COMP 22111

Two hours

UNIVERSITY OF MANCHESTER
SCHOOL OF COMPUTER SCIENCE

Processor Microarchitecture

Date: Wednesday 17th January 2018

Time: 09:45 - 11:45

Please answer all FOUR Questions.

Use a SEPARATE answerbook for EACH Question.

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.

a) The choice of implementation hardware places constraints, such as area, on the design that may affect the final implementation. Discuss two constraints and how they affect a design produced using full custom design or a programmable logic device. (2 marks)

b) Discuss the difference between unit, integration and regression testing. A design for a 32-bit signed adder/subtractor has been produced from two 16-bit signed adders, would unit or integration testing be performed on the new design? (2 marks)

c) The Stump processor uses PC relative and indexed addressing modes. Briefly discuss these two addressing modes giving an example Stump instruction for each. (2 marks)

d) Figure 1 lists some Verilog code that forms the basis of a Verilog module design. Discuss, using a suitable diagram, how the simulator will handle the events created on the rising edge of `clk`. (2 marks)

```
module  dtype(input         D,
              input         clock, reset, CE,
              output reg    Q,
              output        Qbar);

always @ (posedge clock)

if(reset)
  Q <= 0;
else if (CE)
  Q <= D;
else
  Q <= Q;


assign Qbar = ~Q;

endmodule
```

Figure 1

e) The Stump register bank has unique feature that allows additional instructions such as MOV and CMP to be implemented. Discuss this and state corresponding Stump code examples for the implementation of MOV and CMP. (2 marks)

2.

a) Estimate roughly the number of flash transistors needed in a 64 GB memory card. Make sensible assumptions and explain your estimate. (2 marks)

b) Consider a flash memory card manufactured using a 20 nm process. Estimate how much more memory your card would store if the same chip area would be used in a 15 nm manufacturing process? Explain your estimate, assuming (for simplicity) that all parts of the chip will shrink proportionally. (2 marks)

c) Consider a memory card with 100 billion transistors where each transistor requires a size of 40x50 nm. How many dies of $1cm^2$ have to be stacked for building the memory card. For simplicity, we ignore other transistors needed. Provide details of your calculation. (3 marks)

d) Discuss briefly three important factors that impact monetary cost of an assumed memory card when moving to a more advanced process node using smaller transistors. (3 marks)

3.

a) The Stump processor offers three instruction formats: Type 1, Type 2 and Type 3. Briefly discuss the features of these three formats. (4 marks)

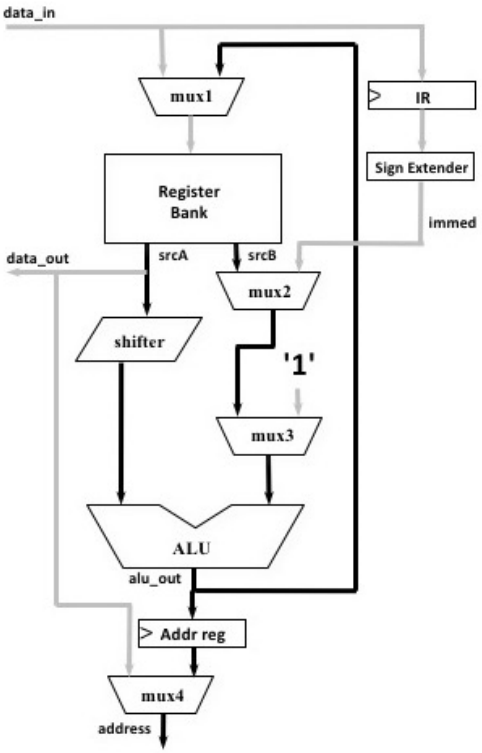
b) Figure 2 illustrates four completed signal usage charts (control signals have been removed for clarity) for the Stump processor where the path usage has been highlighted for the following cases:

a) Execute cycle of `AND R3, R0, #-9`
b) Fetch cycle of `SUB R6, R1, R4`
c) Memory cycle of `ST R4, [R3]`
d) Execute cycle of `BAL #36`

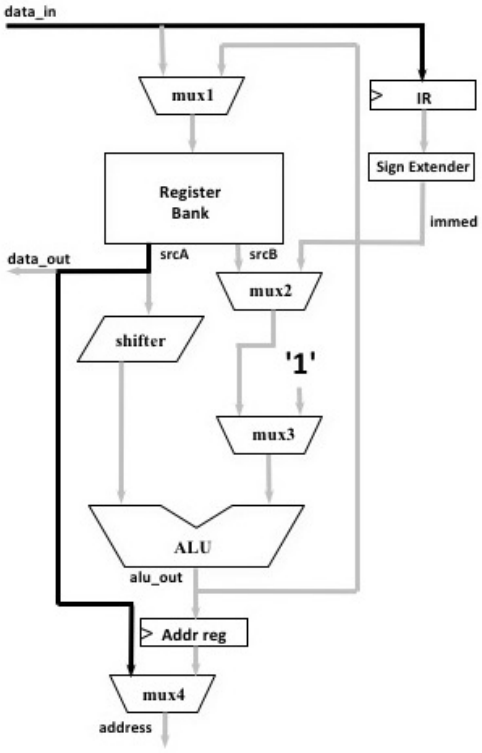
However, when completing the path usage diagrams some mistakes have been made. For each example identify explain where the errors have been made, outlining how it should be corrected. (8 marks)

(Question 3 continues on the following page)

