

CS 431 Lab #2

Interrupts and Timers

Spring 2015

1 Overview

In this lab you will be introduced to interrupts and Timers on the dsPIC33F board. You must write a program that does all of the following.

1. Toggle LED4 inside your main program loop on every iteration. (5 points)
2. Toggle LED1 every 2 millisecond using Timer2's interrupt. (10 points)
3. Toggle LED2 every 1 second using Timer1's interrupt. (10 points)
4. In the main loop, display in a static location the time since the last reset on every 2000th iteration. The format should be mm:ss.xxx where mm is minutes, ss is seconds, and xxx is milliseconds. (25 points)
5. Reset the time to 00:00.000 when the joystick trigger is pressed using edge-triggered external interrupt request 1 (not by polling the button). Read the documents about interrupts to see how to properly enable the interrupt. (25 points)
6. Use the 16-bit Timer3 to measure how long it takes to execute each iteration of the main program loop. Display the time in a static location as both the TMR3 count value (cycles) and in milliseconds to 4 decimal places. (25 points)

2 Procedure

1. Before getting started, read sections 3.3, 3.4, and 4.6 in the CS 431 Laboratory Manual.
2. Use your code/project from a previous lab as a starting point for coding this lab and remove all unnecessary functionality. A HelloWorld template along with instructions can be found on the course Labs page.
3. A demonstration version of the Lab 02 program that you now need to write is provided in compiled form. To upload it to your dsPIC33F, download load-lab02.bat and lab02.hex and follow the demo upload instructions posted on the course Labs page.
4. Update lab02.c such that it fulfills the requirements specified in the Overview section.

- (a) Timer2 configuration.
 - i. Use the 1:256 prescaler
 - ii. Set the period value such that an interrupt will occur after 1 milliseconds
- (b) Timer1 configuration.
 - i. Use the asynchronous 32.768 kHz clock source.
 - ii. Disable synchronization
 - iii. Set the prescaler and period value such that an interrupt occurs once per second (1 Hz).
 - iv. Make sure to use the builtin write OSCCONL() macro and terminate it with a semicolon.
- (c) Use the T1Interrupt ISR to count the number of seconds since the last reset.
- (d) Timer3 configuration.
 - i. Use the 1:1 prescaler.
 - ii. Make sure to reset the counter (TMR3) to 0 in a way such that no work is ignored in the displayed time interval.
 - iii. Read TMR3 to get the current count value.

At the start of Lab 3, each lab group will be asked to demonstrate and explain your Lab 2 code to the TA.

3 Questions to Ponder

The following questions are provided for your lab group to think about. No written response is required.

1. LED1 should toggle every 2 milliseconds. What do you observe on the oscilloscope when the probe is attached to the LED1 test port?
2. How often will LED1 toggle if the T2Interrupt ISR does not reset TMR2 to 0?
3. LED4 should toggle once with every pass through your main program loop. With the probe attached to the LED4 test post, how long does one iteration take according to the oscilloscope? How does this compare to the time measured using Timer3?
4. Considering its use in this program, is any form of debouncing necessary on the trigger button?