UNIVERSITY of York

Department of Electronic Engineering

Assessments 2019/20

ELE00067M

Digital Design

This assessment (**Final Project**) contributes **30%** of the assessment for this module.

Clearly indicate your **Exam Number** on every separate piece of work submitted.

Unless the assessment specifies a group submission, you should assume all submissions are individual and therefore should be your own work.

All assessment submissions are subject to the Department's policy on plagiarism and, wherever possible, will be checked by the Department using Turnitin software.

Submission is via VLE and is due by **12:00** on **20 January 2020 (Spring Term, Week 3, Monday)**. Please try and submit early as any late submissions will be penalised.

Please remember that if this is your first year of study, you need to complete the mandatory Academic Integrity Tutorial http://www.york.ac.uk/integrity/

ELE00067M Digital Design

Final project: Part III

Design of a (parameterizable) single-cycle datapath

THE PROJECT SHOULD BE DONE IN GROUPS OF TWO STUDENTS

Report formatting:

There is no formal report structure – you will be marked on the items listed within the script. The report will include code printouts and simulation screenshots – see Lab 1 appendices for guidelines.

Only one report for each group for the entire project has to be submitted, in a single pdf file, via the VLE by the deadline indicated on the front page of the script. The PDF file should be named **Yxxxxxxx-Yxxxxxxx_DDMSc_Project.pdf**, where the Yxxxxxxx are the exam numbers of the group members. The PDF file should be compressed and submitted as a .zip archive.

The report should include all the material listed in each of the scripts that make up the complete project. The material for each script should start on a new page, containing all the items required in the order specified. The exam number should be printed on the first page of each script report.

In all cases, read carefully the instructions on the VLE submission page. Failure to follow the instructions could lead to your assignment not being marked and in any case to a mark penalty.

Submission weight on module mark: 30%

Marking criteria:

In general, the marking will take into account:

- Design quality, i.e. does the circuit generated by your VHDL operate correctly and efficiently.
- VHDL quality, i.e., is it written in accordance with the guidelines presented in the lectures.
- Documentation, i.e. are all required documents present? Are they legible? Is the code commented? (Refer to VHDL Style Guide for acceptable standards)

A parameterizable datapath **[10 marks]**

The objective of this task is to design the datapath for a single-cycle processor implementation: The datapath should use the ALU and the register bank implemented in the previous two labs.

The connections between components (including muxes) should correspond to the architecture described in the lectures with the label "Architecture B" (attached at the end of this script as a reminder). Ignore the control logic for the moment - all control signals, immediate values, addresses, and flags should be I/O ports of your entity and assigned/read in the testbench.

The size of the data in the datapath should be parameterizable and defined by a generic value. Again, you can use the *size* and *log2* functions.

The operation of the datapath should be partially verified (the full verification will only occur at the end of the project) with the following sequence of instructions (for which you will have to define the necessary control signals and immediate values, provide them at the appropriate time, and verify that all outputs are correct and that timing is respected). For the testbench, assume a data and address width of 16 and 32 registers in the register bank. All *don't care* values should be assigned value '0' in the testbench.

```
inc R1, R0;          [store the value 1 into register R1]
addi R2, R0, 0005;   [store value 5 into register R2]
shl R3, R1, 3;       [store value 8 into register R3]
storr R2, R3;        [store 5 into memory at address 8]
loadi R5, 1f1f;      [load into R5 the contents of the memory at address 1F1F – assume
                      that the value in question is 7 and provide it in your testbench]
```

Note: the DATA memory should not be implemented, but emulated in the testbench (providing the data value for the load operations at the correct time). Also, the OEN signal should be generated by the testbench, but should not be used by the datapath (eventually, it will be sent directly to the data memory together with the data to be written).

REPORT: The report for this exercise should include:

- The commented VHDL code for the datapath and its components
- The commented VHDL testbench. Use a record to define the control signals.
- A printout of the simulator results for the testbench – please comment the screenshots to highlight the operations being performed and make sure that the result of each operation (content of the written register or output to memory) is clearly readable.
- The "RTL Component Statistics" part of the synthesis report

ARCHITECTURE B: