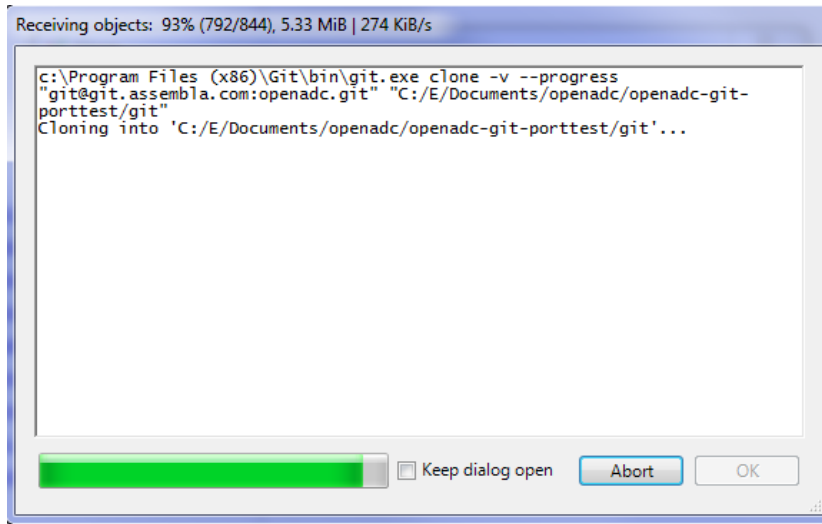


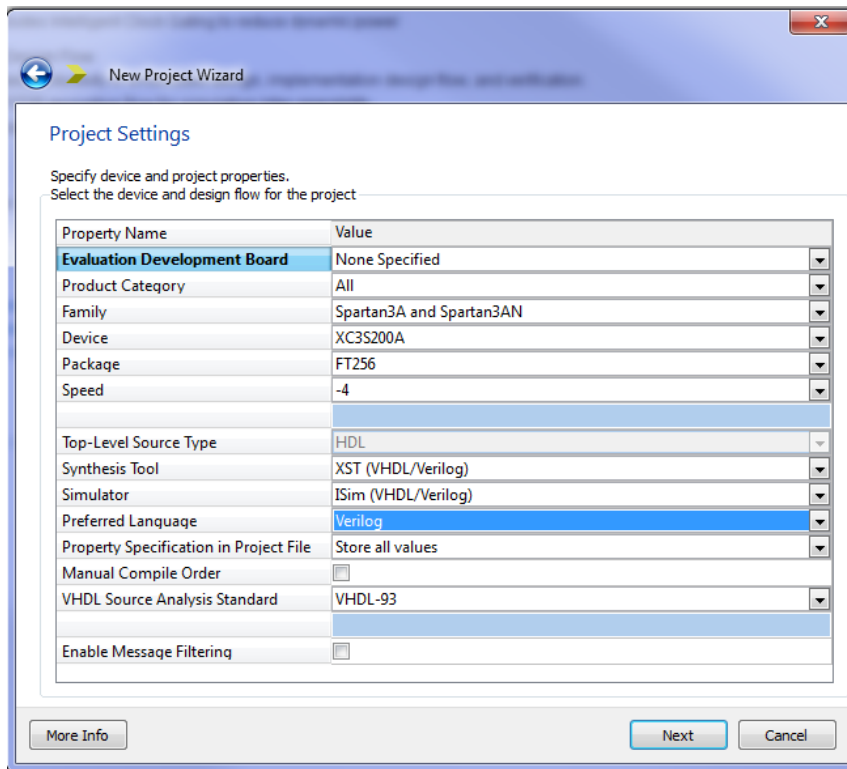
Porting Guide for OpenADC

Getting Started

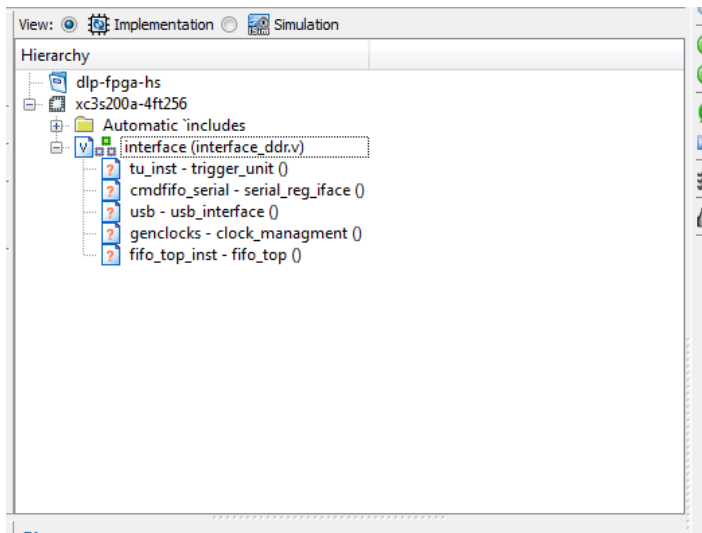
To begin with, you'll need to get the code. For now just pull from GIT:



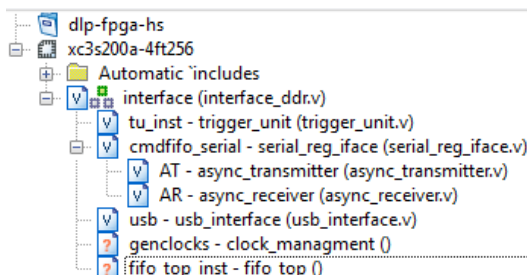
