

ECE2049: Homework 3

Due: Tuesday, 15 June 2021 by 2pm EDT

Content: Lectures 4-5

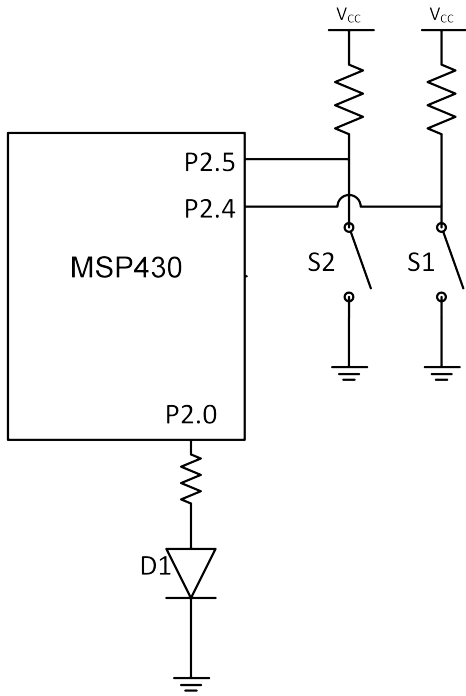
Submission notes:

- For full credit, please show your work and denote your answers with a circle or a box.
 - Always write or draw your diagrams neatly! We cannot be expected to GUESS what you meant to write! Some problems (such as those involving code) must be typed to be graded—the others may be handwritten (neatly!) or typed.
 - Points for each problem are as indicated. Some portions of problems are marked as “BONUS,” which count as extra credit.
1. Using our notes from lecture, answer the following questions about memory on the MSP430F5529. Your answers should be at most one or two sentences.
- a. The MSP430 has both RAM and flash memory. Why does it need both? (In other words, why not just use one or the other?)
 - b. When you download code from CCS to the MSP430, is the code stored in RAM or Flash? How do you know?
 - c. True or False: The MSP430 CPU core has three sets of memory access instructions to access each region of memory (RAM, Flash, and peripherals). **Explain your reasoning.**

(Continued on the next page)

ECE2049: Homework 3

2. (5 pts) In this example, two switches and a blue LED are connected to an MSP430F5529 running the code shown below. Answer the following questions about the configuration.



```

#include <msp430.h>
void configure_io(void)
{
    P2SEL &= ~(BIT5|BIT4|BIT0);
    P2DIR &= ~(BIT5|BIT4);
    P2DIR |= (BIT0);
}

void main(void)
{
    char inVal, s;

    configure_io();

    while(1) {
        inVal = P2IN;
        s = (inVal & 0x30) >> 4;

        if(s > 1) {
            P2OUT &= ~BIT0;
        } else {
            P2OUT |= BIT0;
        }
    }
}

```

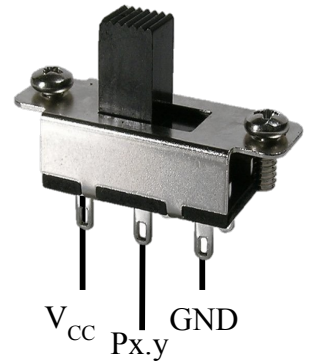
- a. This configuration example code does not use the P2REN register to enable internal pull-up/pull-down resistors. Does it need to be configured in this case? Explain why or why not.
- b. Assuming that the configuration is correct, what is the state of switches S1 and S2 (pressed or unpressed) when inVal has the following values? For each value of inVal, what is the state of the LED (on or off)?
 - i. inVal = 0x7a
 - ii. inVal = 0x1c

One way to format your answer for this part is to complete the following table:

inVal	S2	S1	LED
	(Pressed or unpressed)		ON or OFF
0x7a			
0x1c			

ECE2049: Homework 3

3. (10 pts) Say you want to connect four slide-switches like the one shown at the right to P7.2 and P4.6-4. The switches operate such that when the switch is slid to the right, the pin is connected to GND (0V), and when slid to the left is connects to V_{CC} (3.3V).



- a. Complete the two functions `config_switches()` and `read_switches()` based on the skeleton below. Your code must be typed—you can find this example as a C file on the course website as `switches.c`.

```
#include <msp430.h>

// Function prototypes
void config_switches(void);
char read_switches(void);

// Example main() to demonstrate how the functions are used--no need to modify it.
void main(void)
{
    char val;
    WDTCTL = WDTPW + WDTHOLD;    // Stop watchdog timer
    config_switches();

    while(1)
    {
        val = read_switches();
        // Assume something with val happens here...
    }
}

void config_switches(void)
{
    // Configure switches here!
}

// Return a value between 0-Fh corresponding
// to the value of the switches, with the values
// of each switch in the following bit positions:
// MSB                                     LSB
// Bits 7-4   Bit 3   2       1       0
// 0          P7.2   P4.6   P4.5   P4.4
char read_switches(void)
{
    char ret_val = 0;

    // Read switches and place the output
    // into the appropriate bit here!

    return ret_val;
}
```

- b. Assuming that your program has properly configured the slide switches, what should the function `read_switches()` return given the following register values?

$P7IN = 0x55$, $P7OUT = 0x44$, $P4IN = 0xCF$, $P4OUT = 0xD8$

ECE2049: Homework 3

4. **BONUS (5 pts):** Say you are using a different microprocessor that exposes the memory bus so that you can add new memory-mapped peripherals to it. Assume that you attach your peripheral device and that it has one value that the CPU can read at address 0x1104. You may assume that this processor has a 16-bit data bus, like the MSP430.

How would you write code to read the value at this address? Like all register definitions in C, you can do this with a single `#define` statement. Complete the definition below, which includes an example of how the register should be used.

```
#define MY_REG    (/* Fill in your definition here! */)

void main(void)
{
    int val;
    val = MY_REG; // Read the value of the peripheral at address 0x1104

    // . . .
}
```