CPE301 – SPRING 2019

Design Assignment 4

Student Name: Alira Coffman
Student #: 5003236350
Student Email:coffma2@unlv.nevada.edu
Primary Github address:
https://github.com/Alira-Coffman/submission-repo.git
Directory:

Submit the following for all Labs:

1. In the document, for each task submit the modified or included code (only) with highlights and justifications of the modifications. Also, include the comments.
2. Use the previously create a Github repository with a random name (no CPE/301, Lastname, Firstname). Place all labs under the root folder ESD301/DA, sub-folder named LABXX, with one document and one video link file for each lab, place modified asm/c files named as LabXX-TYY.asm/c.
3. If multiple asm/c files or other libraries are used, create a folder LabXX-TYY and place these files inside the folder.
4. The folder should have a) Word document (see template), b) source code file(s) and other include files, c) text file with youtube video links (see template).

1. COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS

List of Components used :

- Atmega328p
- Motor
- MD08A Driver
- Battery-pack with two double AA
- Breadboard

2. INITIAL/MODIFIED/DEVELOPED CODE OF TASK 1/A

```
/*
 * DA4A.c
 *
 * Created: 11/27/2019 12:58:34 PM
 * Author : Alira
 */

#include <avr/io.h>
#include <stdlib.h>
#include <stdio.h>
#include <avr/interrupt.h>
#define F_CPU 16000000UL
#include <util/delay.h>
#define BAUDRATE 9600
#define BAUD_PRESCALLER (((F_CPU / (BAUDRATE * 16UL))) - 1)
/*GLOBAL VARIABLES */
volatile uint8_t Direction = 0;
volatile int readValue;
volatile float adc_in = 0;
char buff[5];

ISR(ADC_vect);
ISR(PCINT1_vect);
/*ADC FUNCTIONS*/
void adc_init(void)
{
    ADMUX = (0<<REFS1)|    // Reference Selection Bits
    (1<<REFS0)|
    (0 << ADLAR);

    ADCSRA = (1<<ADEN)|
    (1<<ADATE)|
```

```
    (1<<ADIE)|
    (1<<ADPS2)|
    (1<<ADPS1)|
    (1<<ADPS0);

    ADCSRB = (1 << ADTS1) | (1 << ADTS0);
}
uint16_t clearADCChannel(uint8_t channel){
    ADMUX = (ADMUX & 0xF0)|(channel & 0x0F); //clear channel
}
void ADCEnableDisable(int on)
{
    if(on)
    ADCSRA |=(1<<ADSC);
    else
    {
        ADCSRA &= ~((1<<ADEN)|(1<<ADIE));
    }

}
//MIAN PORTS
void InitPort(){

    DDRC  |= (0<<PINC0);
    PORTC |= (0<<PINC0);
    DIDR0 |= (1<<ADC0D);
    PORTC |= (1<<PINC1);
    PCICR |= (1<<PCIE1);
    PCMSK1 |= (1<<PCINT9);
    DDRB |= (1<<PINB1)|(1<<PINB2);
    PORTB &= ~(1<<PINB2);

}

/*TIMER FUNCTIONS*/
void TimersInit()
{
    /*initalize timer 0 for */
    // Set Initial Timer value
    TCNT0 = 0;
    // Place TOP timer value to Output compare register
    OCR0A = 60;
    // Set CTC mode
    //  and make toggle PD6/OC0A pin on compare match
    TCCR0A |= (1<<COM0A0)|(1<<WGM01);
```

