Victor Udeh  
Cs350 module 2  
2-2 Milestone One Submission  
Sept 7, 2024.  
  
  
  
  
1. What does pwmled2 set the PWM period to?

* In the provided code, the PWM period is set to 3000 microseconds (or 3 milliseconds). This defines the length of one complete cycle of the PWM signal, including both the HIGH and LOW phases of the signal.
* The period value is defined in the variable pwmPeriod and used in the PWM\_Params structure:

uint16\_t pwmPeriod = 3000; // 3000 microseconds

2. Which PWM\_xxx() function sets the PWM period?

* The PWM period is set using the function PWM\_setPeriod() indirectly through the PWM\_Params structure. In this case, the period is set during the initialization of the PWM parameters with:

params.periodValue = pwmPeriod;

* While the PWM\_setPeriod() function is not explicitly called in the provided code, the period is established when the PWM\_open() function is called using the params structure that holds the period value.

3. Which PWM\_xxx() function sets the PWM duty cycle?

* The function that sets the PWM duty cycle is PWM\_setDuty(). This function takes two parameters: the PWM handle (e.g., pwm1 or pwm2) and the duty cycle value (in microseconds).

PWM\_setDuty(pwm1, dutyPWM1); // Set duty cycle for PWM1

PWM\_setDuty(pwm2, dutyPWM2); // Set duty cycle for PWM2

4. What is the purpose of the while(1) loop in pwmled2?

* The while(1) loop is an infinite loop that keeps the program running continuously. In embedded systems like this, the code must run indefinitely, performing the same operations over and over. In this case, the loop ensures that the LEDs keep alternating between their 90% and 10% duty cycles until the system is powered off or reset.

while (1) {

// Code that toggles the duty cycles runs indefinitely

}

5. What is the purpose of usleep() in the while(1) loop?

* The usleep() function introduces a delay within the loop. It pauses the execution for a specified period, which is given in microseconds. In this code, usleep(1000000) makes the system wait for 1 second before switching the duty cycles. This allows the LEDs to stay in their current state (either 90%/10% or 0%/90%) for 1 second before changing.

usleep(sleepTime); // Pauses for 1 second