Week 8 Lab

CC2511

Task Description

Your task is to measure light intensity and visually display the result using pulse width modulation (PWM). The CC2511 development board has a light dependent resistor (NSL-19M51) that acts as an analog light sensor. You will read the analog voltage generated by this sensor, and display it to the user.

Recommended steps

1. Locate the light sensor on the CC2511 Assignment 1 schematic, and consider its operational mechanism. Refer also to the sensor's datasheet on LearnJCU. Will a brighter light correspond to a higher or lower voltage?

   Brighter light -> less resistance/higher voltage
   Darker light -> more resistance/lower voltage

2. Identify the microprocessor pin that is connected to the sensor output.

   GPIO26_ACD0

3. Configure the analog-to-digital converter (ADC) to measure the voltage on this pin.

4. Display the ADC result on the serial terminal. Make a note of typical values when the sensor is covered by your finger, and when the sensor is exposed to typical indoor light.

5. Implement a scheme to visually represent the range between darkness (covered by your finger), and indoor light. For example, you might use a single colour only and modulate the intensity with PWM, or you may prefer to switch from one colour to another as the intensity changes. Yet another option would be to change the speed at which the light appears to blink. A more sophisticated approach might be a smooth transition along a colour spectrum, for example from blue to green to yellow to red.

Hints

- Consider a linear equation that relates the ADC measurement to the desired PWM ratio.
- Remember the implications of performing arithmetic on integer numbers, and use floating point numbers where necessary.

- Make sure you appropriately handle the case where your ADC measurement falls outside your calibrated range. PWM ratios cannot be negative!

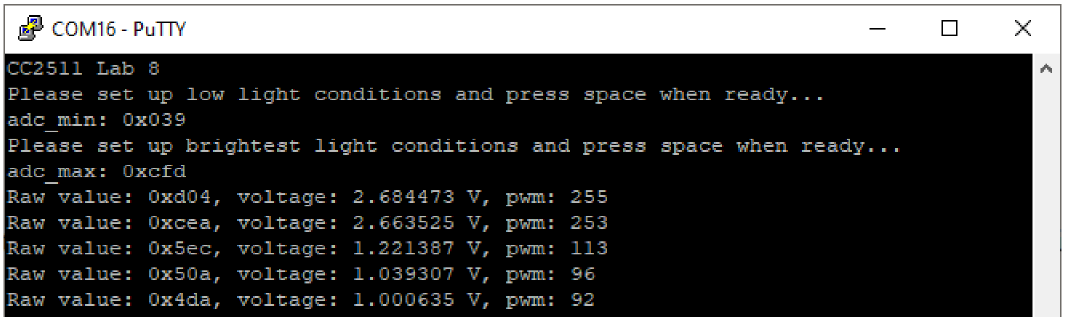
Assessment

To complete this lab task, you must demonstrate to your prac tutor:

- A working board where the light intensity is visually displayed on the LED in an appropriate manner. Note that the transition from maximum brightness to minimum brightness must be clearly visible, i.e. you must scale the ADC result appropriately.
- Your GitHub webpage showing your code uploaded to your repository.

Optional Extension

- Add a user-driven calibration phase.
- Request user input and measure the levels for low and high light intensity, e.g.:



```
CC2511 Lab 8
Please set up low light conditions and press space when ready...
adc_min: 0x039
Please set up brightest light conditions and press space when ready...
adc_max: 0xcfd
Raw value: 0xd04, voltage: 2.684473 V, pwm: 255
Raw value: 0xcea, voltage: 2.663525 V, pwm: 253
Raw value: 0x5ec, voltage: 1.221387 V, pwm: 113
Raw value: 0x50a, voltage: 1.039307 V, pwm: 96
Raw value: 0x4da, voltage: 1.000635 V, pwm: 92
```

2nd Optional Extension

- Use free running mode, and create an event handler to read the ADC inputs (see lecture notes).
- Design an event loop for main().
- Throttle the frequency of reports to the serial output using a timer, e.g.:

```
absolute_time_t next = 0;
...
if (get_absolute_time() > next) {
  printf(...);
  next = make_timeout_time_ms(1000);  // Next serial output in 1000 ms
}
```