

Pulse Width Modulation (PWM)

with general purpose timer

Project 4

Prepared for

CSE 410 Advanced Topics in Embedded System Design

By

Steven Collins (sac9)

October 20, 2024

Table of Contents

Section 1: Division of Work.....	2
Section 2: Program Overview.....	3
→ Program Overview.....	3
→ Program Summary.....	3
→ High Level Flowchart.....	4
Section 3: Subroutine Descriptions.....	5
→ Timer0A_Handler.....	5
Section 4: Subroutine Flowcharts.....	6
→ Timer0A_Handler Flowchart.....	6

Section 1: Division of Work

I, Steven Collins, wrote the entirety of the pwm project. Some subroutines may have been used from CSE 379, on which my partner Frank Peretti and I worked together to implement.

Section 2: Program Overview

→ Program Overview

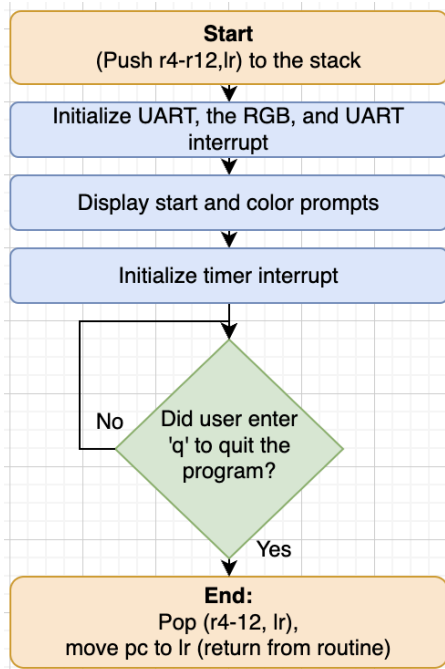
- ◆ Run Program.
- ◆ User enters a numerical code on the keyboard, corresponding to a color, and the RGB LED lights up based on the color code entered.
- ◆ Color Codes:
 - 1 - Orange
 - 2 - Magenta
 - 3 - Light Pink
 - 4 - Purple
 - 5 - Capri Blue
 - 6 - Light Purple
 - 7 - Yellow-Green
 - 8 - Light Sea Green
 - 9 - Light Emerald
 - 0 - Rainbow
- ◆ The color code options are 0-9, where an invalid keystroke will be ignored (or pause the rainbow if code 0 is current). The user may press 'q' at any time to quit, or otherwise continually enter color codes to change the RGB LED.

→ Program Summary

The pwm with general purpose timer is setup to start the RGB LED as orange, and allow the user to change the color afterwards. Code 1-9 are generated by way of timer interrupt, interrupting every 0.01s. Code 0 (rainbow) is generated by enabling the pwm modules. Entering code 0 will activate pwm modules; entering a different code afterwards will disable the pwm module generators and use the general purpose timer interrupts again. Re-entering code 0 will reactive the pwm generator, however the rainbow may not begin in the same spot it left off, or the beginning of the sequence.

→ High Level Flowchart

An overview of the entire program



Section 3: Subroutine Descriptions

Description of each subroutine's functionality, the arguments it passes through, and what the return values in each register are.

→ Timer0A_Handler

◆ Functionality:

- This subroutine is called every .01 seconds. It calls Advanced RGB LED which determines what needs to happen to each RGB pin based on the color code it receives.

◆ Arguments:

- r0 - color code (0-9)

◆ Return registers:

- There are no return registers for the functionality of this subroutine.

Section 4: Subroutine Flowcharts

→ Timer0A_Handler Flowchart

