Your NAME

CPE301 – SPRING 2016

Design Assignment X

**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 |  |  |
| 1. | CODE OF TASK 1 |  |  |
| 2. | CODE OF TASK 2 |  |  |
| 3. | CODE OF TASK 3 |  |  |
| 4. | SCHEMATICS |  |  |
| 5. | FLOWCHART |  |  |
| 6. | SCREENSHOT OF EACH DEMO |  |  |
| 7. | VIDEO LINKS OF EACH DEMO |  |  |
| 8. | GOOGLECODE LINK OF THE DA |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 0. | COMPONENTS LIST |  |  |

* Atmega328p
* DC Motor
* Stepper Motor
* Servo Motor
* 1k Potentiometer
* TIP120
* ILD74 Opto-isolator
* ULN2003A

|  |  |  |  |
| --- | --- | --- | --- |
| 1. | CODE OF TASK 1 |  |  |

/\*

\* Da 5 Task 1.c

\*

\* Created: 4/7/2016 3:24:01 PM

\* Author : Dominique

\*/

#define F\_CPU 8000000UL

#include <avr/io.h>

#include <avr/sfr\_defs.h>

#include <util/delay.h> // Needed for delay function

#include <stdint.h> // needed for uint8\_t

#define ENABLE 6

void Delay**(**unsigned int time**);**

float ADC\_ratio **=** .0009775**;** // 1/1023

int time **=** 50**;**

int value**;**

int main**(**void**)**

**{**

DDRD **|=** **(**1**<<**6**);** // Enable Port D.6 as an output

PORTD **=** **(**1**<<**ENABLE**);** // Output 1 at Port D.6

/\*Enable the ADC Conversion\*/

ADMUX **=** 0**;** // use ADC0

ADMUX **|=** **(**1 **<<** REFS0**);** // use AVcc as the reference

ADCSRA **|=** **(**1 **<<** ADPS0**)** **|** **(**1 **<<** ADPS1**);** // 8 prescaler

ADCSRA **|=** **(**1 **<<** ADEN**);** // Enable the ADC

ADCSRA **|=** **(**1 **<<** ADSC**);** // Start the ADC conversion

**while** **(**1**)** // Infinite while loop for code execution

**{**

ADCSRA **|=** **(**1 **<<** ADSC**);** // Start the ADC conversion

loop\_until\_bit\_is\_clear**(**ADCSRA**,** ADSC**);**

value **=** ADC**;**

PORTD **=** PORTD **|** **(**1**<<**ENABLE**);**

Delay**((**unsigned int**)**time**\***value**\***ADC\_ratio**);**

PORTD **=** PORTD **&** **(~(**1**<<**ENABLE**));**

Delay**((**unsigned int**)**time**\*(**1**-(**value**\***ADC\_ratio**)));**

**}**

**return** 0**;**

**}**

// Function delays for the number of ms provided

void Delay**(**unsigned int length**)**

**{**

int i **=** 0**;**

**for** **(**i **=** 0**;** i **<** length**;** i**++)**

\_delay\_ms**(**1**);**

**}**

|  |  |  |  |
| --- | --- | --- | --- |
| 2. | CODE OF TASK 2 |  |  |

/\*

\* Da 5 Task 2.c

\*

\* Created: 4/7/2016 6:27:28 PM

\* Author : Dominique

\*/

#define F\_CPU 8000000UL

#include <avr/io.h>

#include <avr/sfr\_defs.h> // Needed for if\_bit\_is\_clear function

#include <util/delay.h> // Needed for delay function

void Delay**(**unsigned int time**);**

float ADC\_ratio **=** .0009775**;** // 1/1023

int time **=** 50**;**

int value**;**

int main**(**void**)**

**{**

DDRB **=** 0xFF**;**

/\*Enable the ADC Conversion\*/

ADMUX **=** 0**;** // use ADC0

ADMUX **|=** **(**1 **<<** REFS0**);** // use AVcc as the reference

ADCSRA **|=** **(**1 **<<** ADPS0**)** **|** **(**1 **<<** ADPS1**);** // 8 prescaler

