache and you can ignore the effect of instruction cache misses in this problem. This system is used to run the following code:

```
for (i=0; i<4096; i++)
  X[i] = X[i] * Y[i] + C
```

   Assume that both X and Y have 4096 elements, each consisting of 4 bytes (single precision floating point). These arrays are allocated consecutively in physical memory. The assembly code generated by a naive compiler is the following:

```
loop: lw f2, 0(r1)       # load X[i]
      lw f4, 0(r2)       # load Y[i]
      multd f2, f2, f4   # perform the multiplication
      addd f2, f2, f0    # perform the addition
      sw 0(r1), f2       # store the new value of X[i]
      addi r1, r1, 4     # update address of X
      addi r2, r2, 4     # update address of Y
      addi r3, r3, 1     # increment loop counter
      bne r3, 4096, loop # branch back if not done
```

   i. How many data cache misses will this code generate? Breakdown your answer into the three types of misses. What is the data cache miss rate? [*6 marks*]

   ii. Provide a software solution that significantly reduces the number of data cache misses. How many data cache misses will your code generate? Breakdown the cache misses into the three types of misses. What is the data cache miss rate? [*6 marks*]

   iii. Provide a hardware solution that significantly reduces the number of data cache misses. You are free to alter the cache organisation. How many data cache misses will your code generate? Breakdown the cache misses into the three types of misses. What is the data cache miss rate? [*6 marks*]

3. (a) What is forwarding? Explain with an example how forwarding minimises stalls due to data hazards in a 5-stage RISC pipeline? [*4 marks*]

   (b) Is forwarding implemented in the CDC 6600 Scoreboard? Justify your answer. [*3 marks*]

   (c) Is forwarding implemented in Tomasulo's algorithm? Justify your answer. [*3 marks*]

   (d) Explain how a static branch predictor is different from a dynamic branch predictor. [*4 marks*]

   (e) Can a static branch predictor ever perform better than a dynamic branch predictor? Justify your answer. [*3 marks*]

   (f) Can a 1-bit dynamic branch predictor ever perform better than a 2-bit dynamic branch predictor? Justify your answer. [*3 marks*]

   (g) What is a Branch Target Buffer? [*2 marks*]

   (h) In the presence of indirect branches, typically the Branch Target Buffer does not perform very well. Explain the reason for this behaviour. [*3 marks*]