

COMPUTER SCIENCE TRIPOS Part IB

Tuesday 5 June 2018 1.30 to 4.30

COMPUTER SCIENCE Paper 5

Answer **five** questions.

Submit the answers in five **separate** bundles, each with its own cover sheet. On each cover sheet, write the numbers of **all** attempted questions, and circle the number of the question attached.

You may not start to read the questions printed on the subsequent pages of this question paper until instructed that you may do so by the Invigilator

STATIONERY REQUIREMENTS

Script paper

Blue cover sheets

Tags

SPECIAL REQUIREMENTS

Approved calculator permitted

1 Computer Design

- (a) Given an active high reset signal, how is an asynchronous reset described in SystemVerilog? [2 marks]
- (b) For each of the following six `always_ff` blocks, what sequence or error will be produced and why? You should assume `clk` is a clock and that all registers are reset to zero at the start (as they are for FPGAs). [3 marks each]

```
logic [2:0] rb, rc, rd, re, rf, rg;
always @(negedge clk)
  $display("rb=%d rc=%d rd=%d re=%d rf=%d rg=%d",
          rb, rc, rd, re, rf, rg);
```

```
always_ff @(posedge clk)
  rb <= (rb<6) ? rb+1 : 0;
```

```
always_ff @(posedge clk)
begin
  if(rc>=6) rc <= 0;
  rc <= rc+1;
end
```

```
always_ff @(posedge clk)
begin
  rd <= rd+1;
  if(rd>=6) rd <= 0;
end
```

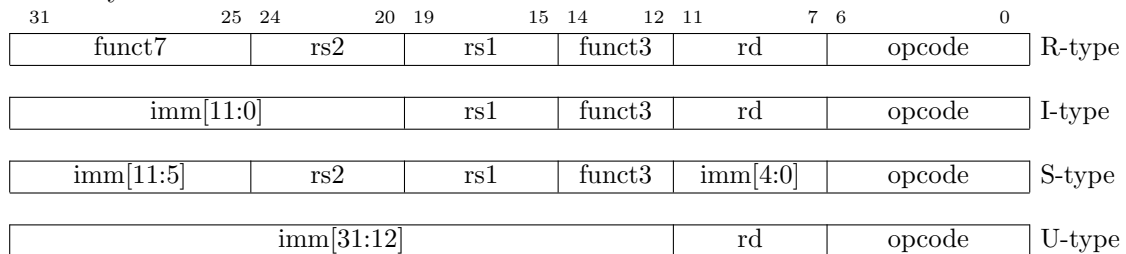
```
always_ff @(posedge clk)
begin
  if(re>=6) re = 0;
  re = re+1;
end
```

```
always_ff @(posedge clk)
  if(rf<6) rf <= rf-14;
  else    rf <= 0;
```

```
always_ff @(posedge clk)
  casex(rg)
    3'b0??: rg<=rg+1;
    3'b11?: rg<=0;
  endcase
```

2 Computer Design

The RISC-V base ISA instruction formats are reproduced below. Each immediate subfield is labelled with the bit position ($\text{imm}[x]$) in the immediate value being produced, rather than the bit position within the instruction's immediate field as is usually done.



- (a) In assembler, what is an *immediate*? [2 marks]
- (b) Give examples of *load*, *branch* and *arithmetic* instructions in assembler that use an immediate. Note that it is not critical for the RISC-V assembler syntax to be perfect, but for each instruction please explain your answer. [3 marks]
- (c) For a pipelined processor implementation, why is it an acceptable design choice for the bit position of immediates to vary between instruction formats? [2 marks]
- (d) For a pipelined processor implementation, why is it advantageous to have the source registers in the same bit position independent of the instruction format? [2 marks]
- (e) Why can there be many more register-to-register instructions with no immediates than instructions with immediates? [3 marks]
- (f) For a pipelined implementation of the RISC-V ISA, the sequential instruction execution model needs to be preserved. Cases where the pipelined implementation might deviate from this sequential model are often referred to as *hazards*. For a simple pipelined implementation, what hazards might exist and how might they be resolved? [8 marks]

