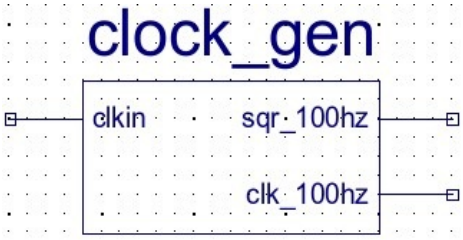
ENGR-433
Lab 2, note 2

You have been working hard on this assignment. Some of you have things working and others do not. While conceptually fairly straight forward, creating the 100hz clock has been time consuming I think and may still be troubling some of you. So further comments on it.

First, to confirm if/or not your clock is working you need to look at the divided down signal. Unfortunately we can't just stick a scope probe into the FPGA and look at various circuit nodes. What you need to do, and some of you have been doing this, is to route the signal from circuit nodes that you wish to observe to a connector on the FPGA board where you can attach a scope probe. For this lab use the extout0, extout1, etc that can be monitored on the header connector labeled EXTOUT. Plug a short hook-up wire, an inch or two long, into the connector and grab the other end with the scope probe. The output voltage should range from about zero to 3.3v.

If you try your divider circuit, it doesn't work, you do some checking and don't find the problem, I am supplying a pre-made circuit that can be installed as a component in your design. I would like you to get further than just trying to make the clock divider work. I have created a component that divides the 50Mhz clock down to 100hz as a square wave. It has one input, the 50Mhz and two outputs. The symbol for this circuit is shown below:



The sqr_100hz output is a direct connection to the Q output of the last flip-flop in the divider chain. (The clk_100hz output is the sqr_100hz with a bufg component added.) Use the clk_100hz output to drive the clock input of other flip-flops or counters. If you wish to take the 100hz off the chip to look at it with a scope use the sqr_100hz output (if you connect an obuf to the clk_100hz signal you will get an error. It won't synthesize).

To use this component in your lab2 design do the following:

1) Locate the folder (directory) that has the many files created by ISE as you synthesize the design. There should be a file whose extension is .xise, a file with extension .sch, and many more.

2) Download from the class web page a file named "lab2_20_clock_gen.tar" into the folder for your project. This is what we call a tar file in Unix/Linux land. It is functionally like a .zip file. Put this file into the directory you identified in step one above.

3) Use the file manager to navigate there, select the file, and select extract. Alternatively, open a terminal window, use the cd command to make the folder where your ISE project files are located the current directory, and then enter this command:

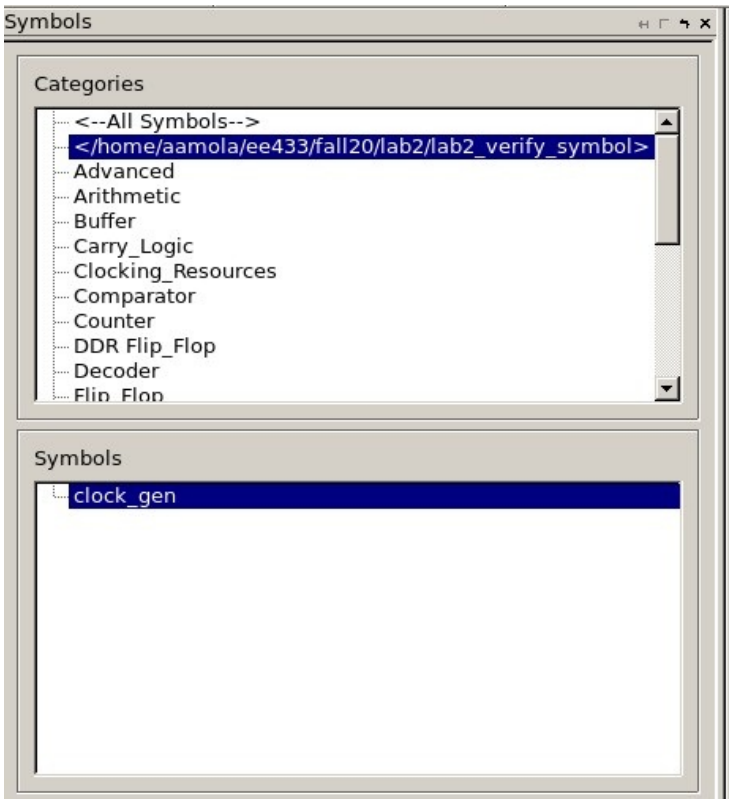
tar -xf lab2_20_clockgen.tar

Either method will extract these two files:

clock_gen.sch and clock_gen.sym

which are a schematic for the circuit and a symbol.

4) To place this circuit in your design, click Add Symbol as if you were planning to add a component from the Xilinx library. In the Symbols panel at the left, in the upper window titled Categories, there should now be an entry at the top which is a path to the folder where your design is. Click on that. In the lower window the component name clock_gen should appear. Click on that and place it in your schematic.



5) Connect clkin of the component to master clock. (If an ibufg and I/O Marker for master clock are not yet in your design then add those).

6) Connect the clk_100hz output of clock_gen to the clock input of flip-flops and counters that need a 100hz clock input.

(If your own clock dividing circuit is synthesizing but just doesn't work you shouldn't need to remove it to try the clock_gen component as long as you don't have an output from each circuit driving the same circuit node)

Below is a simple test circuit showing instantiation of clock_gen into a circuit where clkslow is routed to the extout connector: