Lab 6: Stopwatch

EE 316: Digital Logic Design

1 Overview

This lab is the final lab of the semester and requires you to draw on concepts you learned throughout the semester to make a final product. By the end of this lab, you should be able to:

- design a complex system from given specifications.
- synthesize and implement information in previous labs to solve a problem.

This lab is worth 50% more than the other labs. While this lab is indeed difficult, we believe that, with the content you have learned through lecture and previous labs, you are extremely well-placed to complete this lab.

2 Design Specifications

In this lab, you will design a stopwatch on the seven-segment display. The stopwatch should run at resolution 10ms and display as XX.XX on the seven-seg. For example, if the stopwatch is counting up from 00.00 and displays 12.34, exactly 12 seconds and 340 milliseconds have elapsed since the stopwatch started.

2.1 Modes

There are four modes of the stopwatch determined by SW[1:0]:

- Mode 00 - Count up from 00.00 and stops at 99.99 (does not reset until reset button asserted)
- Mode 01 - Count up from XX.00, where XX is an externally loaded value from SW[15:8], and stops at 99.99
- Mode 10 - Count down from 99.99 and stops at 00.00 (does not reset until reset button asserted)
- Mode 11 - Count down from XX.00, where XX is an externally loaded value from SW[15:8], and stops at 00.00

2.2 Inputs

You will need the following inputs to your system:

- Center button - Start/stop button to control when stopwatch runs. If the stopwatch is stopped and the button is pressed/released, the stopwatch should start running. If the stopwatch is running and the button is pressed/released, the stopwatch should stop. If the stopwatch has reached 99.99 or 00.00 and the button is pressed/released, nothing happens.
- Up button - Reset timer to starting value when pressed. If the stopwatch is in "external-load mode" (modes 01, 11), the stopwatch should load the top two 4-bit digits from SW[15:8] into the stopwatch's tens and ones places, respectively. For example, if SW[15:8]=8'b10001000 and the stopwatch is in either mode 01 or 11, the stopwatch should display 88.00 after the reset button is pressed/released.
- SW[1:0] - Mode select
- SW[15:12] - External value, tens-digit. If greater than 9, set to 9.
- SW[11:8] - External value, ones-digit. If greater than 9, set to 9.

Throughout this lab, be sure to document your work and note key design choices and milestones in your development. You will need to write a report summarizing your design process, as described in the Submission section.

3 Hints and Suggestions

1. Don't start right away, but don't start the night it's due. Come up with a few designs, and discuss your ideas with the TAs.
2. Think about whether you need a button debouncer, and how that might fit into your design.
3. Think about how certain modes can be combined in unique ways. Are modes 00 and 01 really all that different? What might be the differences? Are they significant enough to warrant separate modules?
4. Think about ways you can implement the "roll-over" functionality without using a whole bunch of if-statements - e.g. "don't-cares" in case statements.
5. Think about how a stopwatch will be "reasonably" used and make testcases based on that. Since the problem does not mention any testcases in particular, you can choose what happens for any edge cases you think of. However, be prepared to justify the behavior of your stopwatch at those edge cases.
6. Try to reuse as much code from previous labs as you can. This will prevent you from spending time re-inventing the wheel.
7. Good luck!

4 Submission

For this lab, please submit the following:

1. A .zip file of your design files, any testbenches you wrote, and your constraints file.
2. A PDF file with the following:
   (a) a 400-500-word PDF report detailing your approach to the problem, design process, design choices, key milestones, and key setbacks;
   (b) a short 50-100-word summary of your system;
   (c) and a high-level block diagram of your system, including any modules you re-used from old labs. Your block diagram must be specific enough so that a digital logic engineer can understand your design, but not so detailed that it is overwhelming to look at.

Submit both files together as one submission on Canvas. Be sure to schedule a checkout with a TA.

4.1 Checkout Process

The checkout process involves a 10min one-on-one meeting with a TA the week after you submit your lab. You will schedule your checkout the week the lab is due. The checkouts will be held in EER 0.716, the lab room. During the checkout, the TA will ask you to demonstrate the functionality of your code on the FPGA board, look at your source files, and ask you questions regarding the lab. The questions may range from syntax questions (e.g. "what does .a(a) mean?") to high-level design questions, such as "why did you choose these testcases?" You should be able to answer the questions in enough detail to show the TA that you understand the main concepts in the lab.

When it is time for you to check out, have your lab report (PDF file) and Vivado project open, and flash the board before the TA comes to meet with you. If you aren't prepared at your scheduled time, the TA will move on to the next person. If you miss your checkout time, you will need to re-schedule with the head TA.

4.2 Grading

Your grade on the lab will be based on a combination of your code's functionality on the board, your documentation (PDF file), your responses to the questions in this lab document (see section 4.3), and your oral responses to the TA during checkout. Late submissions will incur a 5% penalty per day up to a week, after which your grade will be a zero.