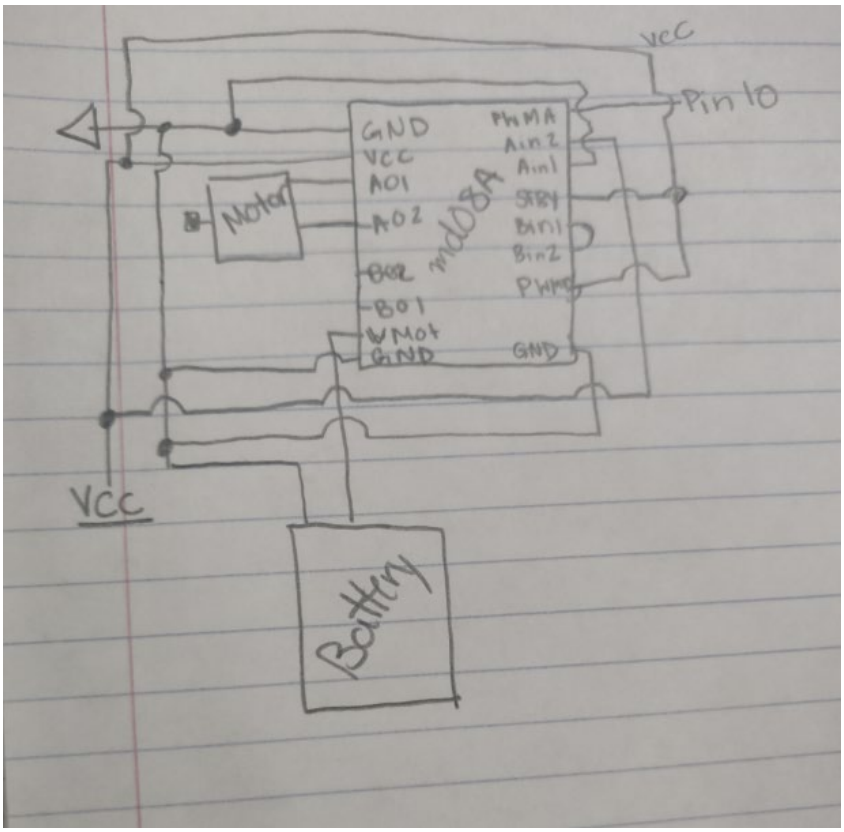
```
    /*TIMER 0 INITALIZED*/

    /*TIMER 1 */
    TCCR1A |=(1<<COM1A1)|(1<<COM1A0)|(1<<COM1B0)|(1<<COM1B1)|(1<<WGM10);
    /*Timer 1*/
}
void startTimer1()
{

        TCCR1B |= (1<<CS12)|(1<<CS10);


}
void startTimer2()
{
TCCR0B |= (1<<CS01);
}
void ToggleTimer1(void){
    TCCR1B ^= (1<<CS12);
    TCCR1B ^= (1<<CS10);
}
void stopTimer0()
{
    TCCR0B &= ~(1<<CS01);
    TIMSK0 &= ~(1<<OCIE0A);
}

int main()
{

    InitPort();
    adc_init();
    clearADCChannel(0);
    TimersInit();
    startTimer1();
    startTimer2();

    ADCEnableDisable(1);

    sei();
    OCR1AH = 0x00;

    while(1)
    {
```

```
        _delay_ms(50);
        readValue = ((adc_in/1023) * 0xFF);
        OCR1A = (readValue);
    }

}

ISR(ADC_vect){
    volatile uint8_t ADC_in_L = 0;
    volatile uint8_t ADC_in_H = 0;
    // clear timer compare match flag
    TIFR0 = (1<<OCF0A);
    // Toggle pin PD2 to track the end of ADC conversion
    PIND = (1<<PIND2);
    ADC_in_L = ADCL;
    ADC_in_H = ADCH;
    adc_in = ((ADC_in_H << 8)|(ADC_in_L));
}

ISR(PCINT1_vect){
    if(!(PINC & (1 << PINC1))){
        OCR1A = 0;
        DDRB^= (1<<PINB1);
    }
}
```

