CPE301 – SPRING 2019

Design Assignment 4B

Student Name: Tian Chen

Student #: 2001086575

Student Email: chent5@unlv.nevada.edu

Primary Github address: <https://github.com/Chentian12138/AAAABBBBB>

Directory: <https://github.com/Chentian12138/AAAABBBBB>

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**

Atmega 328p

Stepper motor

Servo

Potentiometer

ULN2003 stepper motor driver

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

#define F\_CPU 1000000UL

#include <avr/io.h>

#include <util/delay.h>

#include <stdio.h>

#include <avr/interrupt.h>

#include <stdlib.h>

void read\_adc(void); //Read temp with the LM35 using ADC

void adc\_init(void); // Initialize ADC

volatile unsigned int speed; // raw temp value from adc

int main(void)

{

PORTC^= (1<<5);

DDRB= 0xFF; // set the b set of pin out

adc\_init(); //initialize adc

\_delay\_ms(100);

while(1){

read\_adc();

speed = speed/9;

PORTB = 0x06;

\_delay\_ms(speed); // 639c is to set it to the clockwise

PORTB = 0x03;

\_delay\_ms(speed);

PORTB = 0x09;

\_delay\_ms(speed);

PORTB = 0x0c;

\_delay\_ms(speed);

}

}

void adc\_init(void) //initialize ADC

{

ADMUX = (0<<REFS1)| // Reference Selection Bits

(1<<REFS0)| // AVcc - external cap at AREF

(0<<ADLAR)| // ADC Left Adjust Result

(1<<MUX2)| // Analog Channel Selection Bits

(0<<MUX1)| // ADC2 (PC2 PIN25)

(0<<MUX0); // 010

ADCSRA = (1<<ADEN)| // ADC ENable

(0<<ADSC)| // ADC Start Conversion

(0<<ADATE)| // ADC Auto Trigger Enable

(0<<ADIF)| // ADC Interrupt Flag

(0<<ADIE)| // ADC Interrupt Enable

(1<<ADPS2)| // ADC Prescaler Select Bits

(0<<ADPS1)|

(1<<ADPS0);

}

void read\_adc(void) {

unsigned char i =4;

speed = 0; //initialize adc\_temp variable

while (i--) {

ADCSRA |= (1<<ADSC);

while(ADCSRA & (1<<ADSC));

speed+= ADC;

\_delay\_ms(50);

}

speed = speed / 4; // Average a few samples

}

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

#define F\_CPU 1000000UL

#include <avr/io.h>

#include <util/delay.h>

#include <stdio.h>

#include <avr/interrupt.h>

#include <stdlib.h>

void read\_adc(void); //Read angle

void adc\_init(void); // Initialize ADC

volatile unsigned int angle; // angle value

int main(void)

{

DDRD = 0xFF;

TCCR0B=3;

TCCR0A=0x83;

adc\_init(); //initialize adc

\_delay\_ms(100);

while (1)

{

read\_adc();

OCR0A = angle; // set the angle

\_delay\_ms(500);

OCR0A = angle; // set the angle

\_delay\_ms(500);

}

}

void adc\_init(void) //initialize ADC

{

ADMUX = (0<<REFS1)| // Reference Selection Bits

(1<<REFS0)| // AVcc - external cap at AREF

(0<<ADLAR)| // ADC Left Adjust Result

(1<<MUX2)| // Analog Channel Selection Bits

(0<<MUX1)| // ADC2 (PC2 PIN25)

(0<<MUX0); // 010

ADCSRA = (1<<ADEN)| // ADC ENable

(0<<ADSC)| // ADC Start Conversion

(0<<ADATE)| // ADC Auto Trigger Enable

(0<<ADIF)| // ADC Interrupt Flag

(0<<ADIE)| // ADC Interrupt Enable

(1<<ADPS2)| // ADC Prescaler Select Bits

(0<<ADPS1)|

(1<<ADPS0);

