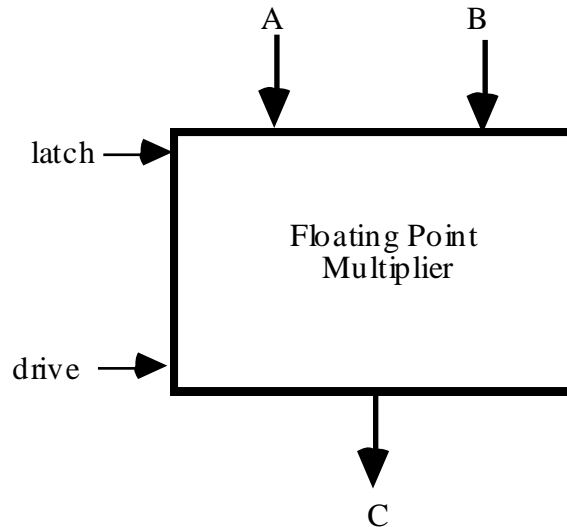


Project Assignment #10 Floating Pt Unit**DUE: - Friday Mar 10**

In this assignment you will use VHDL to describe the function of a floating point multiplier. The multiplier will accept IEEE Standard 754 single precision inputs and produce single precision output. It will support NaN, $\pm\infty$, ± 0 , normalized numbers, and denormalized numbers.

The interface to the design will be:



Inputs will arrive as per the attached specification and test bench. Your design will latch inputs using the latch input signal. Your design will drive the outputs using the drive signal. After driving the output, when the drive signal again goes high, you must drive the bus back to high impedance. Use the component interface given in the testbench.

STEP 1) Write the initial architecture. In the initial architecture, simply latch the input, route the A input to the output, and correctly drive the C bus. If you get this step working correctly the rest will go much easier. **SIMULATE**

STEP 2) Write a VHDL process to do the floating point multiply.

Recommendation : As you start to write your routine handle special cases first. Write the code to handle the NaNs - then simulate and check that you handle the NaNs correctly. Note that $\pm\infty * \pm 0$ results in a NaN.. Then correctly handle $\pm\infty$, and verify through simulation. Then correctly handle ± 0 and simulate. Finally do the cases where you actually have to multiply.

You can use process(es), concurrent signal assignment, etc., as you would like.

You will find the following files in ~degroat/ee762_assign and though the web page.

fpmtb.vhdl - the test bench - the component declaration, configuration, and instantiation have been done but you can change them if you want to.

fpmvectors - a list of the input stimulus
fpm.do - do file for listing and waveform

Other NOTES for floating point multiplier.

The test bench uses a concurrent procedure call that reads the testvector file in ~degroat/ee762_assign/fpmvectors. Use of these vectors is hard coded in the concurrent procedure as are the checks and grading routine. This procedure has been compiled and is in the library assign in this directory. To provide the mapping to it use the pulldown to map to this library or execute from the command line the unix command

```
vmap assign /rcc4/homes/faculty/degroat/ee762_assign/assign
```

In either case you can use a logical name of your choosing. The only constraint is that the logical name you give it must match the name you put in the LIBRARY clause and also in the USE clause. This provides the logical mapping such that the library clause in the test bench will know where library assign is located. This must be done prior to compiling the test bench. The procedure will also do the grading of this assignment. Note that textio is used to read the stimulus and expected results from the file in my directory.

Turn in:

1. A copy of the VHDL code
2. A listing showing the results of simulation
3. A zoom->full_size waveform
4. The waveform for a couple of cycles where the value of the data can be seen on the bus.

Be sure that the final score comes out!!!!!!!!!!