The Underlying Machine

Friday 25th January 2008  Time: 09:45 – 11:15

Please answer Question ONE and one other Question from either Questions TWO or THREE

The use of electronic calculators is NOT permitted
1. **COMPULSORY**

Answer any ten subsections of this question. Each part is worth 2 marks.

a) Show how a transparent latch can be built using only one type of logic gate (of your choice).

b) Write down the truth table for a 1-bit full adder.

c) Why does a cache memory speed up a ‘typical’ programme’s execution?

d) Explain how DMA can make I/O operations more efficient.

e) How many address pins are required on a 2 M word SRAM chip?

f) DAB (Digital Audio Broadcast) radio transmits a number of different stations simultaneously on the same frequency (or ‘multiplex’). Briefly explain how this is possible.

g) Give two general approaches used to speed up a microprocessor implementation.

h) Give one potential advantage and one potential disadvantage of a Chip Multiprocessor (CMP) over a Uniprocessor system.

i) In a few sentences describe how and why pipelining can speed up a single processor's operation.

j) Sketch the state diagram of a modulo 4 up/down counter.

k) Sketch, and label, a ‘three box’ computer model.

l) Which circuit will occupy more silicon area on a chip – a transparent latch or a D-type flip-flop – and why?
2. a) Show how a multi-bit adder may be constructed assuming you already have a design for a 1-bit full adder.  

   (2 marks)

b) Describe where the ‘critical path’ will be in your multi-bit adder.  

   (2 marks)

c) Show how a multi-bit adder can be modified so it can also perform subtraction if required.  

   (3 marks)

Flags are used on many processors to indicate the status of the result of the last arithmetic or logical operation. ARM is typical, having four flags:

<table>
<thead>
<tr>
<th>Flag</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sign</td>
<td>set if result is negative</td>
</tr>
<tr>
<td>Zero</td>
<td>set if result is zero</td>
</tr>
<tr>
<td>Carry</td>
<td>carry out of the adder</td>
</tr>
<tr>
<td>Overflow</td>
<td>set if a two’s complement result is incorrect</td>
</tr>
<tr>
<td></td>
<td>(i.e. the true result was too big or small to be represented)</td>
</tr>
</tbody>
</table>

[For simplicity you may assume a 4-bit datapath in the following answers.]

d) Sketch a schematic showing how the sign and zero flags can be derived from the ALU output.  

   (4 marks)

In the following ARM code:

```
MOV R0, #5 ; put number 5 in R0
CMP R0, #3 ; compare R0 with 3
BLO label ; branch if result was 'lower'
```

e) What is the state of each flag after the CMP instruction?  

   [CMP is effectively a subtraction]  

   (4 marks)

f) State the Boolean conditions needed to derive each of the following conditions from the flags:

   i) EQ (equal)  
   ii) HI (unsigned higher)  
   iii) LO (unsigned lower)  

   (3 marks)

g) Describe a simple way in which the conditional PC update may be implemented for the BLO instruction.  

   (The answer need not include a schematic)  

   (2 marks)
3. A branch predictor (as used in the first Pentiums) can be in one of the four states:

<table>
<thead>
<tr>
<th>State</th>
<th>Prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strongly not taken</td>
<td>Don't branch</td>
</tr>
<tr>
<td>Weakly not taken</td>
<td>Don't branch</td>
</tr>
<tr>
<td>Weakly taken</td>
<td>Branch</td>
</tr>
<tr>
<td>Strongly taken</td>
<td>Branch</td>
</tr>
</tbody>
</table>

When the branch is encountered the state is modified. If the branch was taken the state moves ‘down’ one place in this list; if it was not taken it moves ‘up’ one place. At other times the state does not change. State changes cannot go beyond either end of the list.

The state is used to predict whether the branch will be taken next time it is encountered.

a) If the branch has been taken five times and then not taken once, what next action is predicted? What state will the predictor be in? (3 marks)

b) What inputs and outputs does the predictor need? (3 marks)

c) Draw a state diagram representing the system. (4 marks)

d) What is the minimum number of flip-flops needed to build such a predictor? (2 marks)

e) Using a design flow of your choice derive and draw a schematic of a circuit which implements this prediction algorithm. Clearly label any inputs and outputs. (8 marks)