Week 5 Lab

CC2511

Firstly, you will read digital input from a floating pin to learn to recognise this common error; and secondly, you will use pulse width modulation to smoothly adjust the LED colours.

Part 1—Floating pins / unconnected inputs

A "floating" pin is an input that is unconnected. This could happen because of a broken circuit, a bad solder joint, etc.

To help you recognise the symptoms in your own designs, you'll deliberately create a floating pin.

1. Select a pin that is not connected to anything other than the header around the outside of the board.

   **K20D50M boards:** use pin **PTC0**.

   K22F boards: use pin **PTB19**.

2. Use the **BitIO** component to set up **digital input** on your chosen pin.

3. Create an endless loop that reads the digital input from your chosen pin and displays the result using the **blue LED**. If the pin reads as high, then turn the blue LED on, and if it reads as low, then turn the LED off.

4. Connect a short wire into the header connected to your pin and touch it with your finger. You should see the LED flicker unpredictably as you touch the wire. You might need to experiment with this; floating inputs are inherently unpredictable.

Avoid the floating input

Some microprocessors (including ours) include optional internal pullup or pulldown resistors. A pullup resistor is a large resistance (e.g. > 10 kΩ) connected to Vcc, so that the pin is *weakly* pulled high. Similarly, a pulldown resistor connects to ground so that the pin is weakly pulled low.

1. Search the Reference Manual to learn how to configure the internal pullup resistor. Hint: the relevant section is the port control module.

2. Modify your code so sending a space character over the serial port will toggle the state of the pullup resistor. This enables you to
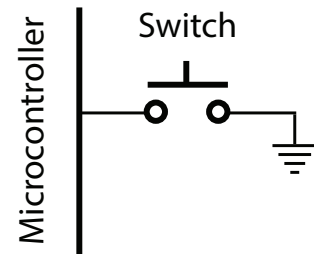
A simple switch circuit with a floating pin. How would you modify the circuit to avoid the problem? Sketch a new version.
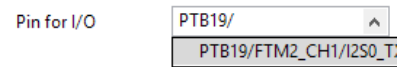
Hint: To quickly pick a pin, type its short name into the drop down list. A trailing slash (as shown) will uniquely identify a single pin.

switch it on or off in software and experiment with its influence on the input pin.

- Dynamic control over the pullup resistor is *not* exposed in Processor Expert. You must manipulate the register(s) directly.

- You can toggle specific bits in a register by using the exclusive or (XOR) operation. For example, to toggle bit 2 and bit 0 in a value X, you could use:

    ```
    X ^= 0b101;
    ```

3. Verify that the presence of the pullup resistor avoids the unpredictable behaviour that you observed earlier.

Conclusion

Hopefully you will recognise the symptoms of a floating pin if you should ever see one. It could save you quite some time in debugging a broken circuit.

Part 2—Pulse Width Modulation

Next, extend your program so that the other two colours (red and green) have variable light intensity. Different intensities can be generated using pulse width modulation (PWM). If the period is fast enough, the flickering will be invisible to the eye, and it will appear as though the LED brightness is changing.

Design a serial port interface to allow interactive adjustment of the RGB colours. For example, pressing "r" might increase the amount of red whereas "R" might decrease the amount of red.

The relevant Processor Expert component is called "PWM". Some hints about using PWM:

- The property "initial polarity" controls whether a ratio of say 20% causes the voltage to be low 20% of the time or high 20% of the time. There are diagrams explaining each case in the help page for the PWM component.

- In addition to the output pin, you may need to select the appropriate FlexTimer channel. Processor Expert calls this the "PWM Device". The channels are indicated in the pin name, e.g.

    ```
    PTA2/UART0_TX/FTM0_CH7/JTAG_TDO/TRACE_SWO/EZP_DO
    ```

    is connected to FlexTimer 0, Channel 7 (as indicated by `FTM0_CH7`).

Assessment

To finish this lab, you must demonstrate to your tutor:

- The source code in Part 1 where you enable the pullup or pull-down resistor.

- Smoothly varying colours on the LED that are controlled over the UART. Each of the two channels must be individually controllable.

- Show your prac tutor your GitHub webpage where your source code is uploaded.