3. DEVELOPED MODIFIED CODE OF TASK 2/A from TASK 1/A

n/a

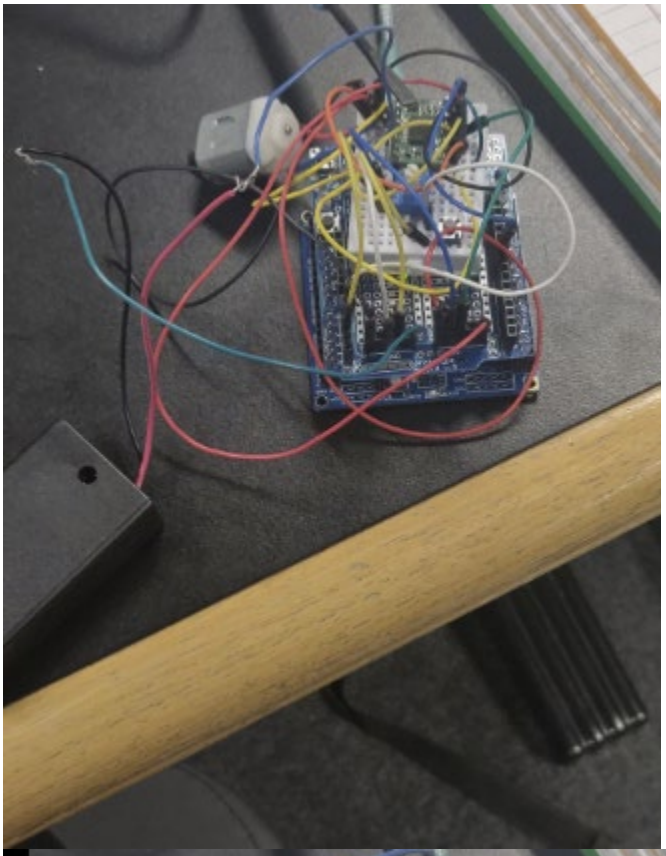
4. SCHEMATICS

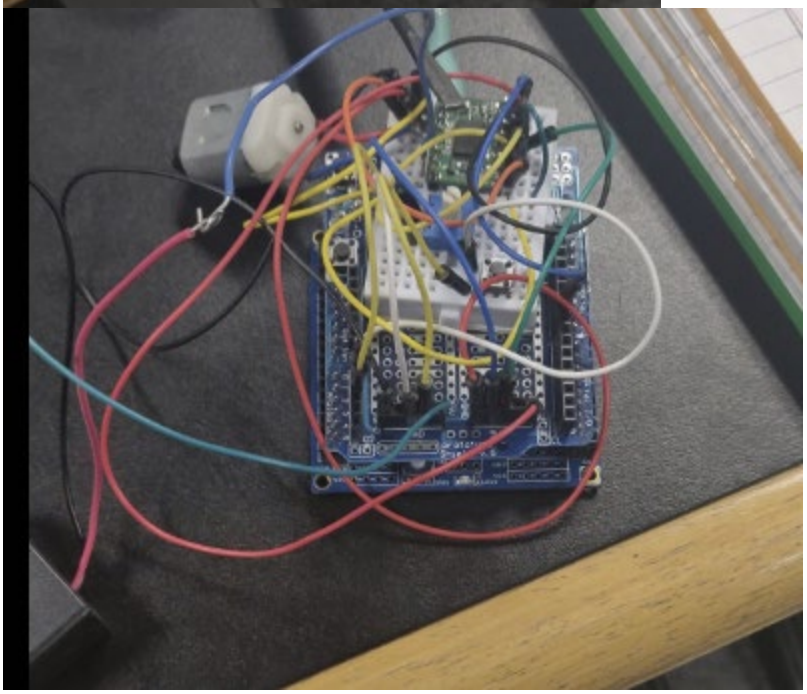


5. SCREENSHOTS OF EACH TASK OUTPUT (ATMEL STUDIO OUTPUT)

n/a

6. SCREENSHOT OF EACH DEMO (BOARD SETUP)





7. VIDEO LINKS OF EACH DEMO

https://youtu.be/qPurNGfIqfU

8. GITHUB LINK OF THIS DA

https://github.com/Alira-Coffman/submission-repo/tree/master/ESD301/DA/DA4A

Student Academic Misconduct Policy
http://studentconduct.unlv.edu/misconduct/policy.html

"*This assignment submission is my own, original work*".
Alira Coffman