EE 331 Digital Systems with HDL
Oregon Tech, Wilsonville, Winter 2014

Lab Assignment 3, due Week 5, February 4

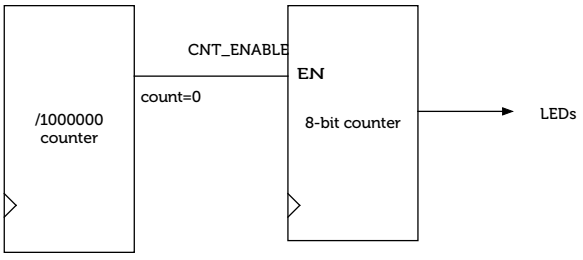
In this lab you will design an 8-bit counter (with value displayed on the LEDs) which increments one count with every press of a push button. To make this work correctly the pushbutton input is first synchronized to the clock, then to de-bounce the pushbutton the input is passed through a low-pass filter, and finally sent through a one-shot so a single depression will produce a single clock-cycle wide pulse that will enable the counter to increment.

I'll show how to set up the clock signal in class.

The design needs two counters. An 8-bit binary counter with an enable input drives the row of LEDs. A second counter counts continuously (with the clock) and divides by 1,000,000 to produce one pulse every 10 milliseconds. The easiest way to make this counter is to have a signal of type integer with a range 0 to 999,999. Have the counter reset to 0 when the value is 999999. The output is generated when the counter value is 0.

You have a decision to make — you can put the counters and the other components like the synchronizer into separate modules, giving a hierarchical design that will look cleaner, or put everything in the top-level module, which will save you time. Particularly in the latter case you need to comment the VHDL well, explaining what each process (register) is for.

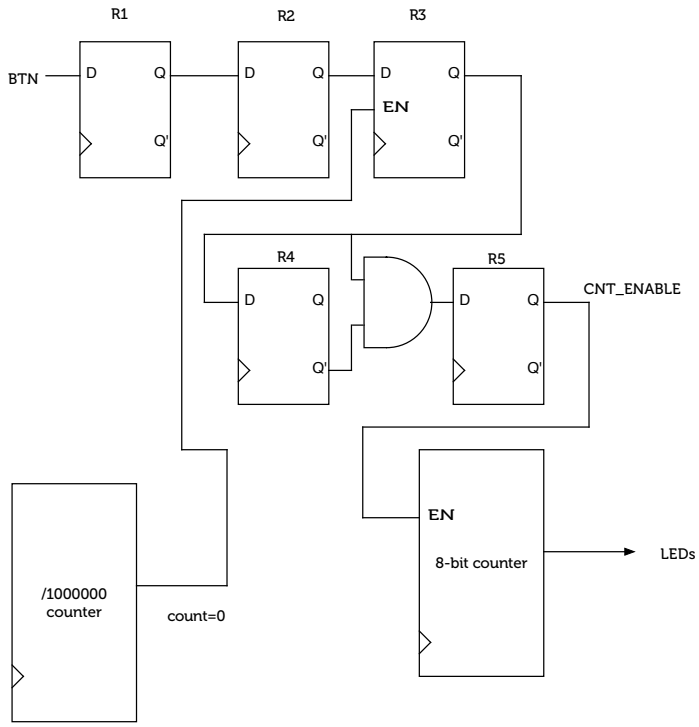
You can start out with the divide by 1,000,000 counter enabling the binary counter, the counter will count 100 counts per second. Note that you never want to have the count=0 signal clock the counter (which would be a gated clock) but instead always use signals like this to enable counters.



Then the next step is to add the button circuitry so the button press enables the counter. The following schematic does the synchronization using R1 and R2, low pass filters with R3 (and the /1000000 counter), and produces a single output pulse per press using R4, R5 and the AND gate which implement a one-shot.

Add the circuit and verify operation by applying (in simulation) a 10 Hz square wave signal to the button input and a 100MHz clock signal. It should count 10 times in 1 second of simulation time. Make a screenshot of the waveform display showing the signal propagating through R3, R4, and R5 when EN on R3 goes high.

Synthesize and observe correct operation.



To Turn In:

1. Report discussing what you did.
2. VHDL source (with comments) for all modules incorporated in the document.
3. Screenshot of simulation for R3 through R5 when EN on R3 goes high.
4. Design Summary showing successful synthesis.