3 Computer Design

- (a) A processor's main memory is usually implemented using DRAM.
- (i) Describe a typical DRAM cell. [2 marks]
 - (ii) Show, with the aid of a diagram, how DRAM is organised, making reference to devices, ranks, banks and arrays. [4 marks]
 - (iii) Describe the difference between an open-page and closed-page row-buffer policy and the types of access patterns they benefit. [2 marks]
- (b) The MOSI cache coherence protocol adds a new owned (O) state to the basic MSI protocol. When a cache holding a line in M state snoops a read request from another cache, it transitions to O state and forwards the data to the requestor. Subsequent snoops for read requests are also fulfilled by this owner cache. An owned line is dirty and only one cache can hold a line in O state at any time.
- (i) Describe the difference between cache coherence and memory consistency. [2 marks]
 - (ii) Draw a state transition diagram for the MOSI protocol, using a new action *Forward* to indicate data being forwarded from one cache to another. [6 marks]
 - (iii) Draw a table showing how the state of a line in one cache limits the states the same line can have in a different cache. [2 marks]
 - (iv) Give two benefits of adding this extra owned state to the basic MSI protocol. [2 marks]

4 Computer Networking

(a) Describe two drawbacks of layering. Provide an example for each. [4 marks]

(b) (i) Explain the single-bit parity error-detection code using a single byte of data. How many bit errors can this code detect?

(ii) Based on the single-bit parity error-detection code devise a new code to detect and correct a single 1-bit error in 4 bytes of data. How many parity bits do you require? You may assume that parity bits are error-free.

[8 marks]

(c) Consider a wireless network. For each of the following cases, state whether the packet transmission would be successful; assume no collision avoidance. Explain your answers.

(i) Nodes A and B are in range of each other; no other node is within range. Node A sends a packet to B.

(ii) Nodes A and B are in range of each other; nodes B and C are in range of each other; A and C are not in range of each other. Both A and C send a packet to B simultaneously.

(iii) Nodes A and B are in range of each other; nodes B and C are in range of each other; A and C are not in range of each other. C is transmitting and A wants to send a packet to B.

(iv) Nodes A and B are in range of each other; nodes B and C are in range of each other; A and C are not in range of each other. A is transmitting and B wants to send a packet to C.

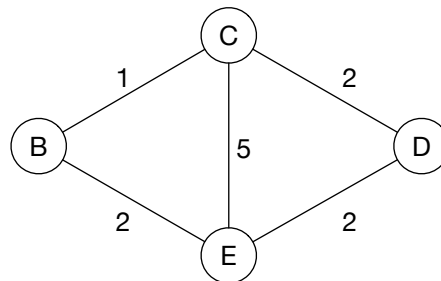
[8 marks]

5 Computer Networking

- (a) Consider two clusters A and B each hosting multiple applications. All applications send bursty traffic between A and B over a link E. Under what conditions is circuit switching more efficient to use as opposed to packet switching? [2 marks]
- (b) Compare the link state and distance-vector protocols in terms of message complexity, processing complexity and robustness. [6 marks]
- (c) Cambridge University is about to open a new School with three new departments A, B and C. The IPv4 address prefix of the new School is 128.232.1.0/24 and it is expecting each department to have the following number of hosts:
- Department A: between 40 and 60 hosts
 Department B: between 100 and 120 hosts
 Department C: between 20 and 30 hosts
- (i) The university wishes to allocate a subnet for each department. Give possible IPv4 subnet masks for each new department. [3 marks]
- (ii) Later, the School opens a fourth department D with 30 hosts. Provide possible IPv4 subnet masks to accommodate all four departments. [2 marks]
- (iii) Finally, the School opens a fifth department E of similar size to B. Provide possible IPv4 subnet masks to accommodate all five departments. [4 marks]
- (iv) Are there any practical problems with your answer to Part (c)(iii)? Briefly discuss an alternative solution to accommodate all five departments. [3 marks]

6 Computer Networking

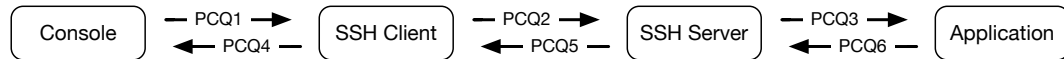
- (a) Briefly describe the main differences and similarities between routers and switches. [4 marks]
- (b) Consider the network shown in the figure below with four nodes. Cost links are shown in the diagram. Give the distance-vector routing tables for node C in the following two consecutive steps.
- (i) Step 0: C knows the distances to its immediate neighbours and [1 mark]
- (ii) Step 1: information from step 0 is exchanged as per the distance-vector algorithm. [3 marks]



- (c) (i) What is the problem that the Karn-Partridge algorithm aims to solve? [2 marks]
- (ii) Karn and Partridge proposed the use of exponential backoff in the TCP timeout value. Why is it that an exponential increase in the timeout value is more efficient than a linear increase for example? [2 marks]
- (d) What is the difference between congestion control and flow control in TCP? [2 marks]
- (e) Assume that three TCP flows f_1 , f_2 , f_3 , share a single link. The bandwidth of the link is 200 Mbps. The desired bandwidth of each flow is: f_1 60 Mbps, f_2 80 Mbps, and f_3 100 Mbps.
- (i) What would be the max-min bandwidth allocation of the flows? [2 marks]
- (ii) Propose a way to “cheat” max-min fairness so that f_3 increases its allocated bandwidth as computed in Part (e)(i). [4 marks]

