2019 COMP3222 Lab 9 Notes and Additional Questions

Lab 9 has two parts. The teaching staff have decided that the second part, Part II, should be optional and that if you do successfully demonstrate that part, it will provide you with a bonus mark for the course. Completing Part I successfully will score up to 3 marks contributing to the lab component of your assessment – one mark for your paper design, comprising a Moore state diagram and multiplexor truth table; one mark for a simulation that replicates the waveform provided in Figure 3; and one mark for code that satisfies the previously mentioned VHDL style expectations to a high standard.

Part I

In this part, you are provided with the block diagram of a processor (Figure 1) and a soft copy of skeleton code (Figure 2a-2c) on the course website that implements most of its components. You are expected to complete the design and implementation of the processor by:

1. Using the skeleton code to identify the control unit FSM and producing a Moore state diagram prior to coding. This must be shown to your lab demonstrator.
2. Producing a truth table for the Multiplexers unit prior to coding. This must be shown to your lab demonstrator.
3. Implementing the adder/subtractor unit.
4. Instantiating all necessary registers and interconnecting them.
5. Producing a simulation waveform as depicted in Figure 3 to validate your design prior to mapping the processor to the DE1 Starter Board. This must be shown to your lab demonstrator.
6. Creating a new (second) project and mapping the processor from steps 1-5 above to the DE1 board as specified in steps 4-6 of the Laboratory Exercise sheet for Part I.

Note 1: We will not mark this part without your paper-based designs for the FSM's state diagram and Multiplexers' truth table.

Note 2: You must adhere to the block diagram given in Figure 1 and implement the circuit as illustrated.

Note 3: For your functional simulation only use the signals and waveforms provided in Figure 3. The necessary signals can be found in the node finder window by listing "Design Entry (all names)" and selecting the Combinational Groups corresponding to the Q outputs of the desired registers. Do not be concerned if the Tsetp_Q output does not appear as illustrated.

OPTIONAL Part II

Note 1: Implement the circuit exactly as shown in Figure 4.

Note 2: Follow the MIF file format as described on the Intel Quartus Memory Initialization File (.mif) Definition page.
(https://www.intel.com/content/www/us/en/programmable/quartushelp/17.0/reference/glossary/def_mif.htm)

Note 3: Demonstrate your solution by encoding precisely those instructions you used to simulate the processor in Part I, step 5 above, in your MIF file and executing these.