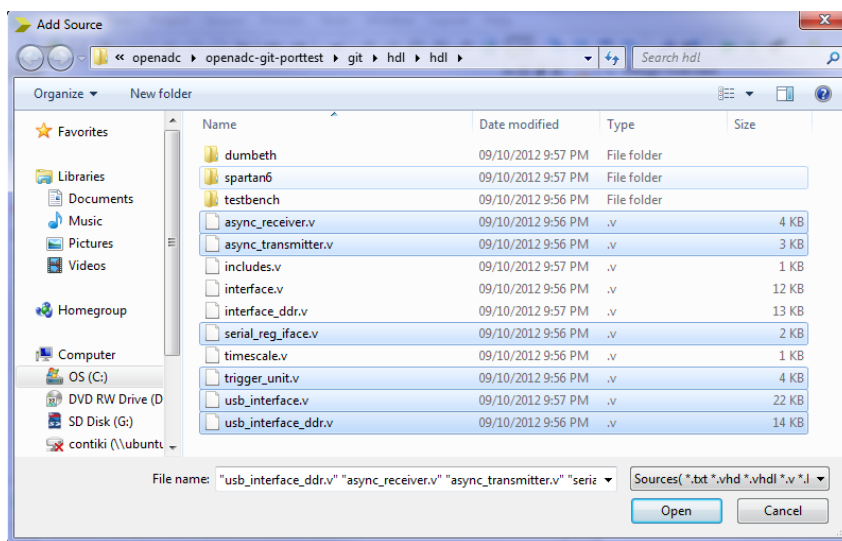
Next, we will create a project in Xilinx ISE Project Navigator.



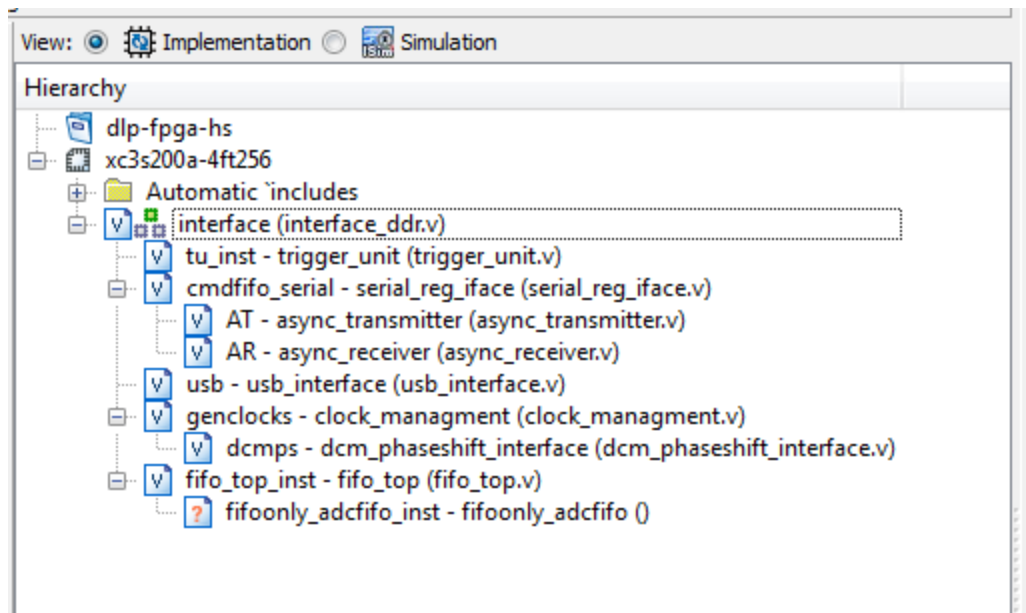
Add the top file: interface_dds.v (even if you don't use DDR!!). It should now look like this:



