A1: Programming in Assembly Language Report

Problem Description

For this assignment we were tasked with programming the MSP430 microcontroller in assembly language. The objective we aimed to meet was a program which would allow output to be produced via push button presses in the form of LED signals. In order to complete this assignment we first had to familiarize ourselves with Code Composer Studio, which we used to interface with the microcontroller. We found out that the buttons are attached to P4.1 and P2.3, and we chose LED1 to activate, which was attached to P1.0.

Pseudocode

```
init:
        Clear port 1 output for LED 1
        Set the bits for p1.0
        Set positions in port 4 and port 2 to be inputs
        Enable pull up/down for port 4 and port 2 ren
        Select pull up for p4out and p2out
        GPIO power on
Main_loop:
check_s:
        Check switch bit value for port 4 and port 2 buttons
        Jump to check_s until either button pressed pressed
Led1_toggle:
        Toggle the bit for p1out
        Jump back to main loop
End pseudocode
```

Assembly Code

```
Init:
                CLR.b  P1OUT                    ;Clear Port 1 output
                CLR.b  P6OUT                    ;Clear Port 6 output

                bis.b    #01,   P1DIR           ;P1.0 output (LED1)
```

```
                mov.b  #01000000b, P6DIR          ;P6.6 output (LED2)

                mov.b  #00000000b, P4DIR          ;set positions in port 4 to be an input
                mov.b  #00000000b, P2DIR          ;set positions in port 2 to be an input

                bis.b  #BIT1, &P4REN              ;enable pull up/down for port 4.1 (switch1)
                bis.b  #BIT3, &P2REN              ;enable pull up/down for port 2.3 (switch2)

                bis.b  #BIT1, &P4OUT              ;select pull up
                bis.b  #BIT3, &P2OUT              ;select pull up

                bic.w  #0001h, &PM5CTL0           ;GPIO power on

main_loop:

check_S1:
                bit.b  #BIT1, &P4IN          ;check the switch bit value for port 4.1(pull up)
                Jz     led1_toggle
                bit.b  #BIT3, &P2IN          ;check the switch bit value for port 2.3(pull up)
                jnz    check_S1

led1_toggle:

                xor.b  #00000001b, P1OUT          ;togle bit P1OUT
                xor.b  #01000000b, P6OUT          ;togle bit P1OUT

                jmp            main_loop          ;repeat to main forever

                nop
```

Basic idea:
Check button 1
If pressed light up led and end code
Else, continue and check button 2
If button 2 not pressed, jump back to check button 1
Else if button 2 was pressed, continue and light up led then end code

Functionality Video

https://youtube.com/shorts/B_az6cR_7Q0?si=hwynvNZD0SEJck-b