ADCSRA **|=** **(**1 **<<** ADEN**);** // Enable the ADC

ADCSRA **|=** **(**1 **<<** ADSC**);** // Start the ADC conversion

**while** **(**1**)** // Infinite while loop for code execution

**{**

ADCSRA **|=** **(**1 **<<** ADSC**);** // Start the ADC conversion

loop\_until\_bit\_is\_clear**(**ADCSRA**,** ADSC**);**

value **=** ADC**;**

PORTB **=** 0X66**;**

Delay**((**unsigned int**)**time**\***value**\***ADC\_ratio**);**

PORTB **=** 0xCC**;**

Delay**((**unsigned int**)**time**\***value**\***ADC\_ratio**);**

PORTB **=** 0x99**;**

Delay**((**unsigned int**)**time**\***value**\***ADC\_ratio**);**

PORTB **=** 0x33**;**

Delay**((**unsigned int**)**time**\***value**\***ADC\_ratio**);**

**}**

**return** 0**;**

**}**

// Function delays for the number of ms provided

void Delay**(**unsigned int length**)**

**{**

int i **=** 0**;**

**for** **(**i **=** 0**;** i **<** length**;** i**++)**

\_delay\_ms**(**1**);**

**}**

|  |  |  |  |
| --- | --- | --- | --- |
| 3. | CODE OF TASK 3 |  |  |

/\*

\* Da 5 Task 3.c

\*

\* Created: 4/7/2016 6:46:02 PM

\* Author : Dominique

\*/

#define F\_CPU 8000000UL

#include <avr/io.h>

#include <avr/sfr\_defs.h> // Needed for if\_bit\_is\_clear function

#include <util/delay.h>

void Delay**(**unsigned int time**);**

const float ADC\_ratio **=** .0009775**;** // 1/1023

int value**;**

int main**()**

**{**

int x**;**

int angle**;**

DDRB **=** 0xFF**;** // Set port B as an output for timer 1

/\*Enable the ADC Conversion\*/

ADMUX **=** 0**;** // use ADC0

ADMUX **|=** **(**1 **<<** REFS0**);** // use AVcc as the reference

ADCSRA **|=** **(**1 **<<** ADPS0**)** **|** **(**1 **<<** ADPS1**);** // 8 prescaler

ADCSRA **|=** **(**1 **<<** ADEN**);** // Enable the ADC

ADCSRA **|=** **(**1 **<<** ADSC**);** // Start the ADC conversion

//TOP = ICR1;

//output compare OC1A 8 bit non inverted PWM

//Clear OC1A on Compare Match, set OC1A at TOP

//Fast PWM

//ICR1 = 20000 defines 50Hz pwm

ICR1 **=** 20000**;**

OCR1A **=** 500**;**

TCCR1A**|=(**0**<<**COM1A0**)|(**1**<<**COM1A1**)|(**0**<<**COM1B0**)|(**0**<<**COM1B1**)|(**0**<<**FOC1A**)|(**0**<<**FOC1B**)|(**1**<<**WGM11**)|(**0**<<**WGM10**);** //TCCR1A = 0x82

TCCR1B**|=(**0**<<**ICNC1**)|(**0**<<**ICES1**)|(**1**<<**WGM13**)|(**1**<<**WGM12**)|(**0**<<**CS12**)|(**1**<<**CS11**)|(**0**<<**CS10**);** //TCCR1B = 0x1A

**while** **(**1**)**

**{**

ADCSRA **|=** **(**1 **<<** ADSC**);** // Start the ADC conversion

loop\_until\_bit\_is\_set**(**ADCSRA**,** ADSC**);**

value **=** ADC**;**

angle **=** 1800**\*(**1**-(**ADC\_ratio**\***value**));** // Total value of 500 is approximately 0 degrees

angle **=** 500 **+** angle**;** // Total value of 2300 is approximately 180 degrees

