Project Assignment #3 **DUE: Monday, April 14th**

1) Copy the files **pr_step3.vhld and ps3_list.do** and **ps3_wave.do**
from directory ~degroat/ee762_assign or download from the web.

2) Write a STRUCTURAL description for a 8 bit ALU using your 1 bit ALU slice.
   a. Write an architecture using 8 component instantiation statements
   b. Write an architecture using component instantiations for the 0th and 7th slice and a Generate Statement for the remainder.
   (reference: Navabi - page 130, Figure 5.22)
   c. Write an architecture using nested generate statements
   (reference Navabi, page 131, Figure 5.25)

Suggestion: Get the architecture using the 8 component instations working first, then step b, and then step c. You will need to use 3 configuration (FOR label:) statements, one for each architecture. (reference Navabi *pp127-140, specifically fig 5.33*).

Note that <u>all three architectures will use the same entity</u>. Instantiate all three architectures into the testbench for simulation. In this way you can easily verify that the output from all three architectures is the same. Note that the output of each "architecture" is a different signal (Za,Coa,Zb,Cob,Zc,Coc). Make sure to list these three output signals as shown below.

The easy way to do this is to use the ".do" files. These files contain commands for the simulator that will set up the list and wave windows as desired. Start the simulator (vsim) and load the "ps3_tb" for simulation. Make sure to use "ns". Also, in the *simulator_options* set the "default radix" to hexademical. Then use the "**main**" simulator window to bring up waveform and list windows (View→List *and* View→Wave). Do not bring these windows up from the "signals" window. When they come up they will be blank. Once these windows are open, go to each in turn and use the "Load Format" option on the File pulldown to load the setup. Use the ps3_wave.do file for the Wave window and the ps3_list.do file for the List window. Now you run your simulation.

3) <u>Turn in</u>: **READ THIS!!!!!**

a) copy of the complete VHDL source i.e. (<u>for all three architectures and testbench</u>).

b) copy of VHDL code for generic unit, and single bit slice.

c) copy of a complete waveform on 1 page.!!!!! The simulation needs to run for 17000 ns to be complete. The easy way to do this is to enter the command >run 17000 ns in the main command window

d) copy of a file listing the results of the simulation as set up by the .do files.

BE SURE THAT THE PRINTOUT DOESN'T LINE WRAP!!!!!!

ns	delta	oper	pval	kval	rval	a	b	cin	za	zb	zc	coa	cob	coc
0	+0	OP_A	C	F	C	00	00	0	00	00	00	0	0	0
0	+1	OP_A	C	F	C	00	00	0	00	00	00	0	0	0
100		OP_A	C	F	C	00	10	0	00	00	00	0	0	0
200		OP_A	C	F	C	11	00	0	10	10	10	0	0	0
300		OP_A	C	F	C	10	10	0	10	10	10	0	0	0
400		OP_B	A	F	C	00	00	0	00	00	00	1	1	1
		•												
		•												
3000		OP_BMINA	9	2	9	10	10	0	00	10	11	0	0	0

You can save the listing out to a text file and then print it. You can view the waveform file in LINUX and then choose to print it.

Note: The values in the table above are <u>not</u> the correct results.. The listing file and waveform also have an error signal that should not have any spikes on it. If your waveform has spikes on these signals, your model has an error. If your simulation has error you do not earn full credit. Whenever the error line spikes, the output does not agree with what is expected.