}

void read\_adc(void) {

unsigned char i =4;

angle = 0; //initialize adc\_temp variable

while (i--) {

ADCSRA |= (1<<ADSC);

while(ADCSRA & (1<<ADSC));

angle+= ADC;

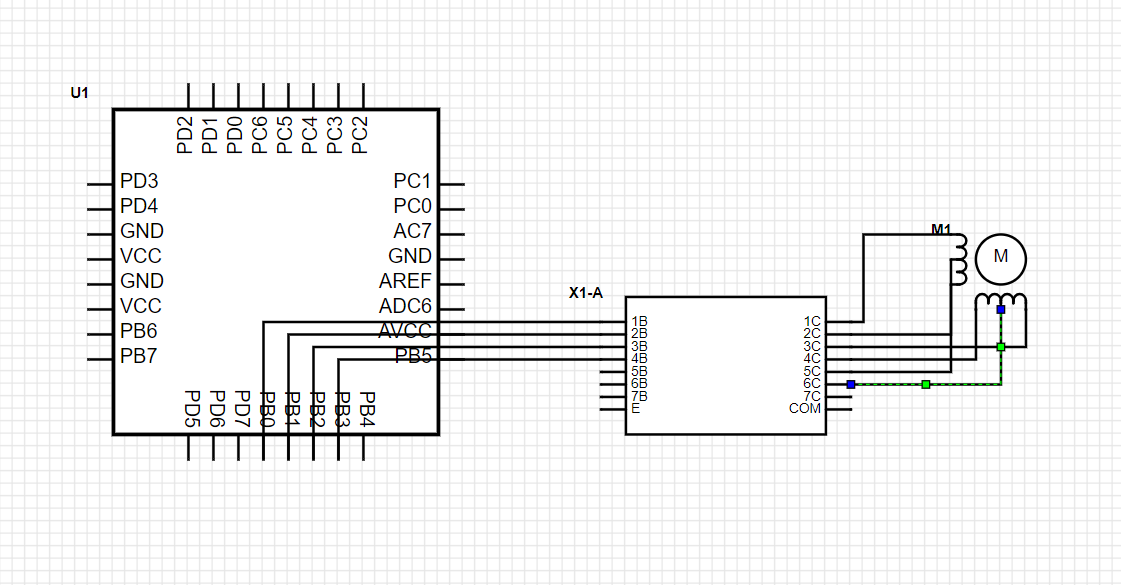
\_delay\_ms(50);

