UNIVERSITY OF LIVERPOOL

Department of Electrical Engineering and Electronics

ELEC373

Digital Systems Design
Assignment 4
NIOS II

Module	ELEC373
Coursework name	Assignment 4
Component weight	20%
Semester	2
HE Level	6
Lab location	PC labs 301, 304 as timetabled, at other times for private study
Work	Individually
Timetabled time	15 hours (3 hours per week – Tuesday 9am - noon)
Suggested private study	10 hours including report writing
Assessment method	Individual, formal word-processed reports (Block diagrams and ASMs can be hand drawn and scanned into the report)
Submission format	Online via CANVAS
Submission deadline	23:59 on Friday 19th May 2023
Late submission	Standard university penalty applies
Resit opportunity	August resit period (if total module failed)
Marking policy	Marked and moderated independently
Anonymous marking	Yes
Feedback	Via comments on Canvas
Learning outcomes	LO1: Ability to design digital systems using the ASM design method LO2: Ability to implement digital systems using the Verilog Hardware Description Language LO4: Ability to implement a SOPC system using Quartus Nios II.

Marking Criteria

Section	Marks available	Indicative characteristics	
		Adequate / pass (40%)	**Very good / Excellent**
Presentation and structure	20%	• Contains cover page information, table of contents, sections with appropriate headings. • Comprehensible language; punctuation, grammar and spelling accurate. • Equations legible, numbered and presented correctly. • Appropriately formatted reference list.	• Appropriate use of technical, mathematic and academic terminology and conventions. • Word processed with consistent formatting. • Pages numbered, figures and tables captioned. • All sections clearly signposted. • Correct cross-referencing (of figures, tables, equations) and citations.
Introduction, Method and Design	40%	• Problem background introduced clearly. • Evidence of a Top Down Design approach • Conceptual Design Choices introduced. • Design of each module follows a logical sequence. • ASMs correspond to designs for each block. • Software is clearly commented	• Appropriate range of references used. • Design decisions justified with alternatives given. • Calculations shown in full, justifying and explaining any decisions. • Correct ASM Syntax used. • Well-structured Verilog Code
Results	30%	• Simulation results present for each block and well annotated. • Results of full system in both simulation and experimentally presented. • Results for each task accompanied by a commentary. • Screen shots of results presented.	• Tests indicate that there are no problems caused by asynchronous inputs. • Clear explanation of how the instructions operate correctly
Discussion	10%	• Discussion on what worked and what didn't. • Critical assessment on the design – strength and weaknesses	• Discussion on how the system was fully tested.

ELEC 373 Assignment 4 (2022-23) Synthesising the NIOS II Processor

Assignment Outline

In Assignment 3 you added some extra instructions to a MIPS processor which was then synthesised and executed on the DE2 board. This assignment aims to introduce you to a commercial synthesised processor targeted for Altera FPGAs, which allows the easy importing of peripherals and the use of an industry standard IDE for software development.

Assignment 4 is split into two parts, Parts A and B. The objective of Part A is to get you familiar with QSYS, the NIOS II Processor and interfacing SRAM and SDRAM, on the Altera DE2 Board, to the NIOS II processor you synthesise. Part B requires you to develop and test a Custom Instruction that will be implemented in the FPGA.

Part A – SRAM & SDRAM

The NIOS II hardware development tutorial synthesises a NIOS II processor with, 20 KB of on-chip memory, a timer, a JTAG UART, 8 parallel I/O pins and a system ID Component. Part A of this assignment requires you to interface the 512 KB SRAM and 8 MB SDRAM on the Altera DE2 Board to this design.

You should then test that the memory is functioning correctly by running the Memory Test programs available within the NIOS II IDE. *You should modify the Memory Test Programme so that your Name and ID number are shown in the terminal window each time the memory is tested.*

Hints

1. Initially use the altera_up_avalon_sram Controller (SRAM/SSRAM) for the SRAM interface
2. Use the SDRAM controller for the SDRAM Interface
3. Add a PLL to advance the clock for the SDRAM by 3ns compared with the system clock.