OCR1A **=** angle**;**

**for(**x **=** 0**;** x**<**500**;**x**++)**

\_delay\_ms**(**1**);**

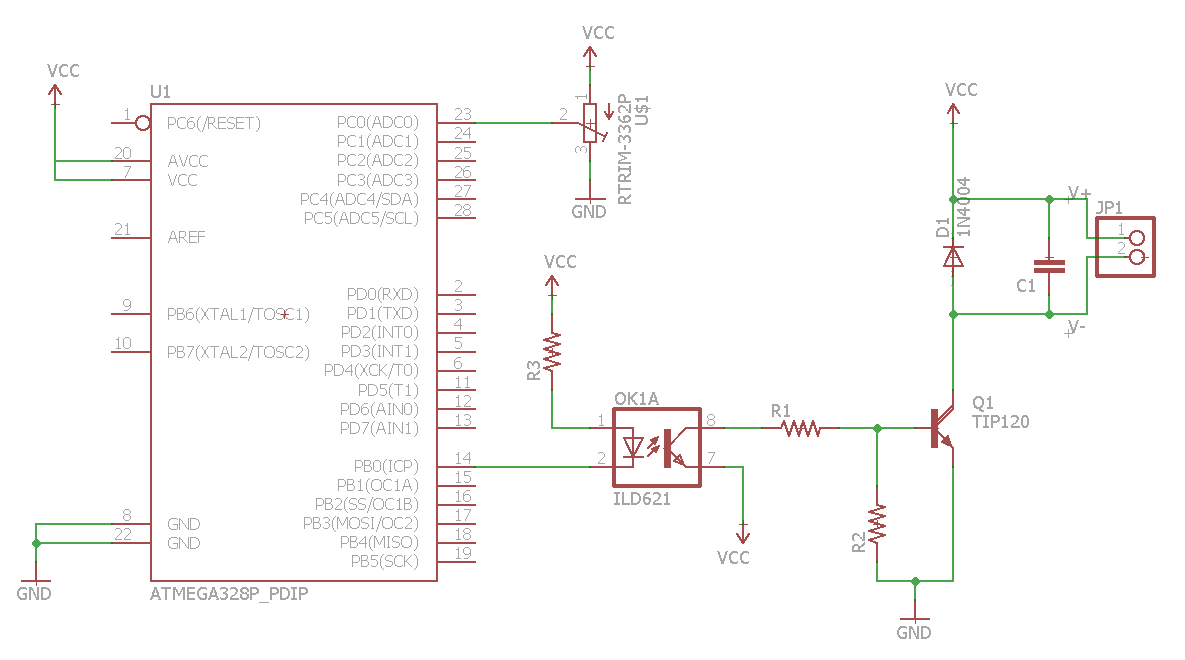
**}**

**return** 0**;**

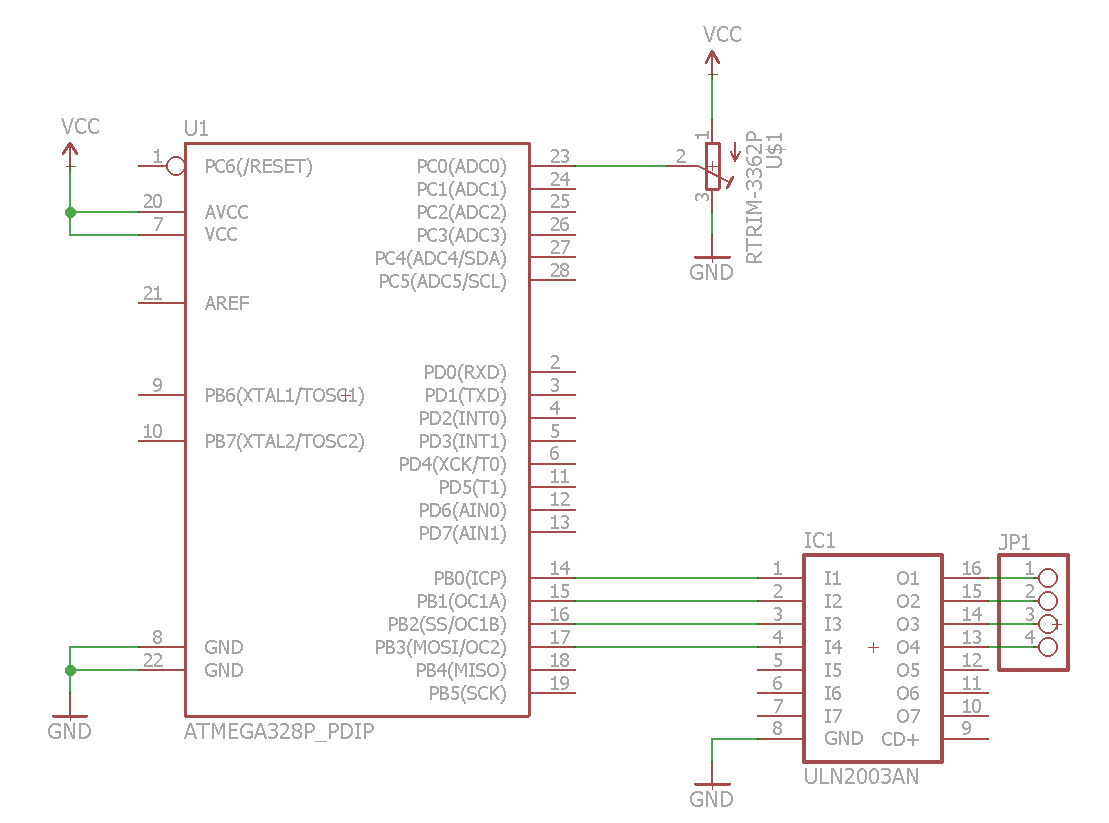
**}**

|  |  |  |  |
| --- | --- | --- | --- |
| 4. | SCHEMATICS |  |  |

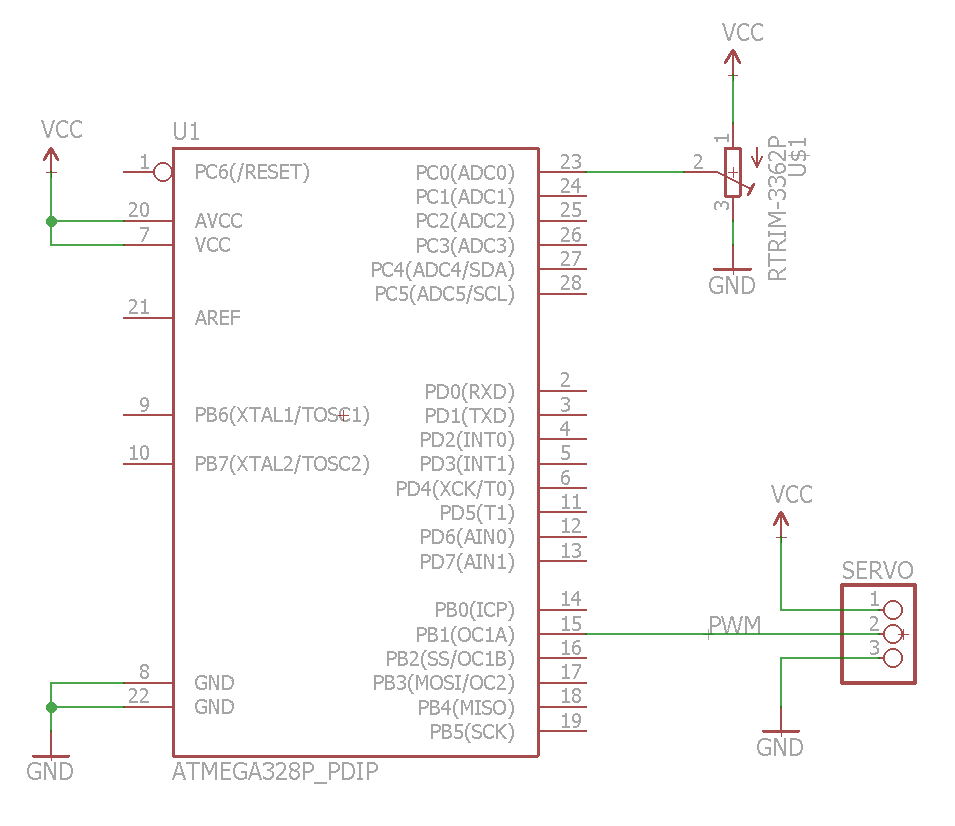
