System-on-Chip modelling with System C

Date: Tuesday 18th January 2011
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.
1. **Compulsory**

Answer all of the following (2 marks each):

a) Explain briefly the difference between algorithmic-level modelling and transaction-level modelling and the language each would use.

b) Explain briefly the difference between timed and untimed transaction-level modelling and the order in which they would be used in the design of a complex System-on-Chip.

c) The Manchester ‘Baby’ machine computed at 700 instructions per second and consumed 3.5kW of electricity. A modern mobile phone microprocessor might compute at 200 MIPS (Millions of Instructions per Second) and consume 20mW. How much more energy-efficient is the modern processor than ‘Baby’?

d) The SpiNNaker chip multiprocessor (CMP) incorporates 18 ARM968 cores each of which has a 32Kbyte instruction SRAM (Static Random Access Memory) and 64Kbytes of data SRAM. Estimate the number of transistors on the CMP assuming that memory dominates the transistor count and the SRAM uses 6 transistors per memory bit.

e) The cost of designing a complex SoC from scratch has become prohibitive. Give two examples of how this cost can be reduced by avoiding designing everything from scratch.

f) What are the possible disadvantages of Transactional Level Modelling?

g) What types of tradeoffs would a systems architect be exploring when using a TLM with untimed computation blocks and untimed channels? With untimed computation blocks and timed channels?

h) What types of tradeoffs would a systems architect be exploring when using a TLM with timed computation blocks and untimed channels? With timed computation blocks and timed channels?

i) A video signal to refresh a screen needs to contain 640 pixels (one scan line) every 25 microseconds. If each pixel is represented by eight bits, what bandwidth is required from a frame store? How could a frame store be composed of static RAM with an access time of 55 nS to achieve this?

j) Why is SystemC designed as an enhancement to C++, rather than as a new language or a Verilog-like language?
2. The specification, design and implementation of a new complex SoC can be described in three phases, each phase addressing a different question. For each of these phases describe the sorts of issues that must be addressed, illustrating your descriptions with examples where appropriate: (5 marks each)

a) What do we want? Defining the target specification.

b) What do we need? Identifying the “kit of parts”.

c) How do we make it? Sourcing the components, and the design flow.

Imagine that you are leading the SoC design group within a mobile phone company. The marketing department has come up with a functional requirement for a new smart-phone chip that you estimate would occupy a 2 cm² die on the process technology you used for the previous chip, a technology that has an average of one manufacturing defect per square centimetre of processed silicon. Discuss your options and how you might respond to the marketing department. (5 marks)

3. a) With the aid of a diagram of the design flow, explain how a design hierarchy assists with the design of a complex chip. (4 marks)

b) What are the general principles for verifying that each level of the design hierarchy is functioning correctly? (4 marks)

c) Considering the line draw function used in the coursework exercises, discuss the tests that it would be suitable to run in order to verify correct operation on the display. (4 marks)

d) At the TLM level, if the display were not operational for some reason, how would you verify whether the line draw function was operating correctly? (3 marks)

e) If the line in the TLM model appeared in the wrong place on the display what measures would you take to trace the problem? (5 marks)
4. a) Explain what is meant by a Transactional Level Model (TLM) and why it is useful. (3 marks)

b) Draw a diagram of the drawing machine TLM you have been designing in the coursework exercise and describe one different way of partitioning the system that you think would be sensible. (5 marks)

c) Using the TLM you drew (in part b), describe every transaction that takes place to draw a line in the frame buffer. (5 marks)

d) Using the same TLM, describe every transaction that takes place while displaying the frame buffer on the screen. (3 marks)

e) By considering the effect on the performance of implementing a drawing function (e.g. circle draw) as a hardware block, or in software, explain how TLM’s can be used to explore tradeoffs between hardware and software. (4 marks)
5. Figure 1 shows the TLM code for “processTransactions”, a method associated with a video channel making access to a frame store in a model similar to your lab exercise.

a) Explain the purpose of this code, and how it is achieved. (6 marks)

b) Explain what type of SystemC feature is “proceed” and what it is used for in the code. (4 marks)

c) With what type of SystemC process (SC_METHOD or SC_THREAD) would you instantiate this code? Explain why. (4 marks)

d) Show, using System C or pseudo code, how you would modify the code to give all requestors fair access to the channel. (6 marks)

```c
void VideoChannel::processTransactions() {
    while (true) {
        /* suspend until next positive edge of the clock */
        wait(clk.posedge_event());
        /* priorities are fixed in this model */
        if (requestIn[CRTC_ID]) {
            grant = 1;
            /* process CRT controller transaction */
            /* give the go ahead */
            proceed[CRTC_ID].notify();
            /* and wait for completion */
            wait(completed);
        }
        else if (requestIn[DE_ID]) {
            grant = 2;
            /* process drawing engine controller transaction */
            /* give the go ahead */
            proceed[DE_ID].notify();
            /* and wait for completion */
            wait(completed);
        }
        else if (requestIn[IQ_ID]) {
            grant = 3;
            /* process inquisitor controller transaction */
            /* give the go ahead */
            proceed[IQ_ID].notify();
            /* and wait for completion */
            wait(completed);
        }
        else
            grant = 0;
    }
}
```

Figure 1
END OF EXAMINATION