Laboratory 3: Counter and Switch (Debounce)

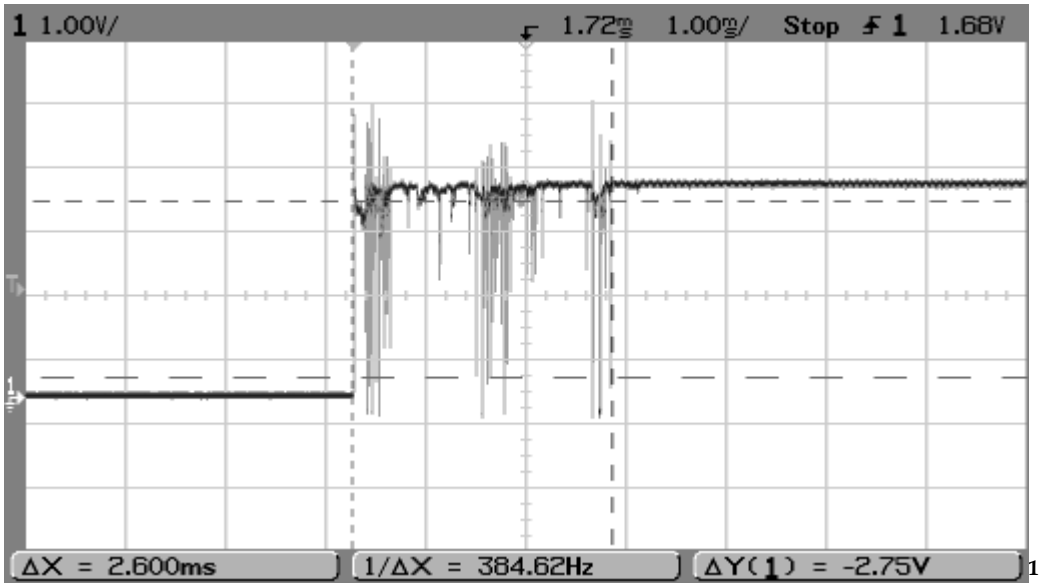
Objectives

1. Synthesis FPGA
2. Able to design debounce switch and input
3. Able to design up and down Counter

Background

Switch and Bounce

Mechanic switches and relays have a common issue call contact bounce (aka. chatter). Switch and relay contacts are usually made of metals. When the contacts strike together, their momentum and elasticity act together to cause them to bounce apart one or more times before making steady contact. (Imagine a ball falling on a fall, it would bounce several times before coming to a complete stop.)

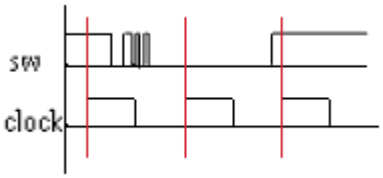


[1]

There are several ways to debounce. Debouncing methods include using capacitor, SR Latch, or Low-pass filtered schmitt trigger. However, those methods usually required special hardware.

To debounce without using special hardware, we can use software method by resampling for input several times.

[1] Image taken from https://upload.wikimedia.org/wikipedia/commons/thumb/a/ac/Bouncy_Switch.png/400px-Bouncy_Switch.png



Please also note that there is a metastable issue. To avoid this, it is generally advised that two D flip flops be placed between the input and the digital circuit.

Exercises

1. Create an up/down 1-digit BCD counter with 4-bit outputs (DCBA) and 1 overflow output (cout), 1 borrow (bout) and 6 inputs (up, down, set9, set0,clock). Write a simulator to show that the counter function correctly.

2. Create a single pulser with one input, clock, and one output. Write a simulator to show that the single pulser work correctly.

3. Use 4 counters from exercise 1 to create 4 digits BCD counter. Connect all display to 4 digits seven-segment displays. (Use the display components from Laboratory II.) Use BTNU for set9 (set the number to 9999). Use BTNC for reset (set the number to 0000). Use SW0 for countdown by 1. Use SW1 for count up by 1. Use SW2 for countdown by 10. Use SW3 for count up by 10. Use SW4 for countdown by 100. Use SW5 for count up by 100. Use SW6 for countdown by 1000. Use SW7 for count up by 1000. If the number is at 0000, a countdown would not decrease the number. If the number is at 9999, a count up would not increase the number. Do not worry about the bounce at the moment. We will fix it in the next exercise.

4. Correct the bounce in exercise 2 by implementing a debounce component for each input.

5. Please answer the following questions and submit (in PDF format) to CourseVille on Friday before 23:59 (midnight).

    a. From the circuit diagram, the BTNx is active High or active Low? Please provide your analysis.

    b. What is a bounce? How do you programmatically debounce the input? Please provide your analysis.

    c. Please show your method for implementing a single pulser. (e.g. draw a state diagram, or verilogHDL code)