Two hours

UNIVERSITY OF MANCHESTER
SCHOOL OF COMPUTER SCIENCE

System-on-Chip modelling with System C

Date: Tuesday 24th January 2012
Time: 14:00 - 16:00

Please answer Question ONE and any TWO Questions
from the remaining FOUR Questions provided

For full marks your answers should be concise as well as accurate.
Marks will be awarded for reasoning and method as well as being correct.

This is a CLOSED book examination

The use of electronic calculators is permitted provided they are
not programmable and do not store text.

[PTO]
1. **Compulsory**

   Answer all of the following (2 marks each):

a)  In 1965 Gordon Moore predicted that the number of transistors on a chip would double every 2 years. If a 1965 chip had 100 transistors and Gordon Moore was right, approximately how many transistors would be on a chip in 2011?

b)  State which of the following aspects of a design flow are considered to be technology dependent and which are technology independent: synthesizable processor IP; foundry process technology; cell library; custom Verilog.

c)  What distinguishes a Network-on-Chip from an on-chip bus interconnect?

d)  How can multithreading help a processor tolerate high memory latency?

e)  What is the key technology underpinning reconfigurable computing systems? What is meant by dynamic reconfigurability?

f)  How does Transactional Modelling distinguish between “Computation” and “Communication”?

g)  A video signal needs to access every pixel from its frame store 25 times per second. Each pixel access takes 50 nanoseconds. What is the maximum number of pixels that can be displayed in the frame (if you neglect line and frame blanking intervals)?

h)  Explain how System C is compiled and executed.

i)  Why does System C offer both timed and untimed transaction modelling?

j)  How could you estimate the simulated speed (in simulated clock cycles per “real” elapsed second) of your Timed TLM model in the lab? Approximately (to within 2 orders of magnitude) how fast do you imagine this is?
2.

a) Discuss the relative merits of the following approaches to delivering system functionality in terms of their respective performance and efficiency:

   i) Custom fixed hardware;
   ii) Special-purpose processors, such as a DSP or GPU;
   iii) General-purpose processors;
   iv) Reconfigurable computing systems.

   (2 marks each)

b) Sketch a graph of algorithmic flexibility against area-efficiency showing where each of the above approaches sits relative to the others, stating your reasons for their relative positions.

   (4 marks)

c) Sketch a system diagram that incorporates a CPU with cache and memory bus, showing how a reconfigurable function unit, a reconfigurable coprocessor, a reconfigurable hardware engine and a reconfigurable peripheral might fit into the architecture.

   (4 marks)

d) Give an example of how each of the reconfigurable units in c) might be used in a complex System-on-Chip design.

   (4 marks)

3.

a) Explain why testing and verification is an important consideration in the development of a new SOC.

   (4 Marks)

b) Identify the features of the design of the “drawing engine” you used in the lab that were provided for testing and verification.

   (5 Marks)

c) You carried out testing at the algorithmic level, with untimed transactions, and with a timed transactional model. Which types of tests were most suitable at each level, and why?

   (8 Marks)

d) Explain why it is not necessarily best practice to run all possible tests on the most detailed model that is developed.

   (3 Marks)
4.

a) At a late stage in the development of the SOC that includes your “drawing engine”, the team discovers that there are unlikely to be resources available to implement line drawing or more complex hardware shape generation into the SOC. A team member suggests using a simple pixel-writing interface to the frame buffer, and programming the microprocessor to generate lines and more complex shapes. What significant functional and performance issues is this change likely to cause? (5 Marks)

b) Explain the steps necessary to take your working and tested timed TLM model for the hardware drawing engine (as in the lab), and reusing as much of your modelling work as possible, to address the issues in (a) as far as possible for other members of the design team of the product. (9 marks)

c) At an even later stage in design, a (different) team member suggests simplifying the hardware yet further by asking the microprocessor to generate the video signal output by accessing the frame store. How would you investigate the feasibility of this using Transactional Modelling? (6 Marks)
5.
   a) Explain what “arbitration” and “routing” are, and why they are needed in Transactional Level Modelling of Channels. (4 Marks)
   
   b) Figure 1 shows the TLM code to arbitrate among three initiators. Explain how this code chooses between different initiators that make requests at the same time. (6 Marks)
   
   c) Hence, or otherwise, explain what would happen if all three initiators requested channel access continuously every clock period. (4 Marks)
   
   d) Show a modified version of this code which would result in the CRTC_ID request receiving two out of every three clock periods, with the third being alternately received by the DE_ID and the IQ_ID request. (6 Marks)

```cpp
while (true) {
    /* suspend until next positive edge of the clock */
    wait(clk.posedge_event());
    if (requestIn[CRTC_ID]) {
        grant = 1;
        proceed[CRTC_ID].notify();
        wait(completed);
    } else if (requestIn[DE_ID]) {
        grant = 2;
        proceed[DE_ID].notify();
        wait(completed);
    } else if (requestIn[IQ_ID]) {
        grant = 3;
        proceed[IQ_ID].notify();
        wait(completed);
    } else
        grant = 0;
}
```

END OF EXAMINATION