# Lab #6: Switch de-bouncing and state machines.

In this lab you are required to handle various switches and also complete the previous lab residue.

## Clock

Clock counter is he cone that has three counters. Hours/Minute and Seconds. At each one second, the second counter is incremented by 1. As soon as it reaches 60, it is reset back to 0 and the minute counter is incremented. When the minute counter reaches 60, it is reset back to zero and hours counter is incremented. When hours counter reaches 24, it is reset to 0.

You will need to display the clock counters on the 7-segment display where first two digits will show the hour while the last two digits will show the minutes.

Two sets of digits will be separated by two DP LEDs which shall blink at each second. (Will be ON for half second and OFF for half second).

You will notice that the DP LED is mounted so that the two DP LEDs appear like a colon on the display. **Alternately you will notice that there is a separate colon display LED which is available on pin P124 of the FPGA.**

You will need to divide the 25Hz clock by 12 (or whatever count needed) to get 2Hz signal. At each clock edge of this you will need to toggle the DP LEDs/Colon LED.

## Switch de-bouncing.

For a good description on the spring based switches and problems related to the bouncing of the spring can be found at <http://www.labbookpages.co.uk/electronics/debounce.html>.

We shall use a mechanism of sampling the switch status every 5ms. And when in two subsequent samples the switch value is the same, we take that as a value. In other words when a switch status is changed, we delay declaring the switch to be changed by at least one sample period of 5ms.

Use the 25MHz clock for this purpose which is scaled down to 200Hz (i.e. 5ms) and at edge of the clock, sample the switch, compare with the earlier value of the switches (as stored in the local variable) and only if it is the same, transfer it to another variable which provides the state of the switch. For your circuits use this variable as the de-bounced switch value.

In the lab, you need to detect the following sequence of the switches pressed.

Button 0 followed by Button 2 and Button 3. Whenever that occurs you should light up LED0. If should remain off otherwise.

Now this transition is tricky. You will have to implement de-bouncing of the switches first. Even with de-bounced switches, you are going to see the following possible sequences.

… 🡪 Button 0 Pressed 🡪 All buttons off 🡪 Button 2 Pressed 🡪 Button 2 and 3 Pressed 🡪 ….

Or

… 🡪 Button 0 Pressed 🡪 All buttons off 🡪 Button 3 Pressed 🡪 Button 2 and 3 Pressed 🡪 ….

Or

… 🡪 Button 0 Pressed 🡪 All buttons off 🡪 Button 2 and 3 Pressed 🡪 ….

You will keep the LED0 ON as long as Button 2 and Button 3 are kept pressed.

Make a state transition diagram. Implement the state machine on the FPGA using the 200MHz clock that is used for de-bouncing of the keys. The state transition shall take place whenever a button is pressed or released.