University of Bristol, School of Computer Science

COMSM1302 Overview of Computer Architecture

Class Test I: Practical (practice)

27 October 2023 Marks available: 50

You are only permitted to use the exam version of Logisim (linked in the test instructions on Blackboard). You are **not** permitted to visit any web sites other than the Blackboard tests. You are permitted to use the Calculator app, but in Standard and Scientific modes **only**. You are **not** permitted the use of physical calculators.

For each question, build the circuit within the Logisim skeleton file provided. For example, the circuit for Question 1 should be built with the Q1 subcircuit. Each subcircuit will already contain every input and output pin required. The components permitted to build the circuit are listed at the end of each question — do not use any components other than those explicitly permitted and those in the "Wiring" folder (e.g. constants and splitters).

Full marks will be given to circuits that display the correct behaviour while only using permitted components. You are not required to use all permitted components. Partial marks will be available. Complexity and neatness will not be marked, unless otherwise stated.

You should submit your completed Logisim file to the "In-Class Test 1 (Practical component) submission point on Blackboard. Multiple submissions are allowed, but only the last one will be marked.

1. **(10 marks)** Create a circuit that implements the Boolean expression:

$$(A \wedge B \wedge C) \vee \neg(\neg B \vee C) \vee \neg A.$$

Permitted components: 2-input AND gates, 2-input OR gates, and NOT gates.

2. **(10 marks)** Create a circuit that implements the below truth table, using at most 7 logic gates. Partial credit will be given for any solution that implements the truth table with more than 7 gates, depending on the number of gates used.

A	B	C	D	Out
0	0	0	0	1
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	0

Permitted components: Any logic gates with 2 or fewer inputs.

3. **(15 marks)** Create a circuit that implements a rising-edge D flip-flop with inputs D and CLK and outputs Q and Q'.

Permitted components: 2-input NAND gates.

4. **(15 marks)** The Logisim skeleton subcircuit for this question contains a pre-built RAM component with data RAM[0], . . . , RAM[63], a button labelled input, and a 8-bit output. Create a circuit which behaves as follows.

While the input button is pressed: The output should become RAM[1] at the first rising clock edge after the input button is pressed. At the next rising clock edge, the output should become RAM[2]. It should continue cycling through RAM in this way until reaching RAM[31]. On the next rising clock edge, the output should become RAM[0]. On the next rising clock edge, the output should become RAM[1] again and the cycle should repeat.

While the input button is not pressed: The output should become RAM[0] at the first rising clock edge and stay at this value.

Permitted components: All components are permitted.