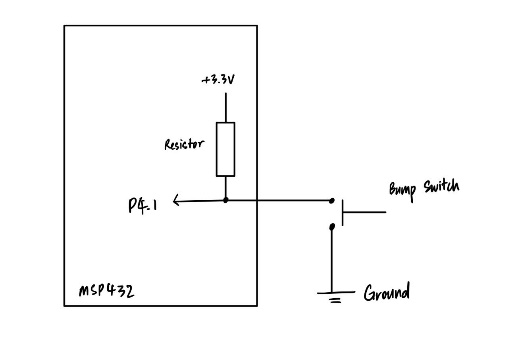
CE2107 Lab3 Assignment Sheet (to be submitted to NTULearn before next lab)

Name: Bryan Lu We Zhern Lab Group: SE4 Date: 14/10/2022

1. Section 6. Other than the procedures outlined in the introduction of Exception Handling, what other registers need to be noted when using the Exception Handling System in ARM Cortex M4F processor? Think global…

PRIMASK, to disable or enable all interrupts indiscriminately.

BASEPRI, to disable interrupts that are of lower priority than the value set.

1. Section 6.2. The bump switch used in the lab is shown below. Pin 1 and 3 of the bump switch are connected to the MSP432. Draw the internal circuit of the bump switch and describe how the MSP432 GPIO can be used to detect that the switch is closed?

When the switch is not pressed, the pin will be pulled high, and therefore reads a 1; else when the switch is pressed, the pin will be pulled low, and therefore the pin reads a 0.

1. Section 6.3. Write down the GPIO configuration used for pins connected to the Bump switches.

P4->SEL0 &= ~0b11101101;

P4->SEL1 &= ~0b11101101;

P4->DIR &= ~0b11101101;

P4->REN |= 0b11101101;

P4->OUT |= 0b11101101;

1. Section 6.3. What is the frequency of the clock source of systick timer? Explain how systick timer is configured to interrupt the system at 1000Hz frequency. Illustrate with detail calculations and APIs used.

Systick timer is of frequency 48 MHz.

Systick timer will have an interrupt when it counts down to zero, and then reload a pre-determined value, so we need to reload in a suitable value so that it counts down to zero 1000 times a second. If we take 48MHz / 1000 Hz = 4800 as the reload value, it will count from 48000 to 0 1000 times every second, which gives us a frequency of 1000 Hz.

The systick reload value is configured by calling the SysTick\_Init function and passing in 48000 as its reload (period) value.

1. Section 6.3. What is the advantage the method of reading Reflectance sensor (in Lab3 section 6.3) has compared to the method used in Lab2?

Using loops and delays to read the sensor readings are resource hogs, since the processor would not do anything during the delay. By using interrupts from the timers to trigger the sensor reading, the processor is free to do anything other tasks.

1. Section 6.4. Reference to PWM\_Init34() in PWM.c, what is the timer base clock used to increment the counters in Timer\_A0? Show the details of how this base clock of Timer\_A0 is derived, starting from processor clock. Note that SMCLK=12Mhz.

bit mode

9-8 10 TASSEL, SMCLK=12MHz

7-6 11 ID, divide by 8

5-4 11 MC, up-down mode

2 0 TACLR, no clear

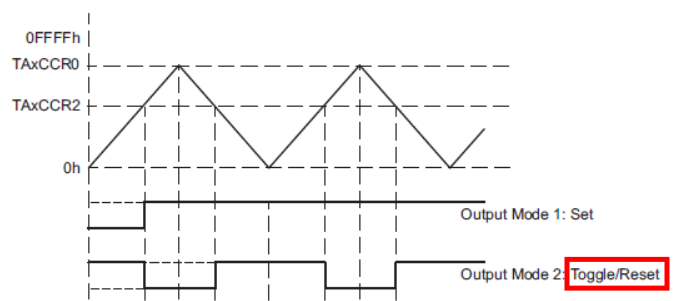
1. 0 TAIE, no interrupt

0 TAIFG

SMCLK is derived from the 48 MHz clock using a clock divider which gives us 12 MHz

Input = 12 MHz, Clock divider = 8, so base clock of Timer\_A0 = 12 MHz / 8 = 1.5 MHz

1. Section 6.5. What is the PWM frequency generated to the motor? Illustrate with detail working.



From PWM.c and Motor.c, TAxCCR0 is loaded with 7500.

The period of the PWM wave is the time from a rising edge to the next rising edge, which is the time taken to count 2 times TAxCCR0 = 15000.

From the previous question, Timer\_A0 frequency is 1.5 MHz.

In one second, the timer can count up and down 1.5 MHz / 15000 = 100 Hz, meaning 100 rising edges is 1 second. Therefore, the PWM frequency is 100 Hz.

1. Section 6.5. Is interrupt mechanism used in the PWM generation via Timers?

No, as the interrupt enable bit (TAIE) in TAxCTL is not set, therefore, no interrupts would be created.

1. Section 6.5. What is the IRQ number corresponding to the interrupt used by Timer\_A1 in Lab3\_TimerCompare\_Motor project use? What is the corresponding Exception number?

IRQ number 10

Exception Number is IRQ number + 16 = 26