

Lab 1 Report

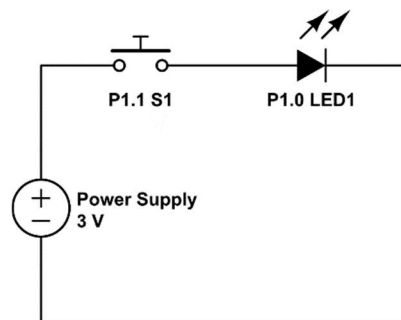
Introduction

We are tasked with understanding and editing the .c and .asm files to perform various functions similar to its original purpose. We will be using one button and one LED for input and output respectively. The user will press the button and the LED will react in different ways depending on the exercise.

Microcontroller Concept

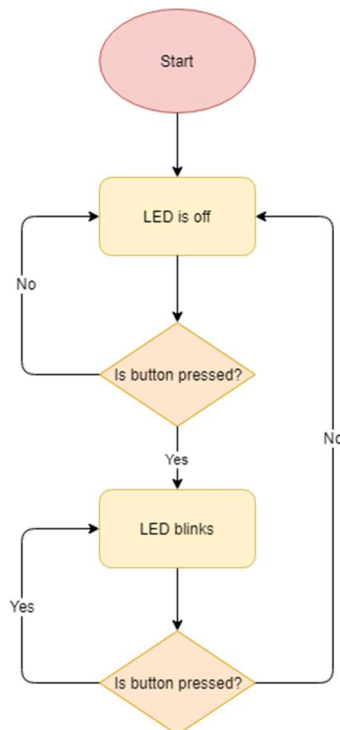
I used an LED and switch. Since I needed a simple input and output, a single button for input and LED for output was all that was needed.

Hardware Design

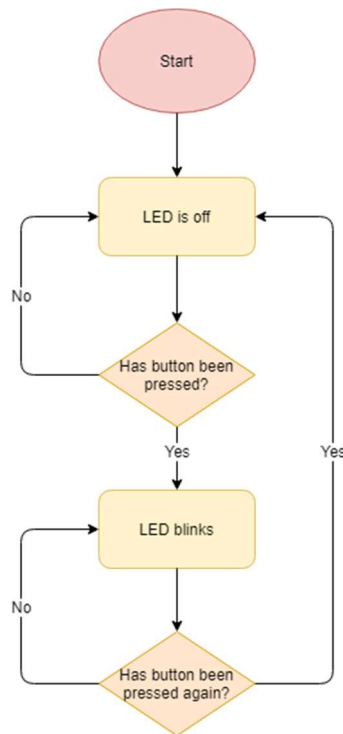


Software Design

Exercise 4



Exercise 5



Conclusion

My lab's outcome was as expected. I didn't run into any problems, but my exercise 5 was a bit finicky when changing the toggle between blinking and off. I could make it better, but it would only marginally improve its reliability.

Questions

- 1) There is a difference in delay between the asm and c code (asm is faster) because assembly is much lower level than c, hence it works quicker and more efficiently.
- 2) For the msp430 $1 \text{ cycle} = 10 * 10^{-6} \text{ s} = 10 \mu\text{s}$. So, for a 0.25s delay we need $25000 \text{ cycles} = 0.25 \text{ s}$.
- 3) The delay matches my result from the formula. I notice a longer time before you can activate the LED again after you change its state. This is to be expected because when pressing the button during the board's "delay" phase, its programming is inactive.
- 6) Step return returns us from a method that has been stepped into.

Debug

- lab1-p5 [Code Composer Studio - Device Debugging]
 - TI MSP430 USB1/MSP430 (Suspended)
 - delay() at main.c:10 0x010078
 - main() at main.c:27 0x010024
 - _c_int00_noinit_noargs_mpu() at boot.c:125 0x004412 (the entry point was reached)

main.c main.c main.c main.c main.asm main.asm

```
1 #include <msp430.h>
2
3 /*
4  * Swtich S1 - P1.1
5  * LED1 - P1.0
6  */
7
8 void delay(void) {
9     volatile unsigned loops = 25000; // Start the delay counter at 25,000
10    while (--loops > 0); // Count down until the delay counter reaches 0
11}
12
13 void main(void)
14 {
15     unsigned
16     WDTCTL =
17
18     P1REN |=
19     P1OUT |=
20     P1DIR |=
21     PM5CTL0 &
```

Expression	Type	Value
loops	unsigned int	24999

Name : loops
Default: 24999
Hex: 0x61A7
Decimal: 24999
Octal: 060647
Binary: 0110000110100111b

Console

lab1-p5