



## Design Assignment 3 RGB

---

CPE 301 Fall 2016

Luis Ruiz

## PART 0

---

The assignment took me about 2 hours to do, the main reason was because I was having trouble understanding how PWM worked.

### 1. PART A

---

My C program drives a RGB LED to display various colors; it consists of two timers. The counters used to generate pulse-width modulated(PWM) signal were TIMER 0 (TC0) and TIMER 2 (TC2). I am using FAST PWM with no pre-scaling and in non-inverting mode; basically it clears at the TOP and sets at OCnx. The pins that output the signals are PD.6 (OC0A), PD.5(OC0B), and PB.3(OC1A). These values are consistently changing based on the duty cycle that is being interpreted. I have one function that returns a value where OCnx will set, this value is based on the passed in duty cycle, once returning a delay of .5s is set in order to have enough time to see the colors being outputted. The max duty cycle generated is 90%, while the min 10%.

## PART B

---

```
/*
 * DA3.c
 *
 * Created: 2/15/2017 5:08:39 PM
 * Author : Luis
 */

#include <avr/io.h>
#include <util/delay.h>
#include <math.h>

#define F_CPU 16000000L
//UPDATE According to delay
#define DELAY 500

char PWM_Desired(unsigned char);
int main(void)
{
    //TIMER 0 A&B
    //Fast PWM Mode
    //WITH NO PRESCALING
    //CHECK OCF0A and OCF0B
    //TOP OCR0A[7:0] and TOP OCR0B[7:0]
    TCCR0A = _BV(COM0A1) | _BV(COM0B1) | _BV(WGM01) | _BV(WGM00);
    TCCR0B = _BV(CS00);

    //TIMER 2
    //Fast PWM Mode
```

```

//WITH NO PRESCALING
//TOP 0CR2A[7:0]
//TIFR2 TO CHECK OCFA
TCCR2A = _BV(COM2A1)|_BV(WGM21)|_BV(WGM20);
TCCR2B = _BV(CS20);

//PD.6 is OC0A PD.5 is OC0B
DDRD = 0x60;
//PB.3 is OC2A
DDRB = 0x08;

while (1)
{
    //DUTY CYLCE FROM 10% to 90%
    //OCR0A = GREEN
    //OCR0B = BLUE
    //OCR2A = RED
    for(unsigned char i = 10; i <= 90; i += 5)
    {
        unsigned char val = PWM_Desired(i);

        //SET the desired counter based on duty cycle
        //and after have .5s delay
        OCR0A = val;