DC Motor Schematic



Stepper Motor Schematic



Servo Motor Schematic



|  |  |  |  |
| --- | --- | --- | --- |
| 5. | FLOW CHART |  |  |

Initialize ADC

Read ADC Value

Delay

Convert ADC value to a ratio used for the delay

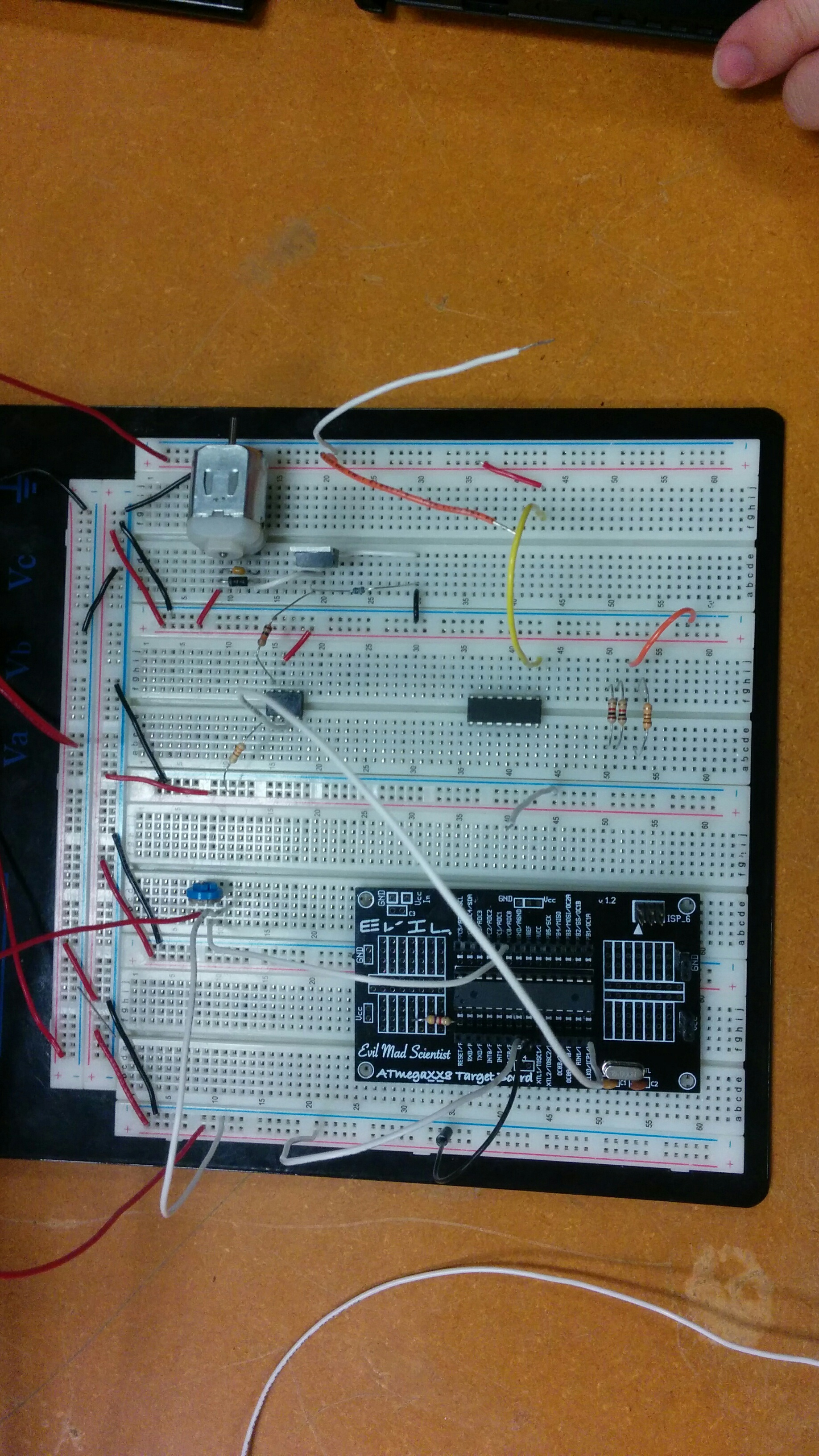
Output the signal

