1. (40 points) Design a circuit that implements a stack in hardware. Your circuit should be able to stack up to six values of 3 bits each. It should have two control inputs, push and pop, two status outputs full and empty, plus 3 data inputs and 3 data outputs. When the push input high, the circuit should stack the value on the data inputs on top of the current stack of values. When pop is high, it should pop the top value off the stack. The data outputs should always be equal to the value on the top of the stack. If both push and pop are asserted at the same time, the circuit should replace the top value on the stack with the new input value. Pushing a new value on to a full stack should push out the value at the bottom of the stack. Popping an empty stack should have no effect. Include a synchronous reset, which makes the stack empty initially. You may use multiplexor and/or demultiplexor components in your design, in addition to gates and flip flops.

Design your circuit using the schematic editor and simulate it using the functional simulation mode to verify that it works correctly. Analyze your circuit to determine the fastest clock period for which it will work correctly, assuming a setup time of 2 ns for the flip flops, a clock skew of 1 ns and a maximum propagation delay of 1 ns for each flip flop and gate. Try running the circuit at this clock period using the unit delay mode simulation, with a simulation precision of 1 ns and determine if it works correctly at this clock period or not. Also try running it for a clock period that is 4 ns smaller to see if it fails. Turn in your schematic and simulation results.

2. (20 points) Design a circuit consisting of three 4 bit registers A, B and C connected by a bus. Each of the three registers has a load input that causes it to load the value on the bus, an enable input, that causes it to place its value on the bus and a synchronous reset input that clears it. In addition, A has an increment input that causes it to increment its value by 1, B has a negate input that causes it to change its value to the 2's complement of the previous value, and C has a rotate input, that causes it to rotate its stored bits to the left by one position.

Draw a schematic for each of the three registers. Then draw a block diagram showing how they would be connected together by the bus. Show each register as a block, labeled with its name (A, B or C) and with all its inputs and outputs labeled within the block. Also, clearly label the inputs and outputs to the overall circuit. These will include the clock, reset, loadA, loadB, loadC, enA, enB, enC, incA, negB, rotC.

Finally, draw a timing diagram showing a sequence of input values over 7 clock ticks that result in the value -4 being stored in B. Do this while only incrementing A one time. Show
the values of A, B and C in your timing diagram (in bus form, not as individual bits). Remember to reset the registers at the start.