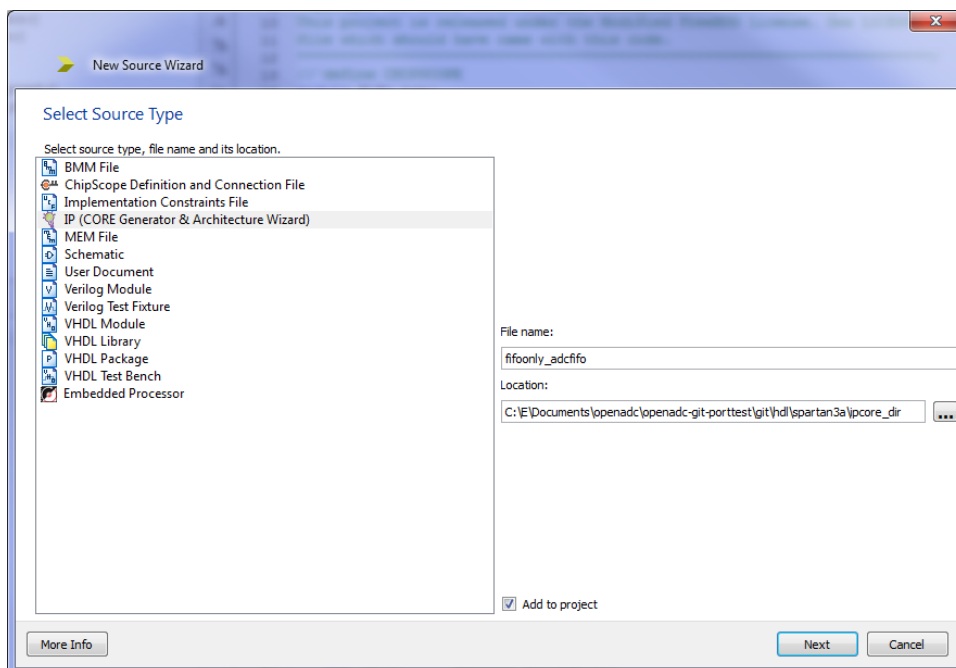
Add some more sources. If you add sources you don't need just delete them from the project:

