Design Name : Two Copies of an Up-Down Counter with Independent De-bounced Inputs/Outputs

Objective :

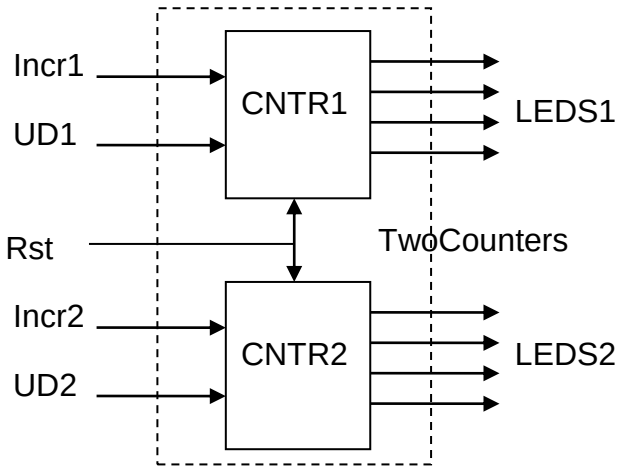
Learn how to create multiple instances of a pre-described design.

Assignment :

1. Describe a 4-bit binary counter with `CLK`, `Incr`, `UD` and synchronous-`Rst` inputs. Name the entity as `UpDnCtr`. This will be the design of a 4-bit up/down fully-synchronous counter with synchronous reset.
2. If you feel that it would be helpful, make your `vhd` file top-level if it is not so already and test it by connecting inputs and outputs to buttons and LEDs respectively as you have already done in the previous experiments.
3. Add a new (empty) `vhd` file with the following interfaces and make it top-level. Name the file and entity as `TwoCounters`. Interface signals of the entity would be

   `CLK` (50 MHz)
   `Rst` (common)
   `UD1, Incr1,`
   `LEDS1` (4 bits)
   `UD2, Incr2`
   `LEDS2` (4 bits)

4. Insert the component description of the counter `UpDnCtr` you have designed in 1 into the declaration part of the architecture section of `TwoCounters`.
5. Insert two instantiations of `UpDnCtr` with names `CNTR1` and `CNTR2` in the `architecture` section after `begin`. Complete `port maps`. Note that `RST` input is common and used to reset both counters.
6. Complete the constraints file by connecting `Incr` and `RST` inputs to push-buttons, `LEDS` outputs to on board LEDs. Connect `UD` inputs to switches. Use pull-up, pull-down resistors when necessary.
7. Power-up and program your FPGA device through USB cable connection. Check if your design works as desired.



You need to detect the transitions of `Incr` inputs and increment/decrement the corresponding counter on the transitions of these inputs. Otherwise, it will count up/down with 50 MHz clock, which is not what we wanted.

8. Insert key de-bouncers on the Incr inputs and test if they work as expected.

Homework :

Convert your two independent counters design to two digit BCD counter. Use on-board clock source (50 MHz) to feed the counter. Divide the clock signal by 50000000 by using another counter designed in `TwoCounters` architecture. Your 2 digit BCD counter will receive 1 pulse per second as a clock.

(hint: this experiment can be a good exam question)