MW 2411 Lab #2
Interrupts and Timers

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 LED1 every 2 millisecond using Timer2's interrupt.
2. Toggle LED2 every 1 second using Timer1's interrupt.
3. In the main loop, display in a static location the time since the last reset (uptime) on every 2000th iteration. The format should be mm:ss.xxx where mm is minutes, ss is seconds, and xxx is milliseconds.
4. Toggle LED3 inside your main program loop on every 2000th iteration.
5. Use the 16-bit Timer3 to measure how long it takes to execute the 2000 iterations of the main program loop where **nothing happens**. Display the time in a static location as both the TMR3 count value (cycles) and in milliseconds to 4 decimal places.

2 Procedure

1. Before getting started, read sections 3.3 and 3.4 in the Laboratory Manual.
2. You can find an MPLAB X IDE project with template code on the Moodle course page.
3. A demonstration version of the Lab 02 program that you now need to write is provided in compiled form.
4. Update lab02.c such that it fulfills the requirements specified in the Overview section.
   (a) Timer2 configuration
      i. Disable the timer
      ii. Use the 1:256 prescaler
      iii. Select internal instruction cycle clock
      iv. Disable Gated Timer mode
      v. Set the period (PR2) value such that an interrupt will occur after 2 milliseconds
      vi. Reset the counter (TMR2) to 0
      vii. Set the priority level
      viii. Clear the Interrupt Flag
      ix. Enable the interrupt
      x. Start the timer
   (b) Timer1 configuration
      i. Disable the timer

ii. Use the 1:256 prescaler
iii. Use the external 32.768 kHz clock source
iv. Disable Gated Timer mode
v. Disable synchronization
vi. Set the period (PR1) value such that an interrupt occurs once per second (1 Hz)
vii. Reset the counter (TMR1) to 0
viii. Set the priority level
ix. Clear the Interrupt Flag
x. Enable the interrupt
xi. Start the timer

(c) For the uptime: Use the T1Interrupt ISR to count the number of seconds. Use the T2Interrupt ISR to count the number of milliseconds.

(d) Timer3 configuration.
i. Disable the timer
ii. Use the 1:1 prescaler
iii. Select internal instruction cycle clock
iv. Disable Gated Timer mode
v. Set the period (PR3) to the highest value possible
vi. Make sure to reset the counter (TMR3) to 0 where it should start to count
vii. Read TMR3 to get the current count value

(e) If your microcontroller is reseting constantly: Have you commented out an ISR?

Due date of code submission can be found on the Moodle submission page for this lab. Only one member of the group must upload the code (all .c and .h files that your project uses compressed in one zip file). At the start of Lab 3, each lab group will be asked to demonstrate and explain their Lab 2 code to the lab instructor.

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. Assuming the timer 2 would not automatically reset TMR2 to 0 when it hits the timer compare value. How often would LED1 toggle?

3. LED3 should toggle once with every pass through your main program loop. With the probe attached to the LED3 test post, how long does one iteration take according to the oscilloscope? How does this compare to the time measured using Timer3?