Michael Ghisilieri

CPE301 – SPRING 2016

Design Assignment 4

**DO NOT REMOVE THIS PAGE DURING SUBMISSION:**

The student understands that all required components should be submitted in complete for grading of this assignment.

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **SUBMISSION ITEM** | **COMPLETED (Y/N)** | **MARKS**  **(/MAX)** |
| 0. | COMPONENTS LIST AND FLOW CHART |  |  |
| 1. | INITIAL CODE OF TASK 1 |  |  |
| 2. | SCHEMATICS |  |  |
| 3. | SCREENSHOTS OF EACH TASK OUTPUT |  |  |
| 4. | SCREENSHOT OF EACH DEMO |  |  |
| 5. | VIDEO LINKS OF EACH DEMO |  |  |
| 6. | GITHUB LINK OF THE DA |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 0. | COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS |  |  |

**COMPONENTS:**

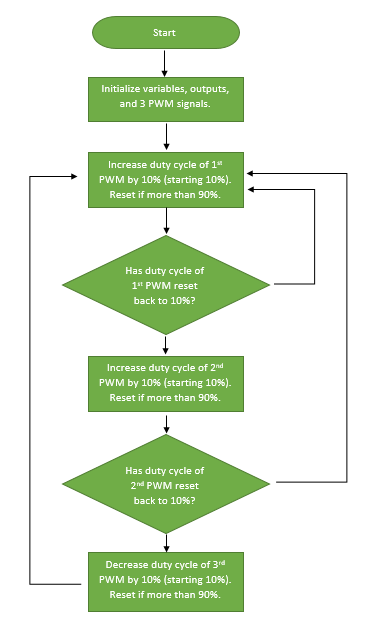
ATmega328P Chip x1

RGB LED x1

360 Ω resistor x2

100 Ω resistor x1

**FLOWCHART:**



|  |  |  |  |
| --- | --- | --- | --- |
| 1. | INITIAL CODE OF TASK 1/A |  |  |

/\*

\* DA4\_Task1.c

\*

\* Created: 3/31/2016 3:23:01 PM

\* Author : Michael

\*/

#define *F\_CPU* 8000000UL // frequency for delay functions

#include <util/delay.h> // library for \_delay\_ms() function

#include <avr/io.h> // library for I/O

void delay (); // function declaration for delay

int main(void)

{

unsigned char i, j, k;

// two variables to update 2nd and 3rd PWMs

unsigned char second, third;

DDRD = 0x40; // OC0A

DDRB = 0x0A; // OC1A and OC2A

// 3 variables being decremented to update 3 timers

// initialize to 230 for 10% value of 255

i = 230;

j = 230;

k = 230;

// both counters for 2nd and 3rd PWM will start at 0

second = 0;

third = 0;

OCR0A = 230; // 230 is 10% value of 255 when decrementing

TCCR0A = 0xC1; // Phase Correct PWM, inverted

TCCR0B = 0x03; // N = 64

OCR1A = 230; // 230 is 10% value of 255 when decrementing

TCCR1A = 0xC1; // Phase Correct PWM, inverted

TCCR1B = 0x03; // N = 64

OCR2A = 230; // 230 is 10% value of 255 when decrementing

TCCR2A = 0xC1; // Phase Correct PWM, inverted

TCCR2B = 0x04; // N = 64

while (1) // loop forever

{

// update 3 PWM values at beginning of each loop

// from delay of 40ms

OCR0A = i;

OCR1A = j;

OCR2A = k;

second++; // increment counter for 2nd PWM every time 1st is updated

i = i-25; // increase duty cycle of PWM1 by 10%

if(i < 30) // if duty cycle is more than 90% (inverted)

i = 230; // reset duty cycle back to 10%

if(second >= 9) { // if PWM1 reset duty cycle

second = 0; // clear counter for PWM2

third++; // increment counter for PWM3

j = j-25; // increase duty cycle of PWM2 by 10%

if(j < 30) // if duty cycle is more than 90% (inverted)

j = 230; // reset duty cycle back to 10%

}

if(third >= 9) { // if PWM2 reset duty cycle

third = 0; // clear counter for PWM3

k = k-25; // increase duty cycle of PWM3 by 10%

if(k < 30) // if duty cycle is more than 90% (inverted)

k = 230; // reset duty cycle back to 10%

}

delay(); // delay for 40ms

}

return 0;