Part B – Custom Instruction

Part B requires you to develop a Custom Instruction to count the number of leading 1s (or 0s – see Table 1) in the 32 bit number passed to the instruction (for example 0xFF000000 would have 8 leading 1s and 0 leading 0s, whilst 0x00F00000 would have 0 leading 1s and 8 leading 0s). You should write a program to test your Custom Instruction. You should also develop a test routine in C or assembler that performs the same function as the Custom Instruction and compare the speed of the Custom Instruction against your software implementation.

Submission

You report should include the following:

1. Block diagram of BDF developed in Part A.
2. Table showing memory Map for Part A
3. Screen dumps of test results showing memory test programs working for SRAM and SDRAM
4. ASM(s) and Verilog code for your custom instruction in Part B
5. C/C++ for your test program of Part B
6. Screen dump showing the results of your program for Part B
7. Results showing a speed comparison between the Custom Instruction and your software implementation.
8. Explanation of your results.

Submission Deadline

Vital: Friday 19th May 2023 @ 11:59pm

Table 1 Part B Custom Instruction Task

ID	Name	Counting
201460324	Alhuwayji, Abdullah Ahmed A	Leading 0s
201600204	Cao, Zebin	Leading 0s
201600220	Chen, Jialei	Leading 1s
201413268	Day, Erik Johan George	Leading 1s
201510374	Denston, Gregory	Leading 0s
201600317	Du, Yuang	Leading 0s
201600335	Feng, Haoru	Leading 1s
201411100	Fitzsimmons, Joseph	Leading 1s
200956435	Glover, Aaron Phillip	Leading 0s
201600366	Gong, Shengwei	Leading 0s
201534766	Haggar, Finlay Michael	Leading 1s
201600417	Hou, Menghui	Leading 1s
201600499	Jiao, Tiankuo	Leading 0s
201600504	Jin, Hanyu	Leading 0s
201600593	Li, Zekun	Leading 1s
201600594	Li, Zhengxing	Leading 1s
201600620	Liu, Binkai	Leading 0s
201600670	Liu, Yuanchen	Leading 0s
201600717	Ma, Hanyuan	Leading 1s
201460239	Mahmoud, Omran Hussain Ali Hassan	Leading 1s
201600730	Mao, Zhenyifan	Leading 0s
201600743	Miao, Yue	Leading 0s
201600773	Qian, Shangyuan	Leading 1s
201600820	Shen, Zhouyi	Leading 1s
201600887	Tao, Mingrui	Leading 0s
201427481	Thompson, Jake	Leading 0s
201600921	Wang, Jiayu	Leading 1s
201457862	Wang, Ruoyu	Leading 1s
201600941	Wang, Shibo	Leading 0s
201600944	Wang, Shiyan	Leading 0s
201600951	Wang, Xinmi	Leading 1s
201600996	Wang, Zhongpei	Leading 1s
201601008	Wen, Qinzheng	Leading 0s
201360350	Woodward, Elliot	Leading 0s
201521301	Wu, Fan	Leading 1s
201580975	Wu, Xuan	Leading 1s
201601082	Xu, Minghong	Leading 0s
201601083	Xu, Nan	Leading 0s
201601154	Yao, Qihang	Leading 1s
201601168	Ye, Zhian	Leading 1s
201601191	Yuan, Chenjie	Leading 0s
201601215	Zhang, Chao	Leading 0s
201522657	Zhang, Quanze	Leading 1s
201601308	Zhao, Yuetian	Leading 1s
201601314	Zheng, Chenyu	Leading 0s
201601336	Zhou, Jinkai	Leading 0s
201601355	Zhou, Yiyang	Leading 1s
201601371	Zhu, Qilong	Leading 1s
201601384	Zhu, Yuhao	Leading 0s

J.S. Smith 24/4/23