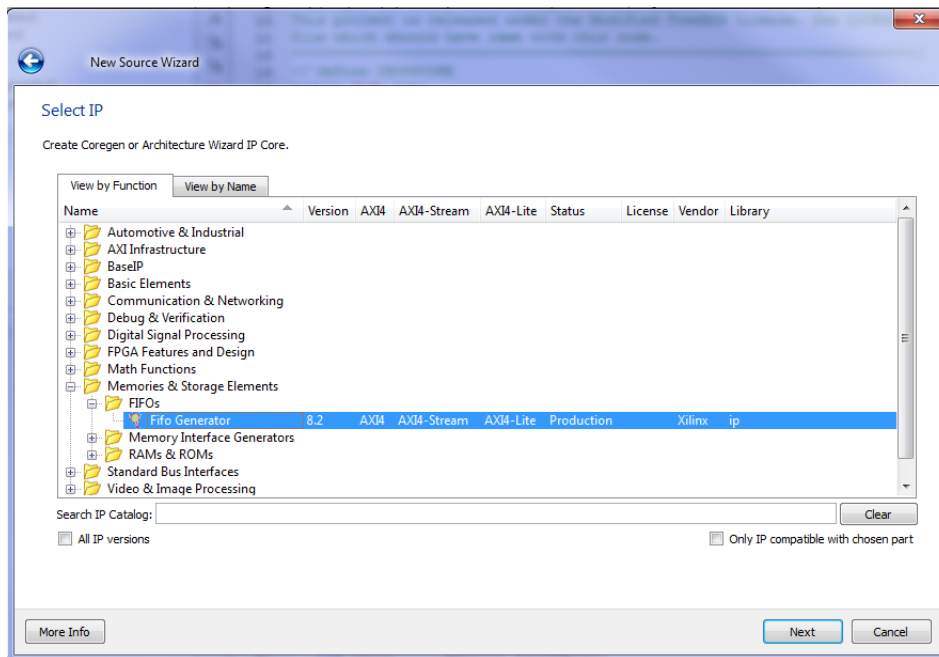


There are a bunch of source files in the 'Spartan6' subdirectory. These files may need to be rewritten for different targets. For now just add them as-is if to see how far off they are. Everything should now look like this:



The final source, the `fifoonly_adcfifo` you need to generate with ISE CoreGen. The following screen-shots show this setup:

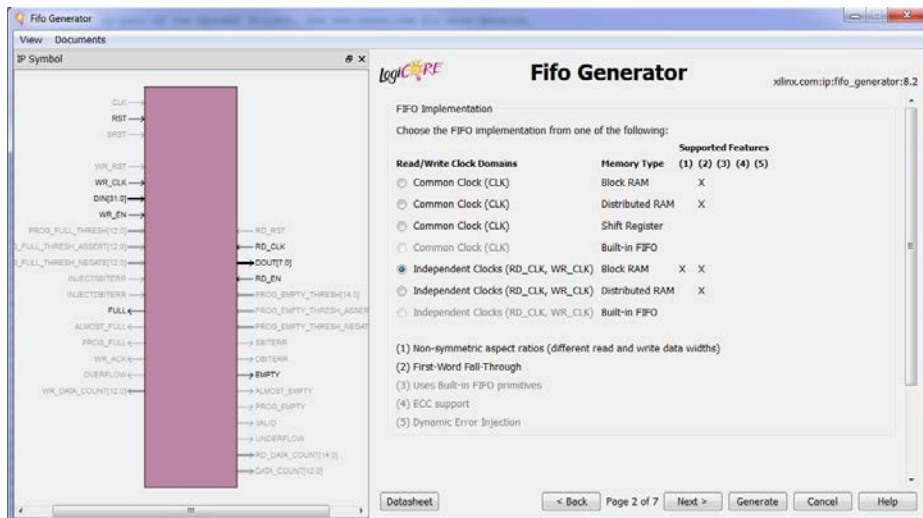




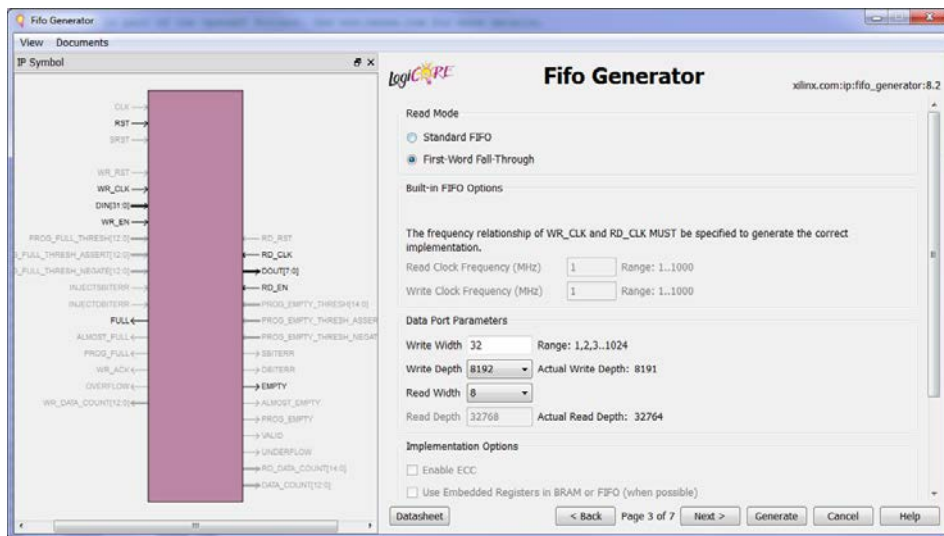
First page: Select 'Native' Interface Type:



