

ECE 2700 Lab 7

**Due the week of April 6, 2015 at the beginning of
your registered session (200 points)**

Objectives

To become proficient in building sequential circuits in Verilog and to understand how to use a timer to control an FSM.

1 Preparation

1. Read this document in its entirety.
2. Draw a circuit diagram or write out Verilog code for Part 2.
3. Draw a circuit diagram or write out Verilog code for Part 3.
4. Draw a circuit diagram for Part 4.

2 Up/Down Counter with Load

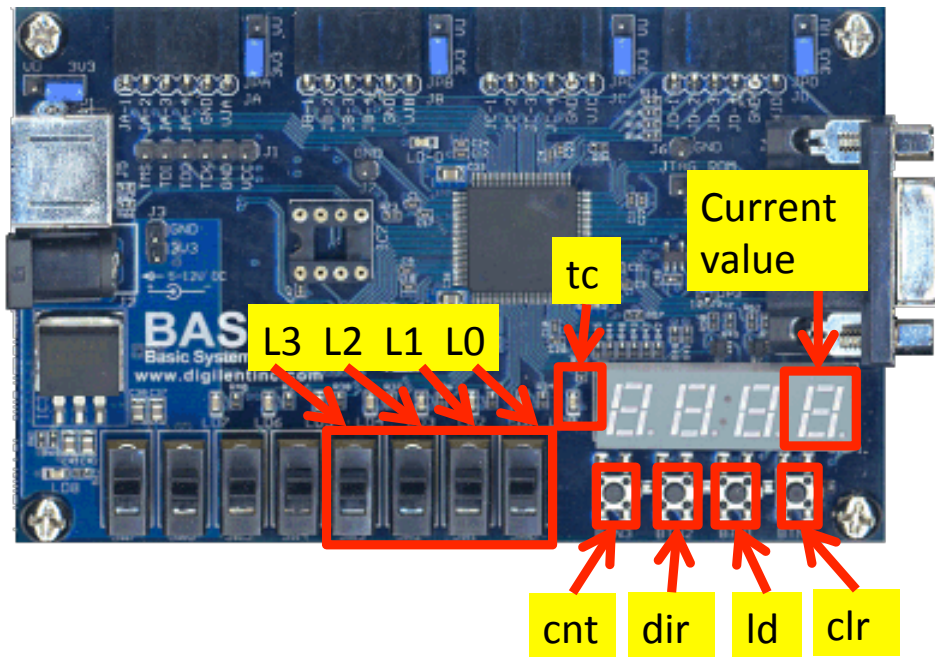
You are to create an up/down counter with load. A top-level diagram of the up/down counter is shown in Figure 4.68 of your main textbook (page 217). A top-level diagram of a counter with load is shown in Figure 4.69 on page 218 of the main textbook. You are to design a counter that has both up/down and load functionalities. Specifically, your counter should behave as follows.

- All operations are synchronous.
- Load has precedence over count.
- The counter will count up or down only if the count button is pressed.
- If the direction is 0, the counter will count up. Otherwise, it will count down.
- The counter value wraps around. That is, if the counter is counting up and the current value is 1111, the next value will be 0000. Similarly, if the counter is counting down and the current value is 0000, the next value will be 1111.
- The `tc` signal is asserted (high) when the counter is counting up and the count is 1111 or when the counter is counting down and the count is 0000.

Make sure that the FSM you create is safe (Section 3.6 in your secondary textbook).

Use the following declaration: (`clk`, `cnt`, `clr`, `dir`, `ld`, `L3`, `L2`, `L1`, `L0`, `tc`, `C`), where `clk`, `cnt`, `clr`, `dir`, `tc` are as defined in Figure 4.68, `C` is a 4-bit output, and `ld`, `L3`, `L2`, `L1` and `L0` are as defined in Figure 4.69. Note that we do not declare `L3`, `L2`, `L1` and `L0` as a vector since they will be individually connected to the switches.

Simulate your design. Once you are confident your counter works, show the results to the TA. Switch to implementation mode, add the clock divider, and create a `.ucf` file. The rightmost digit of the 7-segment display should show the current count. Refer to the figure below for the remaining buttons and switches required.



Recall that you may have to press a button for a little while before changes take place since the button only takes effect on the rising edge of the clock. Also, to count down, the `dir` button needs to be pressed for the entire duration. Show your results to the TA.

3 Automatic Down Counter

For this part of the lab, you are to create an automatic down counter similar to the one in Figure 4.72(a) on page 220 of your main textbook. Specifically, the counter should be loaded with a value of 3 and count down, then repeat. In other words, the automatic down counter should provide the following output sequence: 3, 2, 1, 0, 3, 2, 1, 0, 3, In addition, the `tc` signal should be asserted when the count is 0. Note that one does not need to press a button for the counter to count down (hence the term automatic—this counter is basically a timer). Simulate your design and show the results to the TA. You do not need to load this design to the board.

4 4-Digit 7-Segment Display

Previously, you saw how one could send signals to light up the different digits of the 7-segment display. To show all digits at once, you have to display each digit, one by one, fast enough that the human eye cannot detect the motion. You will use the automatic down counter, a number of muxes, and your 7-segment display code to accomplish this step. The main challenge in this part of the lab is to determine how. Once you finish writing your code, perform some simulations to make sure your code is correct.

Add the clock divider to your design. Switch to implementation mode and generate the bit file. *Hint:* Look inside `clockdivider.v` and identify a value that may need to be changed. It may take some trials and errors to determine the appropriate value. Show your results to the TA.

5 TA Checkoff

- (20 points) Prelab designs
- (60 points) Up/down counter with load

- (30 points) Counter functional on the Basys board
- (30 points) Automatic down counter
- (60 points) 4-digit 7-segment display on Basys board