Homework 4

Improved Single-cycle Processor

The single-cycle processor provided to you in Lab 4 can execute the following instructions: ADD, SUB, AND, ORR, LDR, STR, and B. In this homework 4, you will enhance the processor's capabilities by implementing two additional instructions.

Your updated single-cycle processor must be able to handle the **EOR** and **LDRB** instructions in addition to its basic instruction set. For guidance on implementing these instructions, refer to Appendix B of the reference book.

To verify that your processor works correctly with the new instructions, please use the assembly code provided below. Please use Assdeb to convert this assembler code into hexadecimal instructions.

```
main: SUB     R0, R15, R15
      ADD     R1, R0, #255
      ADD     R2, R1, R1
      STR     R2, [R0, #196]
      EOR     R3, R1, #77
      AND     R4, R3, #0x1F
      ADD     R5, R3, R4
      LDRB    R6, [R5]
      LDRB    R7, [R5, #1]
      SUBS    R0, R6, R7
      BLT     main
      BGT     here
      STR     R1, [R4, #110]
      B       main
here: STR     R6, [R4,#110]
```

Figure 1: Assembly code for testing your single-cycle processor

Before submitting to Inginious :

1. Test your implementation as you learned it in the lab, using a testbench.
2. Check that the name of your main module in your .sv file matches the module description on Inginious. Submit your entire design **including the module definition** on Inginious to be graded.
3. You get **three attempts** before you **start losing 10% of the grade** for each new **failed attempts**.