Lab 4 – PWM with PWM Generator

CSE 479 – Advanced Embedded Systems
Thomas Strade (tstrade)
10/4/2025

Table of Contents:
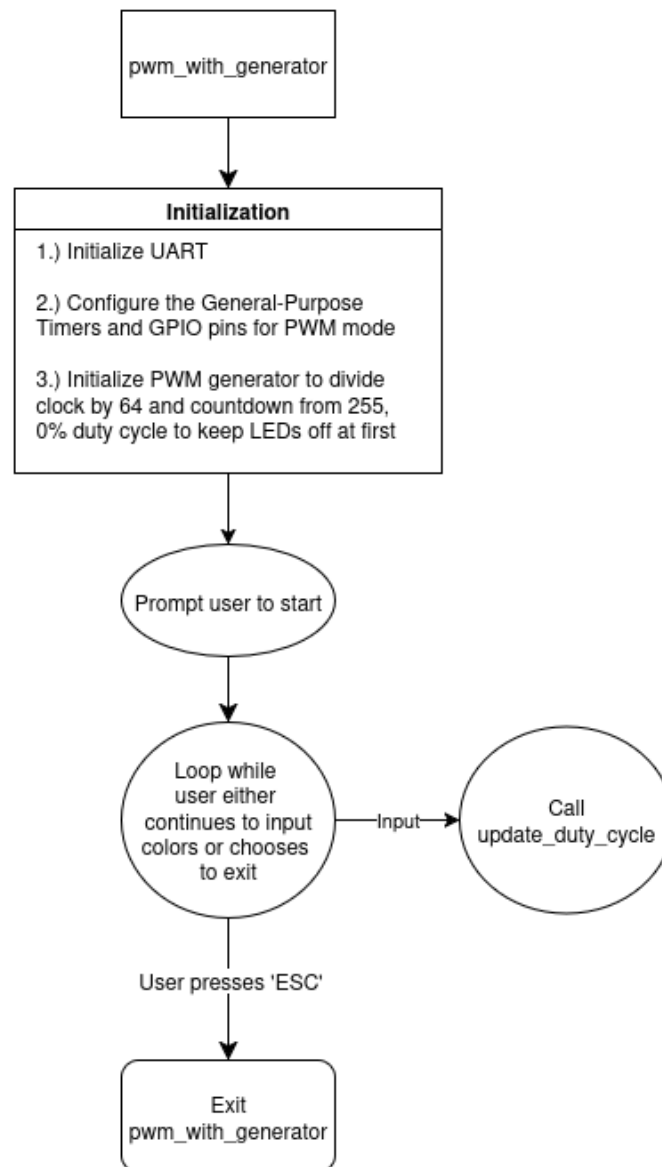
Program Overview:

Description:

The Advanced RGB LED allows users to input the hex code for any color desired. This version of the program operates using a PWM generator, which controls the duty cycle of each RGB value by holding the signal high for a number of ticks ranging from 0 to 255. This ratio of high versus low signals produces a duty cycle that determines the intensity of the corresponding red, green, or blue LED, and can be controlled by changing the generator's compare value.

High-Level Flowchart:

```
              ┌──────────────────────┐
              │  pwm_with_generator   │
              └──────────┬───────────┘
                         │
                         ▼
     ┌──────────────────────────────────────┐
     │            Initialization              │
     ├──────────────────────────────────────┤
     │ 1.) Initialize UART                    │
     │                                        │
     │ 2.) Configure the General-Purpose      │
     │ Timers and GPIO pins for PWM mode      │
     │                                        │
     │ 3.) Initialize PWM generator to divide │
     │ clock by 64 and countdown from 255,    │
     │ 0% duty cycle to keep LEDs off at first│
     └──────────────────┬───────────────────┘
                        │
                        ▼
                ╭───────────────╮
                │ Prompt user to │
                │     start      │
                ╰───────┬────────╯
                        │
                        ▼
            ╭─────────────────╮              ╭───────────────╮
            │   Loop while     │              │               │
            │   user either    │── Input ──▶  │     Call       │
            │ continues to input│              │ update_duty_cycle│
            │ colors or chooses │              │               │
            │     to exit       │              ╰───────────────╯
            ╰────────┬─────────╯
                     │
              User presses 'ESC'
                     │
                     ▼
            ┌──────────────────┐
            │       Exit        │
            │ pwm_with_generator│
            └──────────────────┘
```

Subroutines:

update_duty_cycle:

This subroutine takes the user's hex color code as an input. The red, green, and blue values are extracted and used as the new compare values for their respective PWM generators. This controls the duty cycle for each LED, which allows the LEDs to appear as the color that the user specifies. This works because each time the generators count down to either the compare value or 0, the signal is inverted. The PWM generator is "enabled" each call only to ensure that the LEDs are off until the user begins the program or if the respective RGB were previously set to 0. The flo

```
┌─────────────────────┐
│  update_duty_cycle  │
└─────────────────────┘
          │
          ▼
      ╭─────────╮
     ╱  Take hex ╲
    │ color code  │
     ╲ as input  ╱
      ╰─────────╯
          │
          ▼
     ╭───────────╮
    ╱ Extract each ╲
   │   RGB value    │
   │  and use as    │
   │  new PWM       │
   │ compare value  │
    ╲ for respective╱
     ╲  generator  ╱
      ╰───────────╯
          │
          ▼
   ┌─┬─────────────┬─┐
   │ │ Ensure PWM is│ │
   │ │  enabled    │ │
   └─┴─────────────┴─┘
          │
          ▼
   ╭───────────────╮
   │     Exit       │
   │ update_duty_cycle │
   ╰───────────────╯
```