1. (20 points) Write a VHDL entity specification for the 1 bit comparator component from problem 8 of homework 4, along with a behavioral specification of the logic. Simulate your component for all input combinations.

Write a VHDL entity specification for a four bit comparison circuit and a corresponding architecture. Your circuit should use four copies of the 1 bit comparator component. Use structural VHDL to combine these four components to create the comparison circuit. Simulate your design for the input values specified in problem 8 of homework 4.

2. (15 points) Enter a schematic of a D-latch and simulate it using the unit delay mode with a simulation precision of 1 ns. Demonstrate how the D-latch can be made to oscillate by changing the D input just before the control input drops. Turn in your schematic and simulation output.

3. (12 points) Consider the sequential circuit shown below. Assume that the flip flops have a setup time of 2 ns, a hold time of 1 ns and a propagation delay between 1 and 3 ns. Also assume that the maximum clock skew is 1 ns and that all the gates have a propagation delay between .5 and 2 ns. What is the shortest clock period for which we can be certain that there are no violations of setup times, assuming no changes at the input X? Is the circuit subject to hold time violations? If so, what would you do to eliminate the hold time violations? If the clock goes high at time 0, during what time period must X be stable to ensure that there are no violations of setup and hold times? During what time period is it possible for the output to be changing. If the clock could change anytime between $t=-1$ ns and $t=+1$ ns, how do the last two answers change?
Repeat the above analysis for the circuit shown below. That is, (a) determine if there are any hold time violations, and if so, explain how they can be fixed, (b) determine the fastest clock frequency for which there are no setup time violations, (c) determine when, relative to a rising clock transition each of the inputs must be stable to ensure correct operation. Note that for this circuit, the timing of an output change is dependent not just on the clock, but on the data input as well. How does this affect the construction of larger circuits that might use this component?

4. (12 points) For each of the sequential circuits in problem 3, construct a state table and a state transition diagram. Note that one of these is a Mealy model circuit and one is a Moore model circuit.
5. (12 points) Design sequential circuits that implement the state transition diagrams shown below. Note that one of these is for a Mealy model circuit and one is for a Moore model circuit.
6. (20 points) Design a serial multiply-by-5 circuit. Your circuit will have a single data input \( D \), a synchronous reset input \( R \) and a single output \( Q \). Whenever \( R \) is high, your circuit should output 0, but when \( R \) drops low, your circuit should treat input \( D \) as a binary value, received with the least significant bit first. Your circuit should output a value which is equal to the input value, multiplied by 5. So if you receive the values 0001101 (where the rightmost bit is received on the first clock tick following reset, then the second bit from the right, etc.) your circuit will output 1000001.

Design your circuit as a Moore model circuit. Carry out each of the following steps
(a) create a state transition diagram for the circuit,
(b) determine the number of flip flops needed and select a state assignment,
(c) write logic equations for each flip flop input,
(d) write logic equations for the output \( Q \),
(e) draw a schematic for your circuit and
(f) do a timing analysis of your circuit; specifically, determine if there are any hold time violations and if so, explain how to fix them, determine the smallest clock period that it can handle without risk of setup time violations, determine when the input must be stable relative to the rising clock edge and determine the time period following a clock transition when the output can be changing (use the timing parameters from problem 3).
7. (25 points) Write two different VHDL specifications for the state machine shown below. In the first version, define a state type (as on page 4-27 of the course notes) using the symbolic state names shown on the diagram. Use the symbolic state names in the code, as on page 4-27. Write the second version in the style shown on page 4-26. That is, define explicit signals for the state variables and code the next-state logic explicitly. Use the following state assignment: peach=000, pit=001, plum =010, prune=011, nut=111.

Simulate both of your designs. For your simulations, design a test sequence that exercises all the transitions in the state diagram. To do this, you must first force the circuit into a known state. What initial sequence of input values will guarantee that the circuit starts in a known state?

Turn in listings of your VHDL source code and the simulation output from both runs. On the simulation output, label every clock tick to show the state transition that it exercises. Do this by writing a label consisting of the current state name and the input values that cause the transition to the next state.