Homework #2 – Due 9/11/19
Please submit to the Dropbox on MyCourses

1. On your own, review the tutorial entitled Quartus_VHDL_Introduction. If you haven't already, you will complete it in lab, but you will also need to reference it for creating VHDL projects in Quartus.

Complete the rest of the problems with your homework group. Please use screen shots and/or snipping tool to combine each (and every) part of the homework into a single Word file.

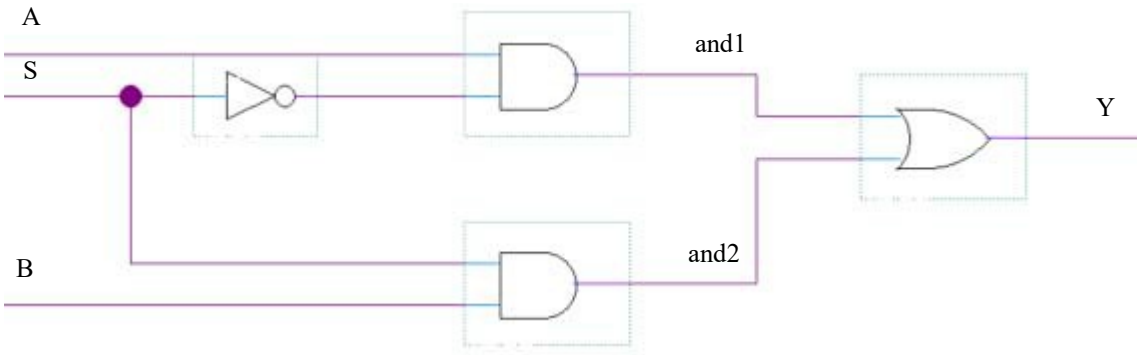
2. Design a 4-input, 1-output function. The inputs are named A1, A0, B1 and B0. The output, named **GT**, will be a 1 if the binary number, formed by A1 and A0 is greater than the binary number formed by B1 and B0. Otherwise the output will be 0.
   a. Complete the truth table for the function **GT.** Submit the truth table.

A1	A0	B1	B0	GT
0	0	0	0	
0	0	0	1	
0	0	1	0	
0	0	1	1	
0	1	0	0	
0	1	0	1	
0	1	1	0	
0	1	1	1	
1	0	0	0	
1	0	0	1	
1	0	1	0	
1	0	1	1	
1	1	0	0	
1	1	0	1	
1	1	1	0	
1	1	1	1	

   b. Directly from the truth table, write the unsimplified equation for **GT.** The equation should be in the sum of products form**.** Submit the equation.
   c. In Quartus, write the VHDL code (entity and architecture) for the equation generated in part b. Use simple assignment statements (AND, OR, NOT, etc..). Submit the code.
   d. Create a simulation waveform file and simulate all 16 input combinations. Verify your VHDL is correct with this simulation. Submit the waveform.

e. Recompile the design with equation generation enabled (directions are at the end of this handout). View the generated equation, keeping in mind that & = AND, ! = NOT and # = OR. Does the equation match your original equation? Submit the equation and your answer.
f. Go back to your .vhd file and comment out the original architecture. Write a new architecture using the equation from part e. Submit the code.
g. Resimulate and verify the results. Submit the waveform.
h. Based on the results, what conclusion(s) can you make about the synthesizer? Submit your answer

3. Consider the circuit below:

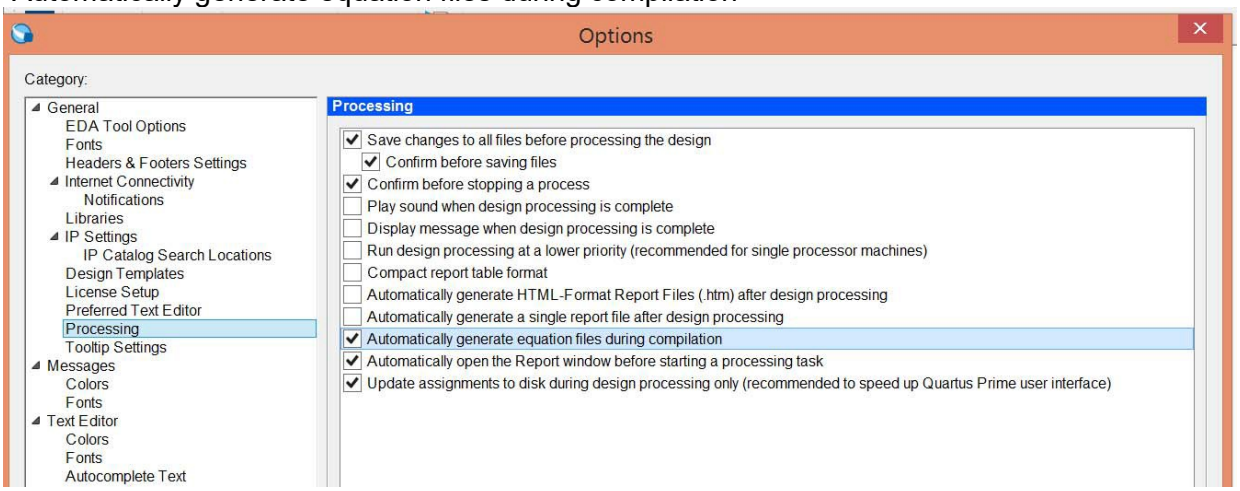


a. In quartus, write the VHDL code (entity and architecture). Use simple assignment statements (AND, OR, NOT, etc..). Submit the code.
b. Create a simulation waveform file and simulate all 8 input combinations. Verify that your VHDL is correct with this simulation. Submit the waveform.
c. Generate the equation. Submit the equation.
d. Comment out the architecture and rewrite the VHDL using two internal signals named and1 and and2. Each signal will be the output of an AND gate as shown above. This architecture will have 3 assignment statements. Submit the code.
e. Examine the equation. Is it the same as the equation in step c? What can you conclude about the synthesizer? Submit the equation and your answers.
f. Now rearrange the order of assignment statements in the architecture so that Y is assigned before and1 and and2. Submit the code.
g. Recompile and examine the equation. Submit the equation.

h. Rearrange the order of the assignment statements in the third possible order. Submit the code.
i. Recompile and examine the equation. Submit the equation.
j. Based on the three equations, what conclusion can you make about the order of statements in VHDL? Is this the same or different than in the C programming language?

To generate the implementation equation, do the following:

- Choose tools > options and then click on Processing. Check the box entitled "Automatically generate equation files during compilation"



- Compile the VHDL. In the Table of Contents, click on Analysis & Synthesis and then Equations (see below).