7 Concurrent and Distributed Systems

UNIX pipes and SSH connections can be modeled as single-recipient *producer-consumer queues* (PCQs). Consider the following processes linked by PCQs forwarding console input to the application, and output back to the console:



Each queue has a fixed limit of N (≥ 1) bytes. A PCQ is *readable* if there is a non-zero number of bytes in its buffer. A PCQ is *writable* if there are fewer than N bytes of data in its buffer. Four (simplified) I/O system calls are used:

b = readbyte(pcq) Read one byte of data from **pcq**; blocks if **pcq** is empty.
writebyte(pcq, b) Writes one byte of data to **pcq**; blocks if **pcq** is full.
waitread(pcq1, ..) Block until at least one argument is readable.
pollread(pcq) Returns true if **pcq** is readable.

With crypto omitted, SSH client and server workloops are implemented as:

```

1: while (1) {
2:   waitread(input1, input2); // Wait for input on either PCQ
3:   if (pollread(input1)) {
4:     b = readbyte(input1);
5:     writebyte(output2, b);
6:   }
7:   if (pollread(input2)) {
8:     b = readbyte(input2);
9:     writebyte(output1, b);
10:  }

```

(a) What is the maximum amount of data buffered across all PCQs? [2 marks]

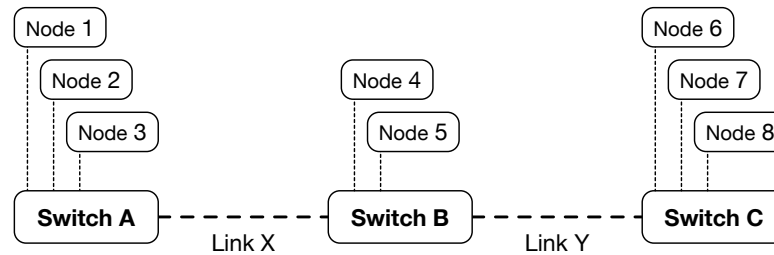
(b) Applications often *echo* user keypresses, printing input characters as they process them. If the user hits the 'A' key, at most how many times will PCQ semaphores be signaled before the character is printed on the console? [2 marks]

(c) The system operator sets N to 1 and pastes a long list of commands into the console. Part way through, the SSH Server and Application processes hang. Succinctly explain what happened, describing PCQ3 and PCQ6 starting states, initial line number for the SSH Server, and event sequence. [6 marks]

(d) Explain why, on a busy system, key press echoes might be delayed when a high-priority user interacts with a low-priority application. Propose a solution, describing how each of the above system calls should be changed, and any additional state required. Describe limitations that might apply. [10 marks]

8 Concurrent and Distributed Systems

The *Bully Algorithm* allows a set of nodes to elect a unique *leader* in the presence of node failures. For the purposes of this question, assume that: the set of nodes is known in advance by all nodes; each process knows its own identifier; processes fail and restart cleanly; node failure can be reliably detected; and message delivery is reliable unless the destination node has failed. Our network contains eight nodes (1–8), which are connected via three switches (A–C):



- (a) In the absence of node or switch failures, which node will be elected leader using the Bully Algorithm? [1 mark]
- (b) A new Node 9 is added to the network, attached to Switch C. Which node will be elected leader using the Bully Algorithm? [1 mark]
- (c) If there is a network partition between Switches B and C, what node will be elected leader using the Bully Algorithm? [2 marks]
- (d) To address the concern about network partition, the authors of a distributed system introduce a majority rule: for a node to declare itself leader, it must receive OK messages from a majority of nodes (including itself). Which combinations of links can fail while still permitting a successful election, and, for each of those combinations, which node is elected leader? [4 marks]
- (e) One key metric used in evaluating distributed algorithms is the number of messages generated. What is the worst-case cost (using O-notation) in messages with n nodes participating in the Bully Algorithm? Explain why this is the case. [2 marks]
- (f) The Bully Algorithm provides a strong invariant that a node can identify that it is the unique leader in a distributed system. However, that leadership status can be preempted (revoked) without node failure. Explain how this may occur. [4 marks]
- (g) Describe how a distributed application can utilise elected leadership status to safely allocate unique transaction IDs. Explain why this works, considering in particular: (1) the steady state; (2) leader change due to leader crash; and (3) leader change due to leadership revocation. [6 marks]

END OF PAPER