Practical 4: Basys3 Hex Calculator

Aim:

The aim of this practical is to design and implement a calculator on the Basys3 board. This calculator will operate and display results in hexadecimal, making heavy use of the available Basys3 board IO. While the exact specification of the calculator is up to you, all calculators should be able to add and subtract, along with some additional features of your choosing.

Learning Outcomes:

On completing this practical you will be able to:

- Design a state machine and use to generate a hex calculator with your own feature set and controls,
- Apply a top down design methodology,
- Review and understand all steps in the FPGA design flow

Design Requirements:

The specification of this assignment is intentionally open; it is up to you to decide what features you wish to implement and how you will implement them. However you should consider the following points when deciding the specification of your block:

- In the absence of a keypad for numeric input, it is advised that you use the 16 switches to capture a 16 bit binary word as input.
- Due to limitations of the seven-seg display, inputs and results should be limited to 4 hex digits.
- Consider how your calculator will handle special cases, e.g. negative results, results with more than 4 digits.
- Decide what operations you would like the calculator to be able to perform, and any additional features such as memory or recall. Using one button as a 'shift' button to allow other buttons to have multiple purposes may be useful if you wish to have more than 5 operations or features.

Instructions

1. You should apply a top-down approach to this design; begin by writing a short specification detailing your design intent, ensuring it fits the design requirements.
2. Draw a top-level diagram showing the separate blocks in your design.
3. Take an incremental approach to the design, aiming to get something working the board ASAP and then expand the functionality.
4. For each generation of the bit-stream, review the synthesis and implementation reports. Ensure that you address any Warnings you can, and comment to explain any remaining Warnings. Save the synthesis and implementation report used to generate your final bit-stream. **Ensure there are no latches in your design.**
5. Your write-up should contain the specification, diagrams and document all design decisions. You **must** comment to explain any implementation or synthesis warnings in your final design.

Submission:

You must submit the following in a single zipped folder with tile pract3p1_surnameinitial.zip, e.g. called pract3p1_WadeR.zip on Blackboard:

- Write up with title in form pract3p1_surnameinitial.pdf, e.g. called pract3p1_WadeR.pdf
- Vivado project folder
    - **Please ensure that your project folder includes all files necessary to reproduce your design on another Basys3 board. When reusing any Verilog files (e.g. xdc, seven_seg), be sure to copy them into your design when importing them.**

Deadline: **Thursday Week 12, 10pm**. Refer to first lecture for Late Assignment Policy.

Plagiarism

'Plagiarism is interpreted by the University as the act of presenting the work of others as one's own work, without acknowledgement. Plagiarism is considered as academically fraudulent, and an offence against University discipline. The University considers plagiarism to be a major offence, and subject to the disciplinary procedures of the University.'

Above is an extract from the College Regulations on Plagiarism. Plagiarism will result in loss of **ALL** marks for this assignment, for the owners of both the original and copied work.