EE 310 Hardware Description Languages
Spring 2018

Laboratory Assignment #3

Due Date: 15 May 2018

In this lab, you will write Verilog RTL model of a digital hardware that performs the following addition and multiplication operations on signed fixed-point or floating-point numbers. You will then load this design into Xilinx Spartan FPGA on the DIGILENT Spartan-3 Starter Board.

$$result = (a + b) * c$$

Your hardware should take format type (fixed-point or floating-point) and three numbers as inputs, and calculate the output. Your hardware should work at 50 MHz.

A fixed-point number is represented using a fixed number of digits before and after radix point. If format type is specified as fixed-point, your hardware should take three input numbers as 10 bit signed fixed-point numbers with 5 bits before the radix point (integer part) and 5 bits after the radix point (fractional part). For example, 1.25 and -0.625 are represented as 0000101000 and 1111101100, respectively, in 10 bit signed fixed-point format.

10-bit fixed point format: $[5\ bits\ integer][5\ bits\ fraction]$

A floating-point number is represented using a *mantissa* and *exponent*. If format type is specified as floating-point, your hardware should take three input numbers as 10 bit signed floating-point numbers with 1 bit sign, 5 bits mantissa and 4 bits exponent. The base of exponent is 2. Therefore, a number given in 10-bit floating point format is equal to $(-1)^{sign} x\ mantissa\ x\ (2)^{exponent}$. For example, 1.25 and -0.625 are represented as 0010000000 and 1010001111, respectively, in 10 bit signed floating-point format.

10-bit floating-point format: $[1\ bit\ sign][5\ bits\ mantissa][4\ bits\ exponent]$

There is also an implied bit of mantissa which is always 1. For example, mantissa 10101 is equal to 1.10101. Therefore, mantissa can take a value between ±1 and ±2.

Output of your hardware should be in the same format as its inputs (10 bit signed fixed-point or floating-point). You can assume that the input numbers will be given such that there won't be overflow after addition and multiplication operations.

After reset, you should give format type input to your hardware according to the table given below by using a switch (SW0) on the board. You should set this switch and press the button (BTN).

SW0	Format Type
0	Fixed-Point
1	Floating-Point

You should then give three 10 bit signed fixed-point or floating-point numbers as inputs to your hardware by using 4 switches (SW0-SW3) on the board. For each input, you should set these 4 switches and press the button (BTN). After getting the inputs, your hardware should calculate the output value and display it on the LCD screen in hexadecimal format. You will be given an LCD controller for displaying hexadecimal values on the LCD screen.

Your hardware design should have one Verilog module with the following interface:

Port Name	Width	Direction	FPGA-pins
Clock	1 bit	Input	C9
Reset	1 bit	Input	K17
SW	4 bits	Input	N17, H18, L14, L13
BTN	1 bits	Input	V4
Data_Out	4 bits	Output	M15, P17, R16, R15
LCD_Control	3 bits	Output	M18, L18, L17

Before synthesizing your design, you should write a Verilog testbench and verify the correctness of your Verilog RTL by simulation. After that, synthesize your RTL code using Xilinx XST synthesis tool targeting Xilinx Spartan XC3S500E FG320 FPGA with speed grade 4. Next, place and route the resulting netlist into Xilinx Spartan FPGA and generate the FPGA configuration bitstream using Xilinx ISE. Then, download the bitstream into Xilinx Spartan FPGA and verify your hardware design on the board.

Finally, put all of your Verilog modules, Testbench, UCF file and FPGA bitstream into one ".zip" file named Lab3_Partner1Lastname_Partner2Lastname.zip (e.g. *Lab3_Azgin_Hamzaoglu.zip*) and submit this zip file using EE310 SUCourse website.