Lab 3: Non-Blocking Switch Management

In this lab, you will use a non-blocking (or, more accurately, short-blocking) switch change detection routine to help you respond to single presses of a switch while a basic program loop continues to operate. For this, you will need to have the *SwCk()* routine with built-in de-bouncing in your *SwLED_Lib* library. Pages 67 and 68 of the current version of the Class Notes provide you guidance (and most of the code) required for developing and using this routine.

1. [1 mark] Enter the code for *SwCk()* into your SwLED_Lib.c library. Make sure you understand what this function is doing, and how to use it!
2. [2 marks] Create a basic program loop that increments the top line of the Seven Segment display, in BCD, once for every 100 times through the loop.
3. [3 marks] Each time the UP switch is pressed, the bottom line of the Seven Segment display will count up by one, in hexadecimal, and each time the DOWN switch is pressed, the bottom line of the Seven Segment display will count down by one, in hexadecimal, rolling over to FFFF past 0000. You probably want these routines to respond when the switches are pressed, rather than when they are released.
4. [1 mark] Pressing the MID switch at any time clears both the top and bottom lines of the display.

Notes:

- Holding down any switch should not have a significant effect on the update time for the top line of the display.
- Your instructor will be impressed if you make it so that holding down the MID switch keeps both lines of the display in RESET (i.e. both stay at 0000 until the MID switch is released). Think back to what you did in ICA-7, and pick the appropriate switch-management technique to make this happen.

Code Submission: [3 marks]

Place your “main.c” file for this project, clearly identified as yours, in the Lab 3 Dropbox so your instructor can grade your descriptive file headers, your code documentation and your use of libraries, and, where appropriate, discuss your coding techniques.