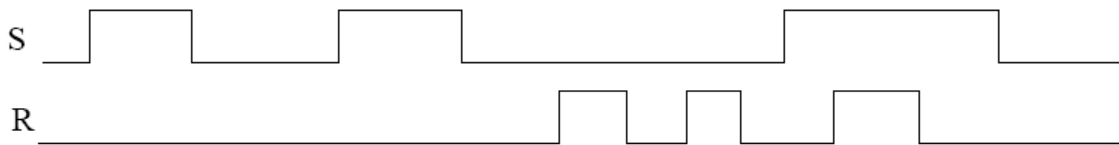


ECE331 - Fall 2007.

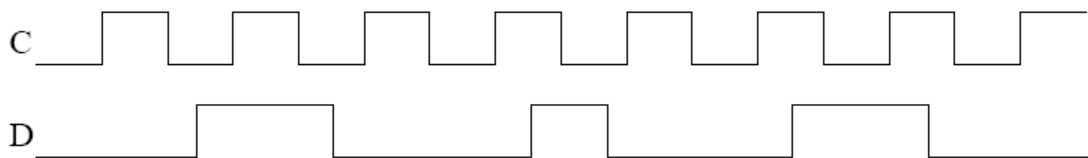
The next two pages contain drill problems (from previous semesters). If you feel you can't work the homework problems, start with these. We won't grade them, but I'll post the answers on the website. DISREGARD problem 4 for now.

1) Flip-flops

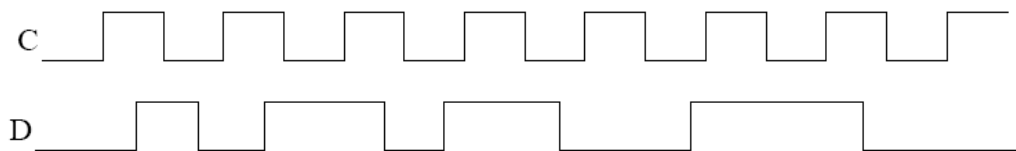
- a) The figure below shows a partial timing diagram for an SR Latch. Draw the corresponding timing diagrams for the two resulting output waveforms Q and Q_1 . Initially the latch is not set.



- b) The figure below shows a partial timing diagram for a D Latch. Draw the corresponding timing diagrams for the two resulting output waveforms Q and Q' . Initially the latch is not set.



- c) The figure below shows a partial timing diagram for a positive-edge-triggered D Flip-Flop. Draw the corresponding timing diagrams for the two resulting output waveforms Q and Q' . Initially the flip flop is not set.



2) Flip-Flops

- a) **JK Flip-Flop:** The textbook describes in section 7.6 a JK Flip-Flop. This flip-flop can be used to build other flip-flops. Construct 1) a D Flip-Flop and 2) a T Flip-Flop from a JK Flip-Flop. In each case use the graphical symbol from figure 7.17c and draw the schematic.
- b) **JK Flip-Flop in VHDL:** The VHDL code that describes a D Flip-Flop is given in the textbook in Figure 7.37. Use this code and modify it to create a JK Flip-Flop. Write a testbench capable of generating all possible inputs. Verify your design and testbench using Active HDL. Submit as solution:
 - Printout of your design source code verified using Active HDL.
 - Printout of your testbench source code verified using Active HDL.
 - Printout of timing waveforms showing values of inputs and corresponding outputs of your circuit.

Do NOT just copy a solution you found on Google. You would not learn anything from that.

- c) **JK Flip-Flop with Asynchronous Reset and Set:** Modify your code from above and add an asynchronous reset and an asynchronous set to your JK Flip-Flop. Write a testbench capable of generating all possible inputs. Verify your design and testbench using Active HDL. Submit as solution:
 - Printout of your design source code verified using Active HDL.
 - Printout of your testbench source code verified using Active HDL.
 - Printout of timing waveforms showing values of inputs and corresponding outputs of your circuit.

3) Shift Registers

This problem requires a bit of creativity! Chapter 7 in the book contains many useful examples.

- a) **Right Shift Register:** Construct a 4-bit shift register using D Flip-Flops that shifts its content to the right. The register should have one data input s_{in} (*serial in*) and one clock input clk . Its outputs are b_3, b_2, b_1, b_0 .
- b) **Bidirectional Shift Register:** Add another input sel (*select*) to your register. If $sel = 0$ the register should shift its contents to the right, if $sel = 1$ it should shift it to the left. Use basic gates (AND, OR, etc.) to realize this new functionality. For the D Flip-Flops use the symbol from figure 7.11b from the textbook.
- c) **Bidirectional Shift Register with Parallel Load:** Add 5 new inputs to your register from part b), namely d_3, d_2, d_1, d_0 . and p . If $p = 0$ the register should function like before. If $p = 1$ then the register should store the input presented on d_3, d_2, d_1, d_0 . and not perform any shift operation. This functionality is called *parallel load*.

4) State Sequencer Analysis

For each subsection below, analyze the sequencer shown on the last page. Use the three-step process to i) Write the output and next state equations, ii) Construct the Transition Table and iii) Construct the State Transition Diagram.

- a) Circuit 1
- b) Circuit 2
- c) Circuit 3