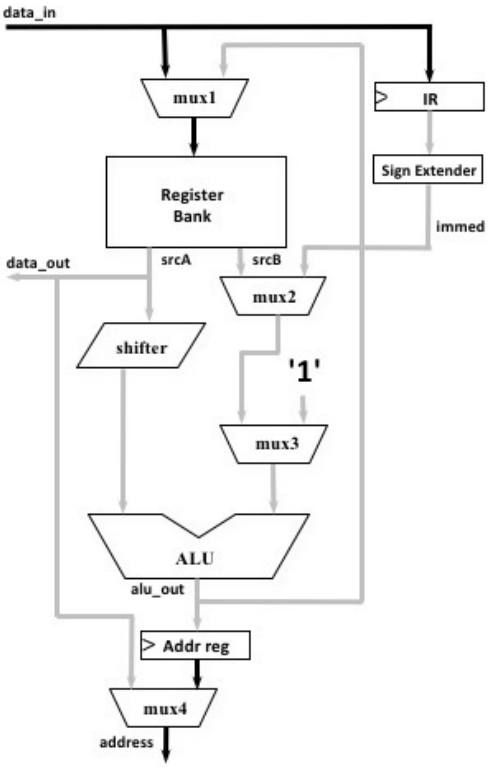
a) Execute - `AND R3, R0, #-9`



b) Fetch – `SUB R6, R1, R4`



c) Memory - `ST R4, [R3]`



d) Execute – `BAL #36`

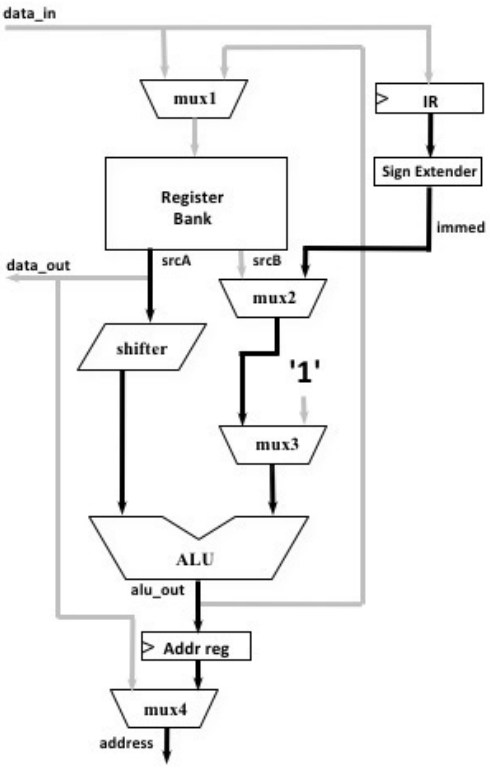


Figure 2

(Question 3 continues on the following page)

(Question 3 continues from the previous page)

c) Figure 3 provides the module definition for the Verilog design of the Stump Sign Extender. Produce the missing Verilog code to implement the required functionality for the sign extender to produce a 16-bit immediate value from the 5-bit or 8-bit data provided in the instruction. (4 marks)

```
module  stump_se(input   [7:0]   ir,
                 input           select,
                 output  [15:0]  immed);

// **********************
// Your design code here
// **********************

endmodule
```

Figure 3

d) Figure 4 provides a module definition for a testbench for the Stump Sign Extender design of Figure 3. Complete the Verlog code for the testbench, selecting appropriate test data to sufficiently test the correct operation of the Stump Sign Extender design. (4 marks)

```
module  test_stump_se();

// **********************
// Your test code here
// **********************

endmodule
```

Figure 3

[PTO]

4. A simple RISC microcontroller contains the following (indivisible) blocks which are used successively in the execution of an instruction. Their critical path timings are also given.

F: Fetch & Decode	5 ns
O: Operand read	3 ns
E: Execute	7 ns
R: Register write	2 ns

a) What is the fastest clock period possible (in ns) when adding 0, 1, 2 and 3 pipeline stages between blocks? Consider all four cases and assume that a pipeline register itself will add latency. Here we assume that the pipeline register will add 1 ns to the critical path delay to the block before the pipeline register and another 1 ns to the path delay related to the block after the register. (6 marks)

b) Comparing the previous results, discuss briefly the largest sensible number of pipeline stages for our simple microcontroller example. (2 marks)

c) Describe briefly the idea and operation behind register forwarding in a pipelined CPU. When is this technique beneficial? (2 marks)

d) Consider that the entire code that we execute on our microcontroller consists of additions that all contain data hazards (i.e. the result of the previous operation is needed for computing the present one). Compare the performance of the unpipelined version of the microcontroller with the fully pipelined version for the case

i) that we use register forwarding, and
ii) that we don't use register forwarding for the pipelined version.

Again, we assume that the pipeline register will add 1 ns to the critical path delay to the block before the pipeline register and another 1 ns to the path delay related to the block after the register. For simplicity, we omit any penalty that adding a forwarding path would incorporate. (2 marks)

e) Describe briefly two major micro-architectural/hardware differences between the fully pipelined and the multithreaded microcontroller. (2 marks)

f) Compare the peak performance of the fully pipelined with the multithreaded microcontroller. How should the workload for each microcontroller variant (fully pipelined and multithreaded) look like (which characteristics) in order to achieve peak performance. Explain your answers. (3 marks)

(Question 4 continues on the following page)

(Question 4 continues from the previous page)

g) As done for d), let us consider that the entire code that we execute on our microcontroller consists of additions that all contain data hazards. Compare the performance of the multithreaded version of the microcontroller with the fully pipelined version for the case

i) that we use register forwarding and
ii) that we don't use register forwarding for the fully pipelined version.

(3 marks)

END OF EXAMINATION