PROBLEMS

1. Determine the appropriate bit settings for TCCR2A and TCCR2B to generate a PWM output using the following specific details

– use channel B for the output waveform,

– use phase-correct PWM mode,

– use clear on up-counting, set on down-counting,

– have TOP defined as OCR2A,

– use a pre-scaler division value of 1.

2. Create a function that takes as an input a floating point number between 0 and 1. Based on the floating point value, the function needs to set the appropriate percent value in the OCR2B register. For example, if the input value is 0.6, then the count should be 60% of TOP.

3. Write a program that outputs a 2 kHz, 0 to 3.3 V saw-tooth PWM signal.

Note that DDR*x* still has to be set to an output for the PWM port pin.

Tip: in order to generate a 2 kHz signal, you will have to repeatedly call the function you created in problem 2 with the appropriate analog value between 0 and 1, and with the appropriate amount of delay between samples. The delay() function is too slow to allow for a 2 kHz signal, but you can use either micros() to see how much time has passed from a previous call (like polling an input switch), or delayMicroseconds() to force a specific delay.