Comments on EEE411/6031 January 2010-02-01

General Issues

The major problem encountered by people taking these examinations seems to have been: not reading the question and, thereby, determining what exactly was being asked for. This seems to have extended also to the hints embedded in the questions.

Question 1

This question was extremely well answered – very few people did badly and it was clear that most people had understood the issues and were able to cope with new problems.

Question 2

Conversely, this question was badly answered:

a(i) – a lot of people drew the overall block diagram for a SIMD processor, or the block showing how broadcast is accomplished. The question clearly said ‘schematic diagram for a Processing Element (PE)’. Reading the question is very important.

a(ii) – similarly people tended to give basic factual information about a PE’s memory rather than talking about how single/multiple portedness affects the ability of the PE to execute operations where two operands are read, executed on and written back in a cycle – affecting performance. Again this requirement is stated explicitly in the question along with a hint. I distinctly remember devoting close to half a lecture talking about exactly these issues – but that was probably because I was there!

b. – Some people took the hint given in the question and got it right (or almost right). If you look at the pattern checked for and look at the sums along the rows/columns you get:

	1	1	5	1	1
1	0	0	1	0	0
1	0	0	1	0	0
5	1	1	1	1	1
1	0	0	1	0	0
1	0	0	1	0	0

So, if you add up rows (separately) across 5 columns and add up columns across 5 rows then the values above can only *in combination* come from the correct pattern. After this initial phase, move the added row values up the rows checking for incoming values of 1,1,5,1,1 in that order using CMP x, #1; INCC y (for example). Then move the added column values left along the columns checking for incoming values of 1,1,5,1,1.

Obviously, people solved the problem in different ways and were credited for this work equally.

Question 3

Quite a mixed set of attempts at this question – ranging from perfect to very poor. Nobody had difficulty with Flynn’s Taxonomy but some people got confused with α, which is the proportion of the task that can be parallelised. In a few cases it was confused with degree of parallelism (this is N, the amount of parallelisation available for the proportion, α, that can be parallelised).

In part b(ii), some people provided the generalisation (again book work) where the task is split into a sequence of elemental sections – each of which has its own value of α and N. A few people thought that this part referred to the comparison between Amdahl’s Rule and pipelining – it was not.

Part c was attempted by most people. Some people got it right; a few people worked out the unpipelined time (1630μs) but then did not remove N (it’s 1 for no parallelisation) – this yielded a more difficult and wrong equation. For part c(i), those who got close to the answer quoted N=41. I credited the marks but technically, because N=41.3, the answer should be 42 because the question says 6000 blocks must be processed – if N=41 then fewer blocks (5989) will be processed.

Surprisingly, given the range of foregoing answers, the final part was answered quite well with people recognising where the bottleneck was and deciding how the problem should be addressed.

Question 4

This question, which was more bookwork than the other questions was done reasonably. However, some people missed what was being asked for in part b where a delineation of the problems associated with constructing a practical TLB was required followed by a description of the DIR:PAGE:OFFSET + TLB used for a practical virtual:physical memory translator. Some people also discussed caches here. The question makes it quite clear that it is concerned with the main memory/virtual memory interface.