Two hours

UNIVERSITY OF MANCHESTER
SCHOOL OF COMPUTER SCIENCE

VLSI System Design

Date: Thursday 28\textsuperscript{th} January 2010
Time: 09.45 – 11.45

Please answer any THREE Questions from the FOUR questions provided

The use of electronic calculators is \textbf{NOT} permitted.
Some questions refer to the 16-bit RISC processor, named STUMP, that you have been designing in the laboratory. Its Register Transfer Level (RTL) design for the datapath is given in Figure 1.
1. a) Identify the features of the RTL diagram of the STUMP processor that identify it as a load/store architecture. (3 marks)

b) Hence, using the RTL diagram, explain how the STUMP processor performs the following operations:

i) register to register
ii) load from memory, and store to memory
iii) branch instructions (9 marks)

c) If the RTL design were to be pipelined, explain why three stages would be appropriate and discuss any modifications to the design that would be required. (4 marks)

d) Given that 5% of instructions are load or store, discuss whether even more throughput could be obtained from adopting a 2-stage rather than a 3-stage pipeline. (4 marks)

2. a) Explain what is meant by

i) an external behavioural model
ii) an internal behavioural model.

Hence explain why an external behavioural model can not usually be synthesised. (5 marks)

b) Figure 2.1 shows the truth table of a 2-to-4 decode block.

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>En</td>
<td>A</td>
</tr>
<tr>
<td>0</td>
<td>X</td>
</tr>
<tr>
<td>1</td>
<td>O</td>
</tr>
<tr>
<td>1</td>
<td>O</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

X = don’t care

Figure 2.1 – Truth table of 2-to-4 Decoder with an Enable

Write a Verilog module to describe the internal behaviour of this block. (7 marks)

(Question 2 continues on the following page)
c) The RTL Verilog code for the state machine in the STUMP processor is given below:

```
reg[1:0] state;
always @(posedge CLK)
bEGIN
  if (RESET == 1) state = 0;
  else if (state == 3) state = 1;
  else state = state + 1;
END
```

Explain the Verilog constructs in this code, in particular the use of the `reg` declaration, the `always` block, the `conditional` statement and blocking assignments. Finally, explain what the code does. (8 marks)

3. a) What is meant by controllability and observability in the context of testing the operation of silicon chips? (3 marks)

b) Hence explain why Reset is an important facility both in simulation and on the fabricated design. (3 marks)

c) Using appropriate diagrams, describe the scan path approach to testability. (6 marks)

d) If a scan path were incorporated throughout the STUMP datapath and control, describe how the effect of a register to register instruction could be monitored on the fabricated processor. (4 marks)

e) What advantages and disadvantages are there between the scan-pathed STUMP of part d) and the testing approach adopted in the laboratory exercises of storing the processor results in a memory? (4 marks)
4. a) With the aid of a diagram, describe what is meant by two-phase non-overlapping clocks and explain why it might be preferable to adopt these in a VLSI design rather than a single phase clock. (4 marks)

b) Draw a diagram of the logic to generate two-phase non-overlapping clocks internally on-chip from a single phase clock input. Explain the operation of the logic, particularly identifying the length of the non-overlap time. (5 marks)

c) How could the non-overlap time of the logic of part b) be increased? (3 marks)

d) Define what is meant by clock skew and describe how it arises. (4 marks)

e) Is it possible that clock skew could make the non-overlap time ‘disappear’ or become negative? If so, what measures can be taken to preserve the non-overlap time. (4 marks)