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

Name: Ang Kai Jun Lab Group: Date:

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 globally…
2. 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?  
   
3. Section 6.3. Write down the GPIO configuration used for pins connected to pin 3 of the Bump switch.

P4 -> SEL0 &= ~0XED;

P4 -> SEL1 &= ~0XED;

P4 -> DIR &= ~0XED;

P4 -> OUT |= 0XED;

P4 -> REN |= 0XED;

Pin3 is set as input pin, with pullup register enabled.

uint8\_t Bump\_Read(void){

// write this as part of Lab 3

uint8\_t result3, result2, result1;

uint8\_t result = 0X00;

result1 = (P4->IN)&0X01; //bump 0

result2 = (P4->IN)&0X0C; //bump 1 and 2

result2 = result2 >> 1; //remove bit 1

result3 = (P4->IN)&0XE0; //bump 5, 4, 3

result3 = result3 >> 2; //remove bit 4

result = result1 + result2 + result3; //combine all the bumps together

// 0000 0001 + 0000 0110 + 0011 1000 = 0011 1111

return (result);

}

Pin 3 corresponds to bump 2, logical shift to the right is done for pin3 to remove bit 1. The values are then added together to give the total result for Bump\_Read.

1. Section 6.3. Illustrate with detail working and APIs used how systick timer is configure to interrupt the system at 1000Hz frequency.

SysTick\_Init(48000,2);

void SysTick\_Init(uint32\_t period, uint32\_t priority){

SysTick->CTRL = 0; // 1) disable SysTick during setup

SysTick->LOAD = period - 1; // 2) reload value sets period

SysTick->VAL = 0; // 3) any write to current clears it

SCB->SHP[11] = priority<<5; // set priority into top 3 bits of 8-bit register

SysTick->CTRL = 0x00000007; // 4) enable SysTick with core clock and interrupts

}

while(1){

WaitForInterrupt();

if(MainCount%1000 == 0)P2->OUT ^= 0x01; // foreground thread

MainCount++;

}

Clk is set at 48MHz, period of sysTick is 48000 cycles. 48MHz/48000 = 1000Hz.

1. Section 6.4. In the Simple motor project, the api used to move the motor forward is Motor\_ForwardSimple(uint16\_t duty, uint32\_t time), where time is number of 10ms units, i.e. if time=2, motot will run for 2ms. Show and explain the code in the function that enable this 10ms unit timing.
2. Section 6.5. Reference to PWM\_Init1() in PWM.c, what is the base clock used to increment the counters in Timer\_A0? Show the details of how this base clock of Timer\_A0 is derived.
3. Section 6.5. What is the PWM frequency generated to the motor? illustrate with detail working.
4. Section 6.5. Is interrupt mechanism used in the PWM generation via Timers?
5. 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?