        _delay_ms(DELAY);
        OCR0B = val;
        _delay_ms(DELAY);
        OCR2A = val;
        _delay_ms(DELAY);
    }

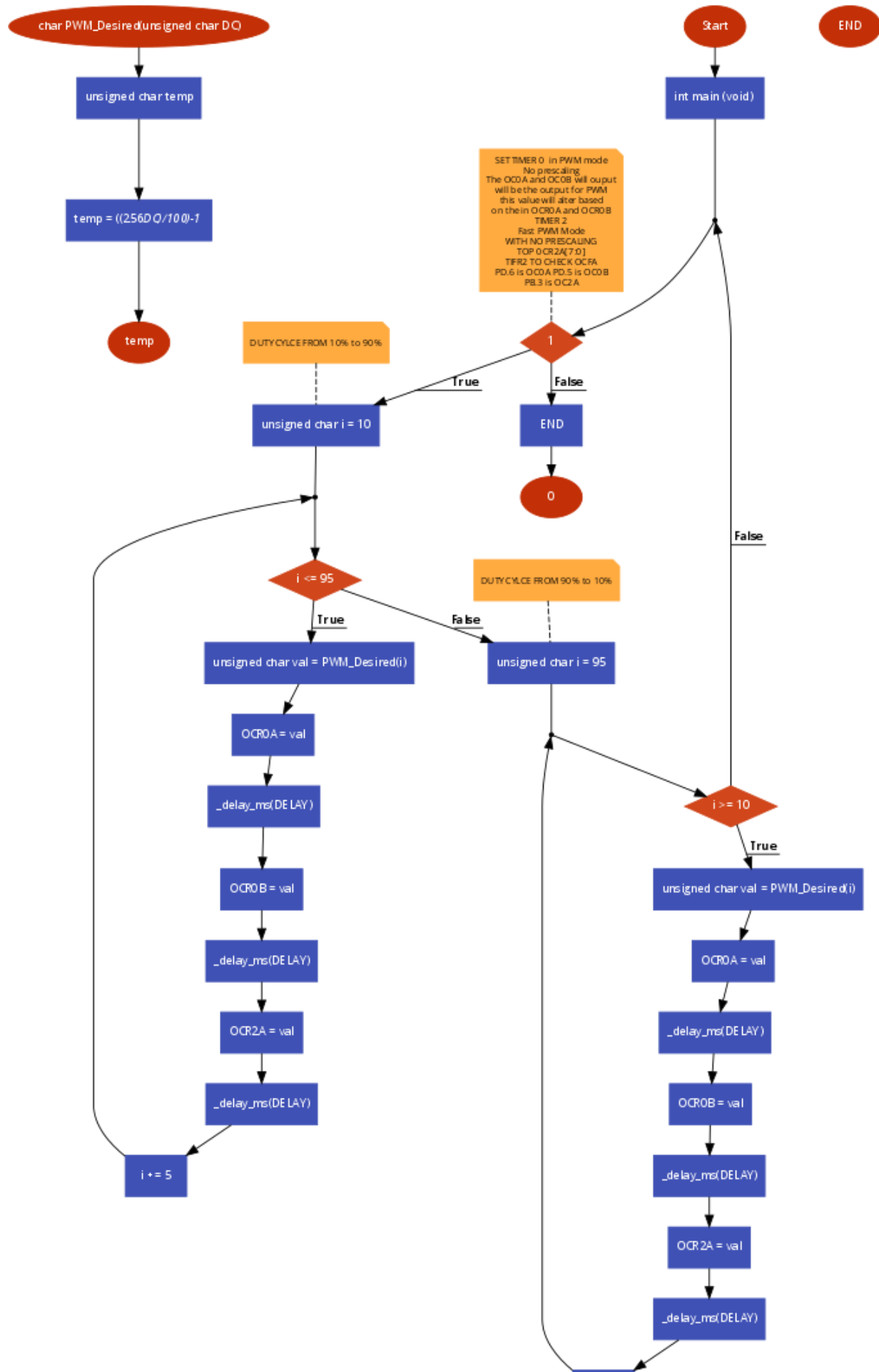
    //DUTY CYLCE FROM 90% to 10%
    for(unsigned char i = 90; i >= 10; i -= 5)
    {
        unsigned char val = PWM_Desired(i);
        //SET the desired counter based on duty cycle
        //and after have .5s delay
        OCR0A = val;
        _delay_ms(DELAY);
        OCR0B = val;
        _delay_ms(DELAY);
        OCR2A = val;
        _delay_ms(DELAY);
    }
}

//Calculate our OCRxx value based on the passed in duty cycle
//OCRxx = ((256*Duty Cycle)/100-1
char PWM_Desired(unsigned char DC){
    unsigned char temp;
    temp = ((256*DC)/100)-1;
    return temp;
}

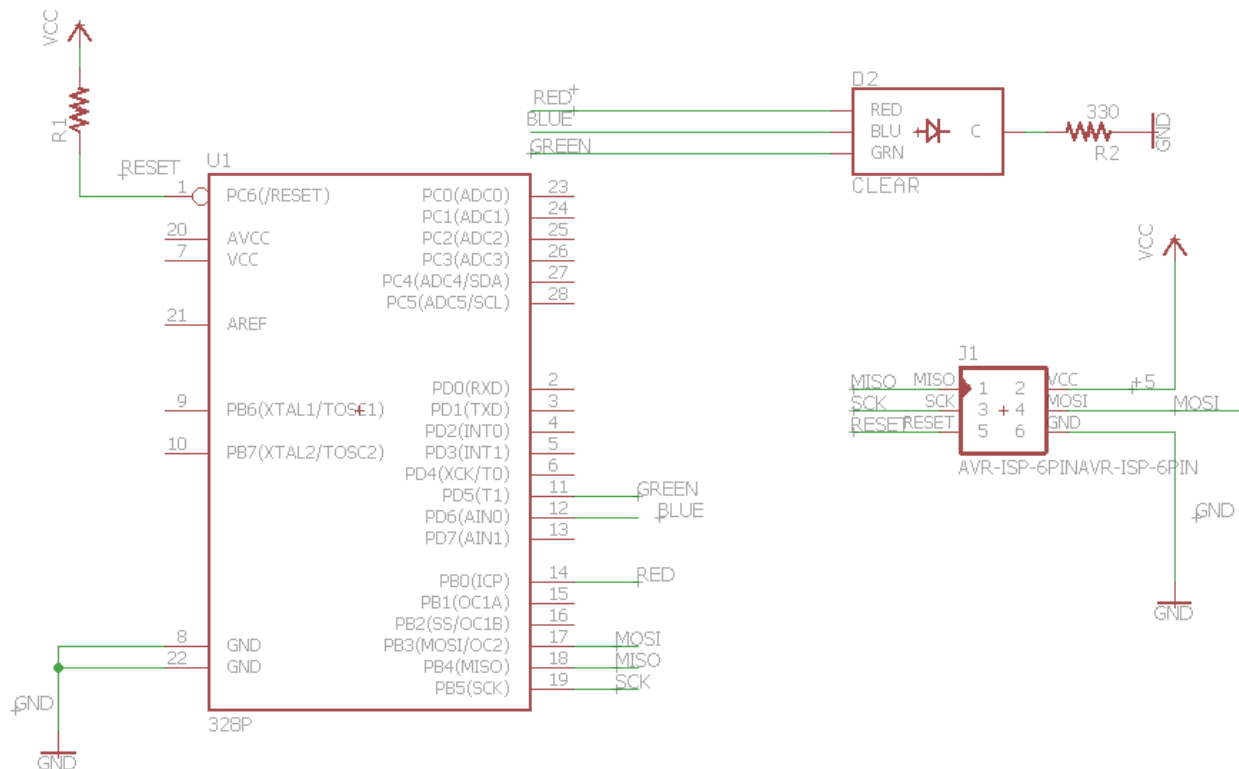
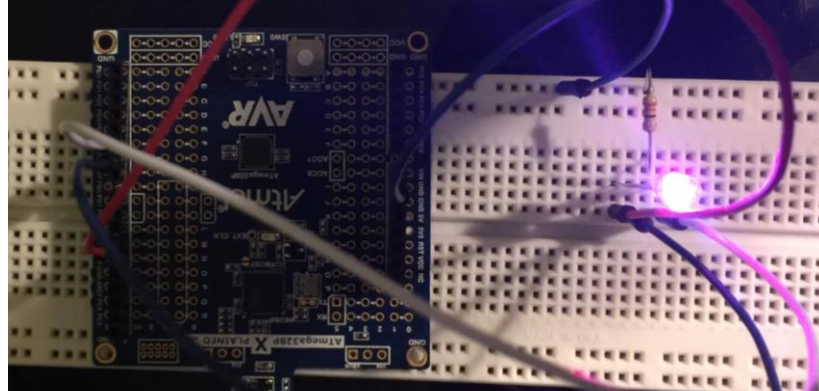
```

## PART C

---



## PART D: Schematic



## PART E

URL Video of Design Assignment 2: <https://youtu.be/y9kXWX6vEJO>

The LED color is shown in two different lighting scenario