Computer Systems Design - Lab 3

Handed Out: **Monday, 30 Jan, 2017**
Due: **Wednesday, 8 Feb, 2017 by 11:59 PM**
Submit via USF Canvas
This is an individual assignment. Teams are NOT allowed.
Late submissions will not be accepted

Objectives: To learn how to create and test an advanced sequential hardware design in Verilog and combine sequential and combinational components in a single design.

Problem: Design and simulate a programmable counter & BCD converter in Verilog.

Description: Design a programmable counter that outputs BCD and has a range of 0-99. The counter must count from zero to a user-input value. The valid input range is 0-99; if a higher number is input, the counter must not output beyond 99 (decimal). The output from your design should be in binary-coded decimal. You will need to design a binary counter that counts *at least* to 99_{dec}, and a module which converts the binary output to two-digit BCD. A single-bit input, called *run*, determines the operation of the counter. When *run* is '0', the counter is set to zero and does not count. The user can set the maximum value here, using an input called *max_count*. When *run* is set to '1', the counter starts counting from zero, and stops when it reaches the maximum. Any changes to *max_count* are ignored when *run* is set to '1'.
Your design will have two 4-bit outputs; *digit_1* and *digit_2*. Each is used to output one BCD digit. *Digit_2* is the tens digit, *digit_1* is the ones digit.

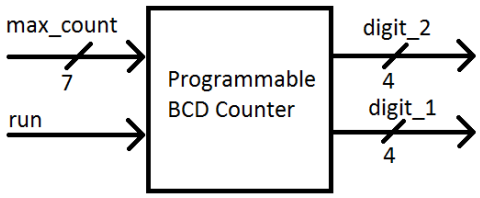

Figure 1: Counter Top-Level Block Diagram

Port	Width	I/O	Function
max_count	7	I	Set the maximum counter value
run	1	I	Set run mode; 0=set, 1=run
digit_2	4	O	Tens Digit BCD Output
digit_1	4	O	Ones Digit BCD Output

1. (30pts) **Design:** Create a Verilog design which will provide the required functionality.
2. (20pts) **Test:** Exhaustively test the design to ensure that it works under all conditions.

Note: Code templates for this project have been uploaded to Canvas. Use of these templates is optional.

Design Constraints: The design must properly function given the following constraints:

- The input for the max count value is called **max_count** and is 7 bits wide.
- The input for the clock is named **clk** and is 1 bit wide.
- A 1-bit input called *run* determines if the counter is running.
- The two outputs are each 4 bits wide, and are called **digit_1** and **digit_2**, and are used to display the BCD output.
- The input clock *must* be 100MHz. You will have to program this in the testbench.
- The absolute maximum value for the counter is **99**.
- You must design at least two sub-modules: a programmable counter, and a BCD converter. How you construct these modules is up to you.

Lab Report: You must submit a written lab report using the template posted on Canvas. This is NOT to be included INSIDE your zip file, but uploaded as a second file in the submission.
All reports MUST be submitted in PDF format. No exceptions.

Deliverables: Submit your report PDF and the code ZIP file separately on Canvas. The PDF should be named <netid>_lab02_report.pdf. To generate the code ZIP file:

- Select **Project** → **Cleanup Project Files...** from the menu.
- Select **OK** on the pop-up box.
- Select **Project** → **Archive...** from the menu.
- On the new window, make sure **Exclude Generated Files From Archive** is *selected*.
- Choose a location for the archive, name it <netid>_lab02.zip.
- Select **OK**

Insert any images and waveforms into the report PDF. Submit the report PDF and source code ZIP separately on Canvas. If you wish, you may include copies of all images in a separate ZIP file, submitted through Canvas. Name it <netid>_lab02_images.zip.

No re-grading will be done for failure to follow the guidelines. Points will be lost, never to return.