}

void delay () {

int i; // declare i for counter

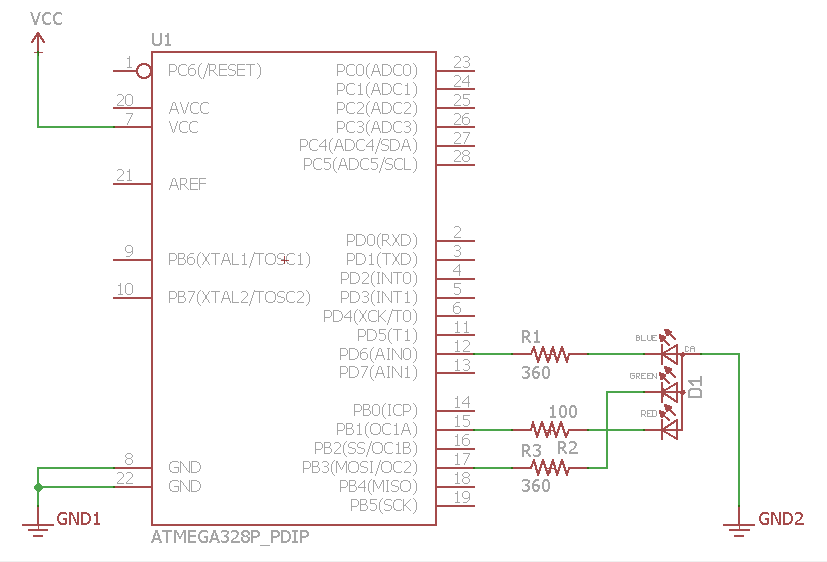
for(i = 0; i < 40; i++) { // loop 40 times

*\_delay\_ms*(1); // delay 1 ms

} // total delay = 40\*1ms = 40ms

}

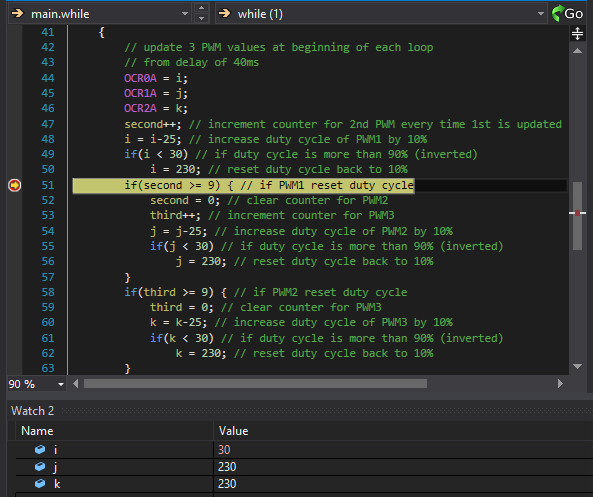
|  |  |  |  |
| --- | --- | --- | --- |
| 2. | SCHEMATICS |  |  |

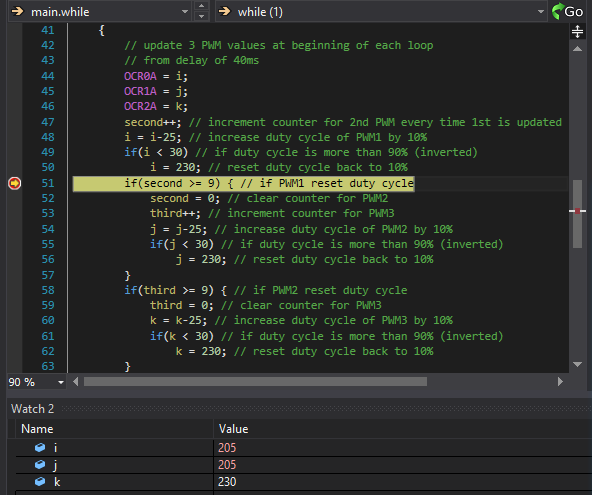


|  |  |  |  |
| --- | --- | --- | --- |
| 3. | SCREENSHOTS OF EACH TASK OUTPUT |  |  |

TASK 1: The first image here shows the value **i** being output to OCR0A as 30 which will be 90% duty cycle using inverted mode. The second image shows the duty cycle of the second PWM being output to OCR1A = **j** increasing after the first PWM’s duty cycle has cycled completely through its own duty cycles.