Turn off the Signal

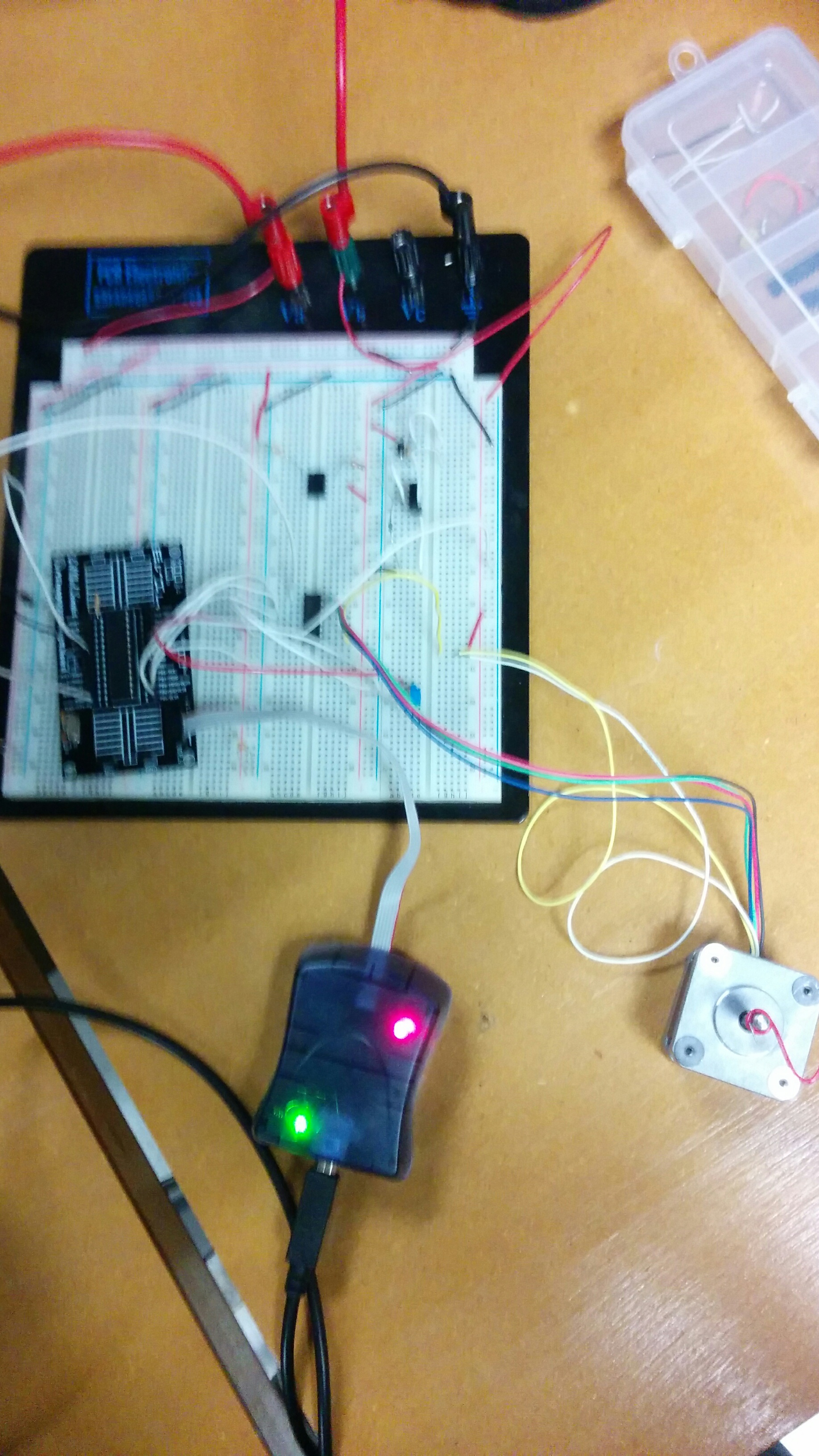
Delay

|  |  |  |  |
| --- | --- | --- | --- |
| 6. | SCREENSHOT OF EACH DEMO |  |  |

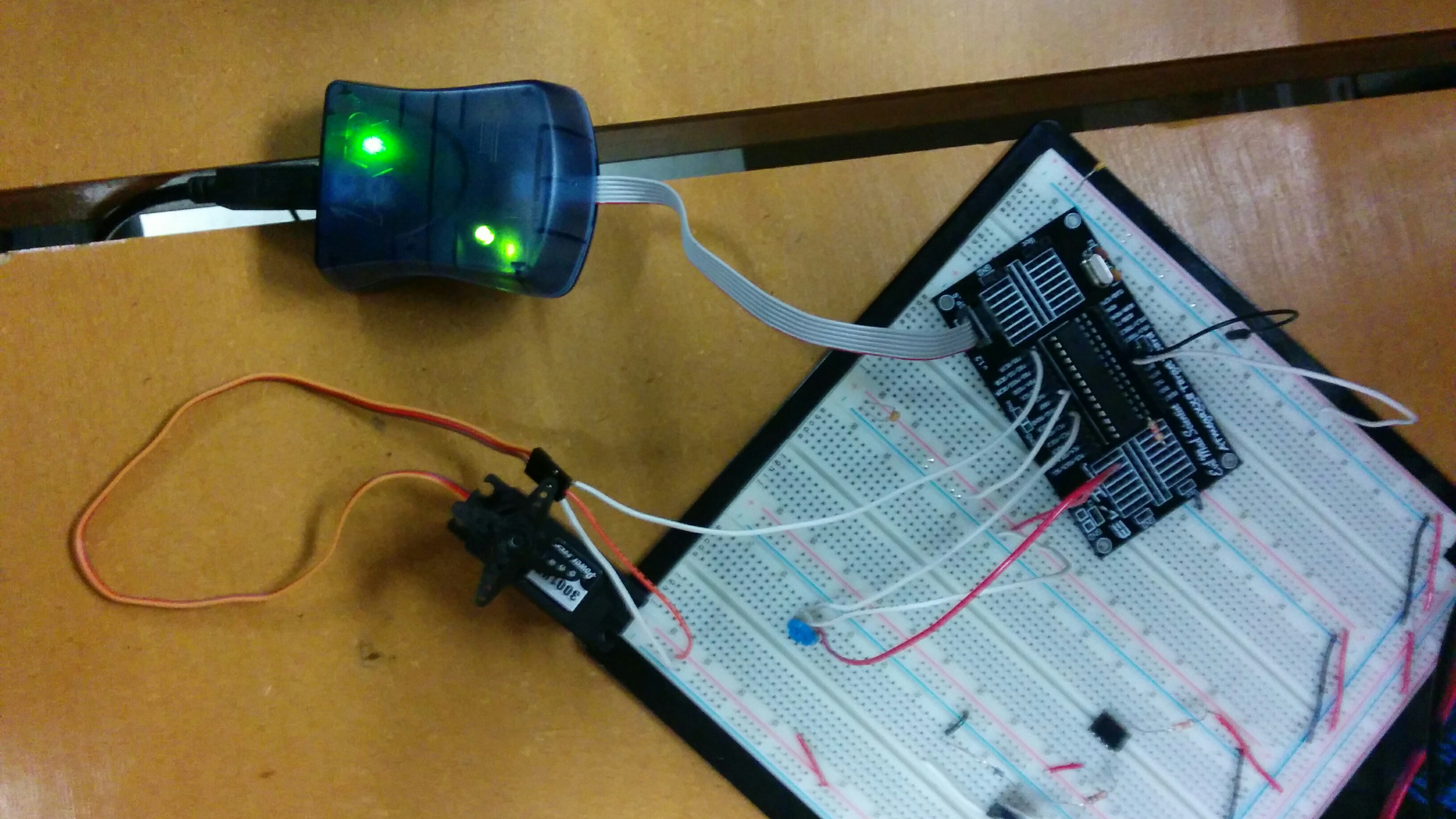
TASK 1: DC Motor Speed Controlled by Potentiometer



TASK 2: Stepper Motor Speed Controlled by Potentiometer



TASK 3: Servo Motor position controlled by Potentiometer



|  |  |  |  |
| --- | --- | --- | --- |
| 7. | VIDEO LINKS OF EACH DEMO |  |  |
| Task 1 - <https://www.youtube.com/watch?v=OU0c8zZMa08> | | | |
| Task 2 - <https://www.youtube.com/watch?v=ZmtLqWix2Vo> | | | |
| Task 3 - <https://www.youtube.com/watch?v=iL-xpnwHwXg> | | | |
| 8. | GOOGLECODE LINK OF THE DA |  |  |
| <https://github.com/Anguian3/anguian3-submissions> | | | |

**Student Academic Misconduct Policy**

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

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

NAME OF THE STUDENT