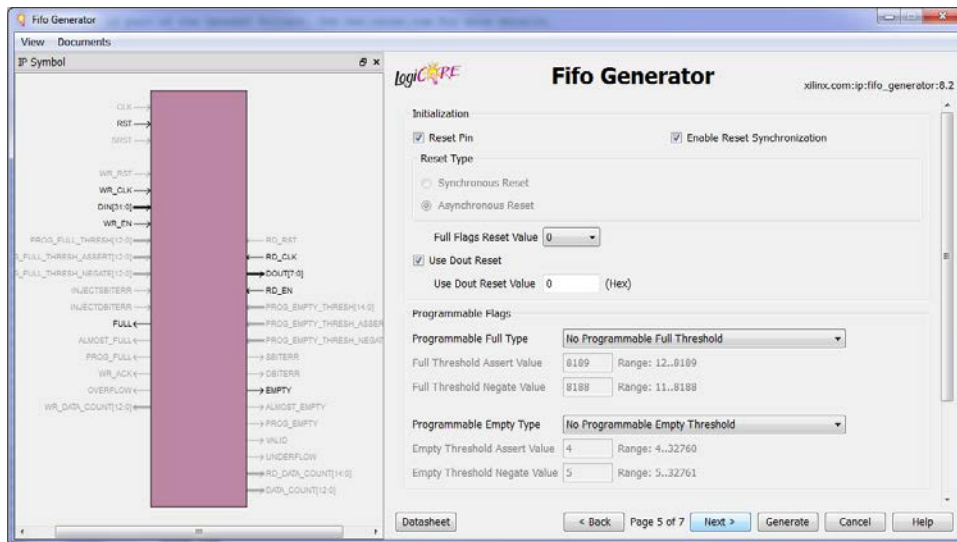
Second page: Select 'Independent Clocks, Block RAM':



3rd page: Select 'First Word Fall-Through', 'Write Width=32', 'Read Width=8'. Select a write depth – exact value depends on your HW, here I've used 8192, on a smaller FPGA select a smaller number. Total samples = this depth x 3.

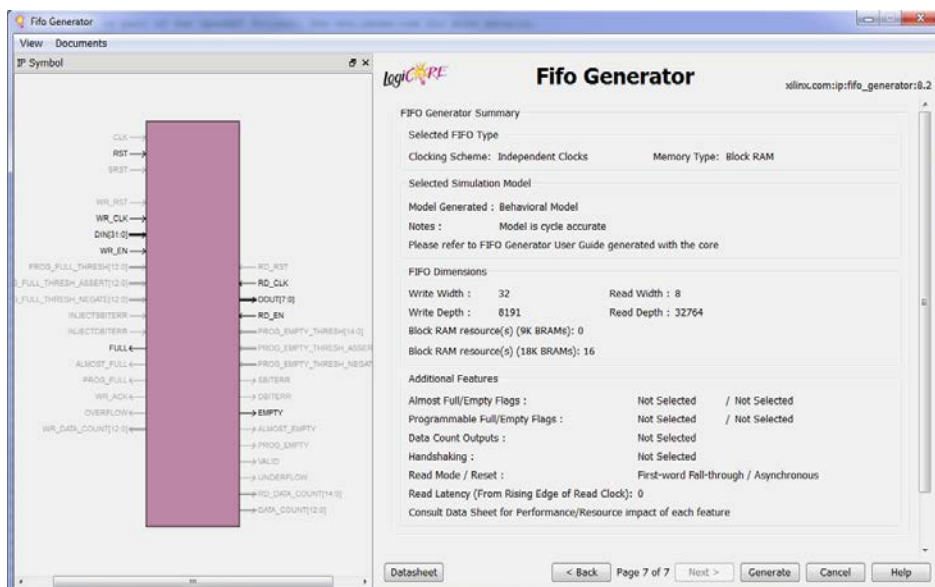


Fourth Page: Reset pin, reset sync, reset Dout to 0. These should all be defaults. No programmable flags.