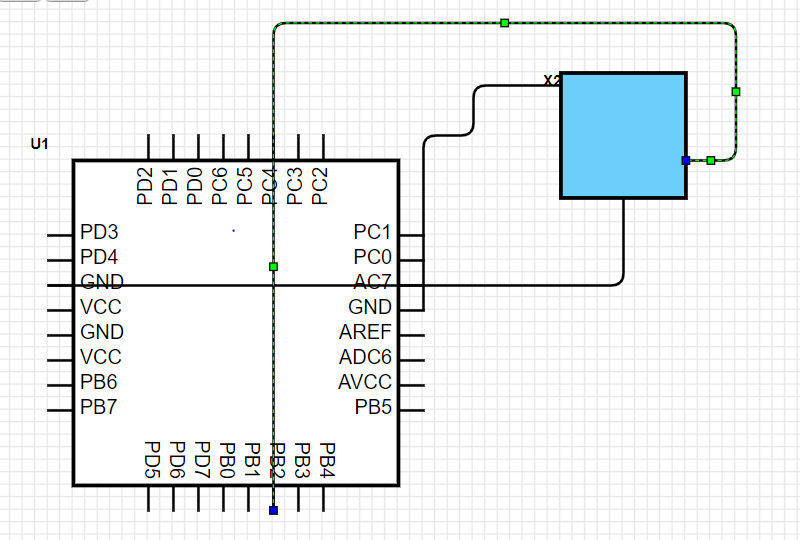
}

angle = angle / 4; // Average a few samples

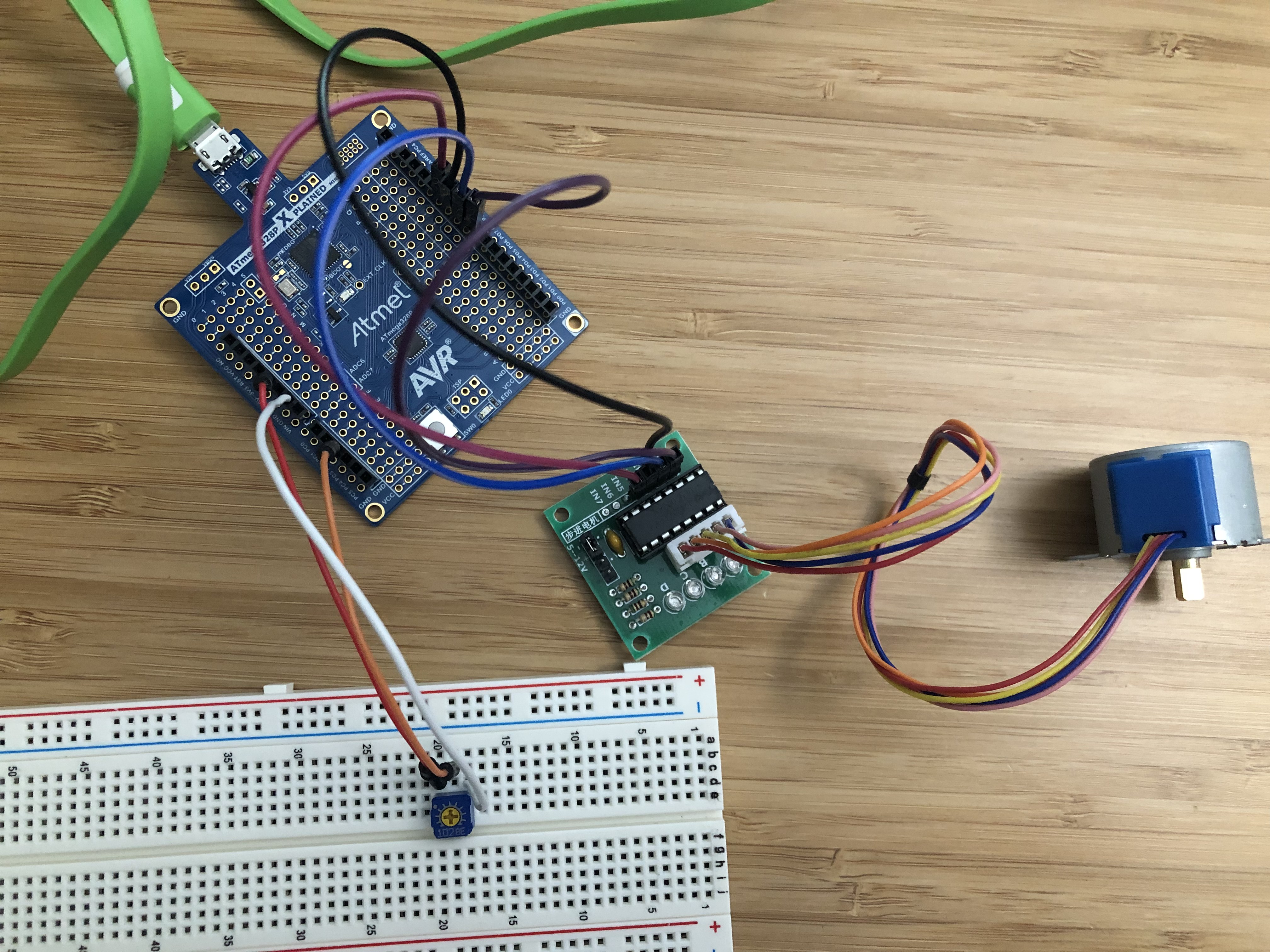
}

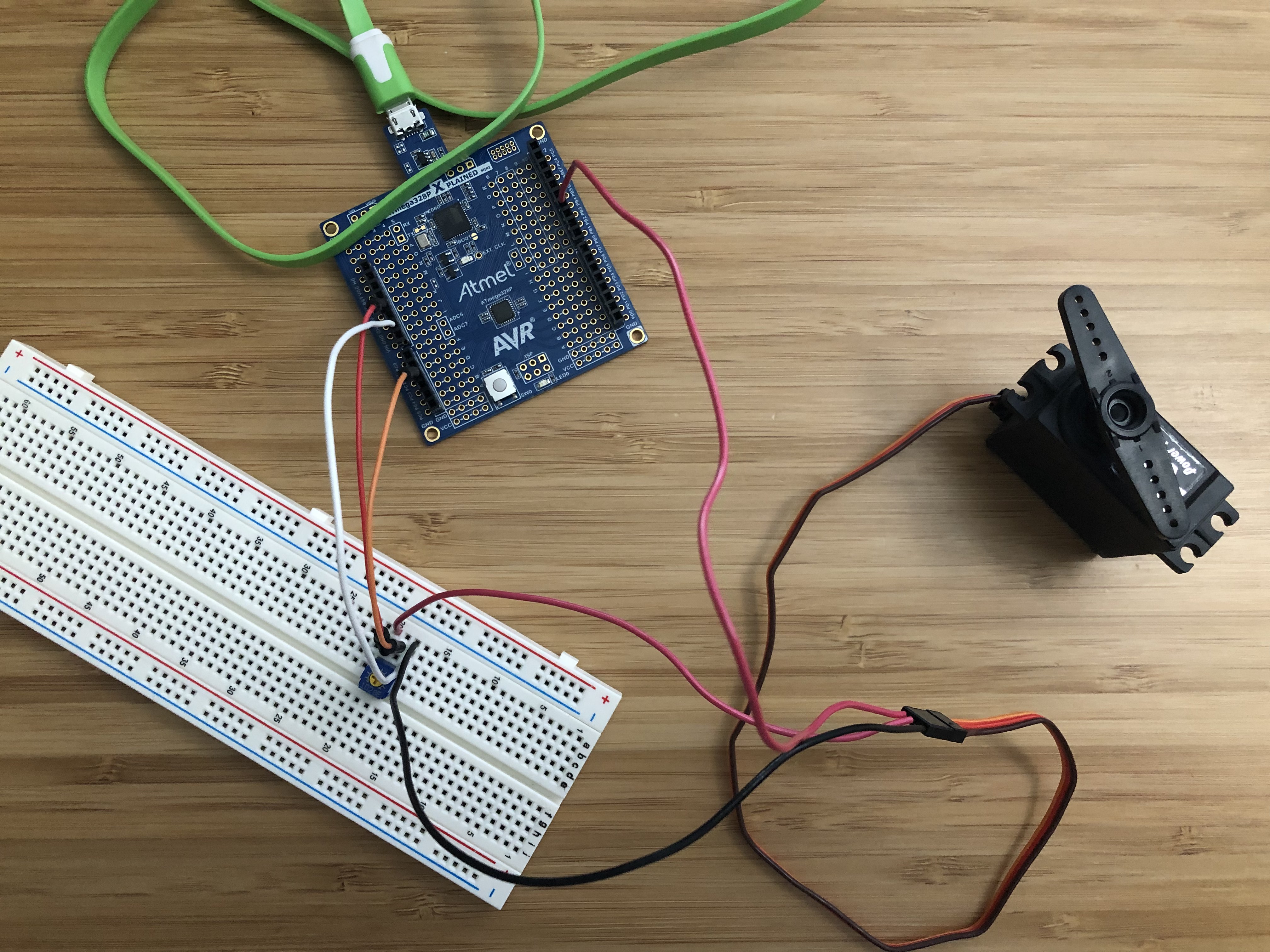
1. **SCHEMATICS**





1. **SCREENSHOT OF EACH DEMO (BOARD SETUP)**





1. **GITHUB LINK OF THIS DA**

<https://github.com/Chentian12138/AAAABBBBB>

**Student Academic Misconduct Policy**

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

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

NAME OF THE STUDENT