



UNIVERSITY OF TEHRAN
Electrical and Computer Engineering Department
Digital Logic Design, ECE 367 / Digital Systems I, ECE 894
Spring 1399-1400
Computer Assignment 6, Week 16
Hierarchical RTL Design

Name:

Date:

In this experiment, you are to design and implement a 32-bit floating-point multiplier that uses the IEEE 754-2019 Floating-Point standard. In a hierarchical fashion, the circuit uses a 24-bit sequential multiplier as done in class.

A 32-bit signed multiplier (32-bit FP A and B inputs and producing a 32-bit result) is to be designed. The inputs are taken from the 32-bit input bus *inBus* in two 2-line responsive handshaking processes, and the output is to be placed on the 32-bit *outBus*. For the inputs a 2-line fully-responsive handshaking is used for making the data available on *inBus*. The external *inReady* becomes 1 when inputs are ready and stays 1 until our circuit issues *inAccept*. This is followed by another round of handshaking for the second input. When result is ready, our circuit issues *resultReady* and waits for the device reading the output to issue *resultAccepted*.

The main floating-point (FP) multiplier has a *startFP* and a *doneFP* output. When its two inputs are ready, a positive pulse of *startFP* starts the FP process. When this is done, the *doneFP* is asserted and stays asserted for as long as a new set of data has not started.

The circuit has a wrapper that handles the external handshaking as discussed above and delivers it to the FP multiplier. The same wrapper also handles the output handshaking.

The circuit uses a sequential multiplier and a combinational adder for its internal processing. The sequential multiplier starts with a positive pulse on *startMul*. When received, the circuit accepts both its inputs and performs an unsigned multiplication. The multiplication result becomes available on the sequential shift-and-add multiplier's 24-bit output and a *doneMul* is issued.

- a. Show the hierarchical block diagram of your system showing how the FP multiplication part uses the sequential multiplier, and how the wrapper circuits are wrapped around the FP multiplier.
- b. Show the complete design of the sequential multiplier.
- c. Show the datapath of the FP multiplier engine (excluding the wrapper). Show its control inputs, control outputs, and input and output busses.
- d. Show the wrappers that you are designing for this circuit. Show datapath and the controller of the wrapper circuits.

You have several choices for completion of this assignment.

- a. Complete design using a Sequential multiplier and hand-shaking wrappers.
- b. Using the multiply operation in place of the sequential multiplier.
- c. Designing the FP multiplier without the wrappers.
- d. And variation and mix of the above.

Deliverables:

Generate a report that includes item discussed below for each of the four parts of this computer assignment.

- a. Show the block diagram of the entire circuit including the three parts, Wrappers, Floating Point, and the Sequential Multiplier.
- b. Show datapath and controller of the FP multiplication part.
- c. Show datapath and controller of the Wrapper circuits.
- d. Show datapath and controller of the Sequential multiplication part.
- e. Write complete SystemVerilog description of the FP multiplier part. Write a testbench and test this part.
- f. Write complete SystemVerilog description of the Wrapper circuits. Write a testbench and test this part.
- g. Write complete SystemVerilog description of the Sequential multiplier part. Write a testbench and test this part.
- h. Put the above three parts together and create the complete design. Simulate this circuit.
- i. Synthesize the Sequential multiplier part and create a symbol for it.
- j. Synthesize the Wrapper circuit(s) and create a symbol for it.
- k. Synthesize the FP multiplier part (excluding the Wrappers and the Sequential multiplier part) and create a symbol for it. Show synthesis reports.
- l. Create post-synthesis output and simulate pre-and post-synthesis outputs in the same testbench.
- m. Show your SystemVerilog description of the design you are simulating and the testbench for it. Best is to use the Snip tools to get the image from Notepad++. This way you see all keywords and indentations. Make sure your SV codes are properly indented and all line-up rules are followed.
- n. Show images of the project that you have created in ModelSim for the simulation of your circuit.
- o. When simulation is complete, or even if you have a partial simulation, include an image of the output waveform showing signal names being displayed.
- p. Show projects and synthesis reports.

Make a PDF file of your report and name it with the format shown below:

FirstinitialLastnameStudentnumber-CAn-ECEmmm

Where *nn* is a two-digit number for the Computer Assignment, *mmm* is the three-digit course number under which you are registered, and hopefully you know the rest. For the *Firstinitial* use only one character. For *Lastname* and for the multi-part last names use the part you are most identified with. Use the last five digits of your student id (exclude 8101) for the *Studentnumber* field of the report file name.