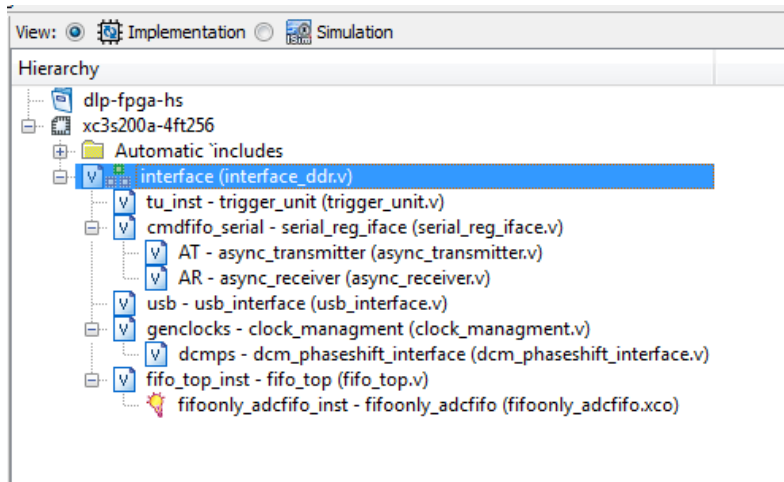


Sixth page: Don't select anything (sorry no screen-shot)

Seventh page: Summary. Can see number of block rams being used here & see how your design is fitting.

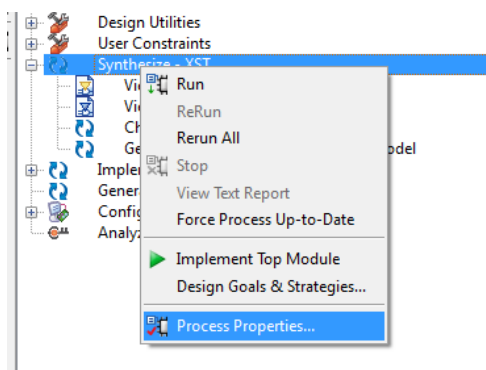


Once you generate this, your file should now look like this:

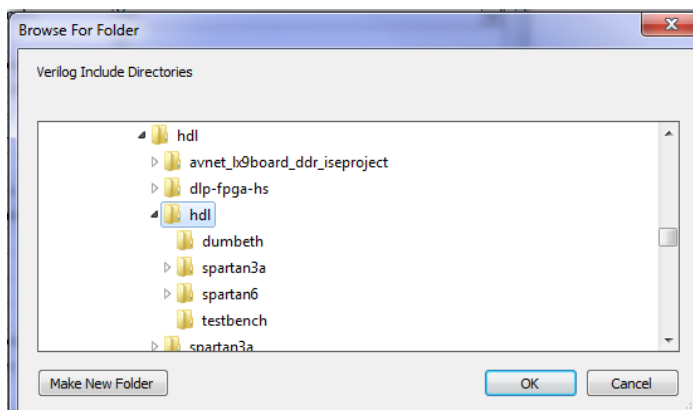


Compile Settings

Click on Synthesis Properties:



And change the 'Verilog Include Directories' to point to the hdl folder (where include.v is):



See What Happens

Alright... run the synthesis process & see what breaks. I don't have much advice here – then run implement after that.

Fix all the errors & check through warnings, although it's still pretty dirty so generates a lot of warnings anyway! You may want to compare with the original project.

Setup Hardware Pins, etc

Finally, setup your hardware. Be sure to adjust:

- Clock frequency defines in includes.v (used for UART baud rate calculation)
- UCF File (create it!)
- Possibly DCM settings