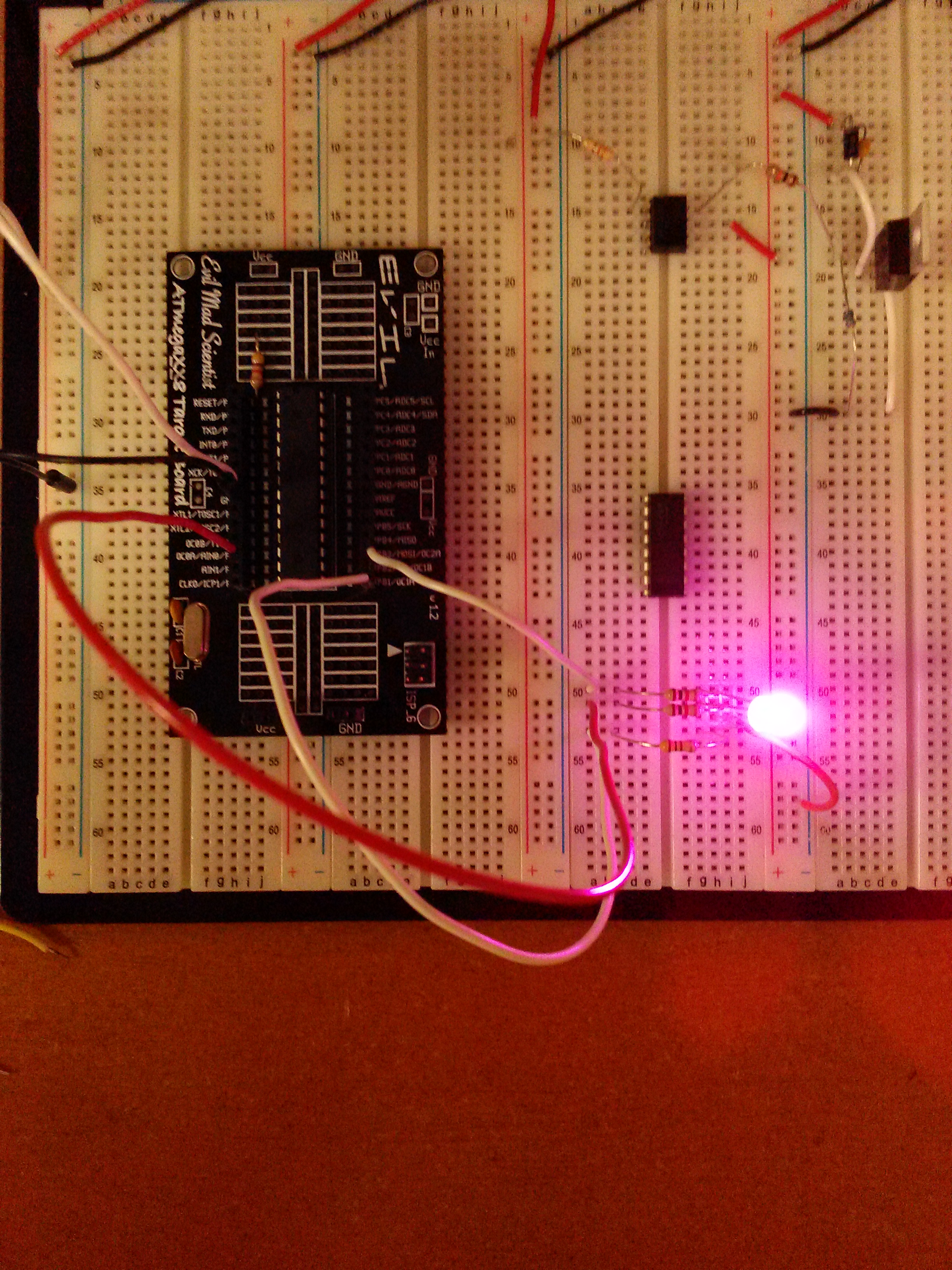
The same exact scenario will take place with the third PWM’s duty cycle when the second PWM’s duty cycle has been completely cycled through and OCR2A will be changed through **k**.





|  |  |  |  |
| --- | --- | --- | --- |
| 4. | SCREENSHOT OF EACH DEMO |  |  |

BREADBOARD: Actual circuit is the ATmega328P connected to RGB with 3 resistors. Image was taken during mid sequence of the RGB.



|  |  |  |  |
| --- | --- | --- | --- |
| 5. | VIDEO LINKS OF EACH DEMO |  |  |
| <https://youtu.be/Ua0ivqkGdVw> | | | |
| 6. | GITHUB LINK OF THE DA |  |  |
| https://github.com/michael-ghisilieri/CpE301\_DAs.git | | | |

**Student Academic Misconduct Policy**

<http://studentconduct.unlv.edu/misconduct/policy.html>

“This assignment submission is my own, original work”.

MICHAEL GHISILIERI