Austin Bumbalough
CPE 325-08
9/11/2019

Lab 3 Solution

Timing Calculations

There are 10 clock cycles per for-loop iteration, and the processor frequency is 1 MHz (1 million clock cycles per second). This means that there are $\frac{10^6 \frac{cycles}{second}}{10 \frac{cycles}{iteration}} = 10^5 \frac{iterations}{second} = 10^2 \frac{iterations}{millisecond}$ so the number of iterations required for a delay of s milliseconds is $d(s) = 10^2 s$.

An on/off period of 5 Hz means the led must toggle at 10 Hz. 10 Hz means the delay should be 100 ms. Thus, $d(100) = 10^2(100) = 10^4$ for-loop iterations.

Similarly, a 2 Hz blinking frequency means the LED must toggle at 4 Hz (250 ms), and a 1 Hz blinking frequency means the LED must toggle at 2 Hz (500 ms).

$d(250) = 10^2(250) = 25 * 10^3$ iterations and $d(500) = 10^2(500) = 50 * 10^3$ iterations.

Source Code

```
#include <msp430.h>
#include <stdio.h>

#define SW1 P1IN & BIT0 // SW1 is bit 0 of P1IN
#define SW2 P1IN & BIT1 // SW2 is bit 1 of P1IN
#define LED1 BIT2 // LED 1 is bit 2 of P2OUT
#define LED2 BIT1 // LED 2 is bit 1 of P2OUT
#define EVER ;;

/*
-------------------------------------------------------------
*
File: Lab_03/main.c
*
Description: Toggle LEDs based on state of two
             switches
*
Input: MSP430 Experimenter Board SW1 and SW2
*
Output: MSP430 Experimenter Board LED1 and LED2
*
Author: Austin Bumbalough
*
Lab Section: 8
*
Date: 9/10/2019
*
-------------------------------------------------------------
*/

int main(void)
{
```

Austin Bumbalough
CPE 325-08
9/11/2019

```
    WDTCTL = WDTPW | WDTHOLD;    // stop watchdog timer

    // Initialize Pins
    P2DIR |= BIT2 + BIT1;

    // Turn LEDs on
    P2OUT |= LED1 + LED2;

    // Infinite loop
    for(EVER) {

        unsigned long int i;
        switch (P1IN & (BIT1 + BIT0)) {
            case (BIT1 + BIT0): // No switches pressed, bit 1 = bit 0 = 1
                P2OUT |= (LED1 + LED2); // LED1 and LED2 are on when no
switch is pressed
                break;
            case BIT1: // SW1 pressed, bit 1 = 1, bit 0 = 0
                for(i=0;i<10000;i++); // 100ms delay --> 5 Hz
                P2OUT ^= LED1;
                break;
            case BIT0: // SW2 pressed, bit 1 = 0, bit 0 = 1
                for(i=0;i<25000;i++); // 250ms delay --> 2 Hz
                P2OUT ^= LED2;
                break;
            case 0: // SW1 and SW2 pressed, bit 1 = bit 0 = 0
                for(i=0;i<50000;i++); // 500ms delay --> 1 Hz
                P2OUT &= 0;
                for(i=0;i<50000;i++);
                P2OUT |= (LED1 + LED2);
                break;
        }

    }

    return 0;
}
```