CPE301 – SPRING 2019

Design Assignment 4

Student Name: Victor Martinez

Student #: 5003131674

Student Email: [martiv6@unlv.nevada.edu](mailto:martiv6@unlv.nevada.edu)

Primary Github address: https://github.com/martiv6

Directory: https://github.com/martiv6/submissions\_da/tree/master/DesignAssignment/DA4

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**

DC Motor

MD08A

(Driver)

BREAD

BOARD

328P

Mini

POWERSUPPLY

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

/\*

\* DA4A.c

\*

\* Created: 4/10/2019 10:48:31 AM

\* Author : victor

\*/

#define *F\_CPU* 16000000UL

#include <avr/io.h>

#include <avr/interrupt.h>

#include <util/delay.h>

void adc\_initializer(void); // void function we will use for ADC

int motor\_control; // to be able to control the motor with the potentiometer

int motor\_status = 0; // to toggle the motor with the switch will be off at first

int main()

{

DDRD = 0b01000000; // sets port d as output

DDRC = 0b0010; // sets port c as output

PORTC |= (0b0001<<0b0001); // enable pull-up pin

PCICR = 0b0010; // enable PCIE1 to make PCMSK1 work

PCMSK1 = 0b0010; // enable pin changes

TCCR0A= 0b10000011; // set fast PWM and clears OCR0A on MATCH

TCCR0B= 0b0101; // set prescaler to 1024

sei(); // enable interrupts

adc\_initializer(); // initializes ADC

while (1)

{ // waits for an interrupt

}

}

ISR(PCINT1\_vect)

{

if(!(PINC & (1<<PINC1))) // when the button is pressed it goes into this loop to determine if will be on or off

{

if(motor\_status == 0) // when the button is pressed the motor will stop

{

OCR0A = 0; // stops the motor

*\_delay\_ms*(1500); // meant for debouncing to make sure its a clean reading

}

if (motor\_status == 1) // when the button is pressed again it will stay on

{

while((ADCSRA&(1<<ADIF))==0); // wait for conversion

motor\_control = ADC; // puts ADC value into motor control so we can then keep it on

OCR0A = motor\_control; // Output to converted value to 0CR0A and makes the motor run

*\_delay\_ms*(1500); // meant for debouncing to make sure its a clean reading

}

motor\_status ^= 1; // update state of motor to on

}

}

void adc\_initializer(void) //initialize ADC

{

ADMUX = (0<<REFS1); // ADC ENable

ADCSRA = (1<<ADEN)| (1<<ADSC)| (1<<ADATE)| (0<<ADIF)| // ADC Start Conversion

(0<<ADIE)| (1<<ADPS2)|(1<<ADPS1)| (1<<ADPS0); // ADC Auto Trigger Enable

// ADC Interrupt Flag

// ADC Interrupt Enable

// ADC Prescaler Select Bits

// ADC Prescaler select bits

// ADC input

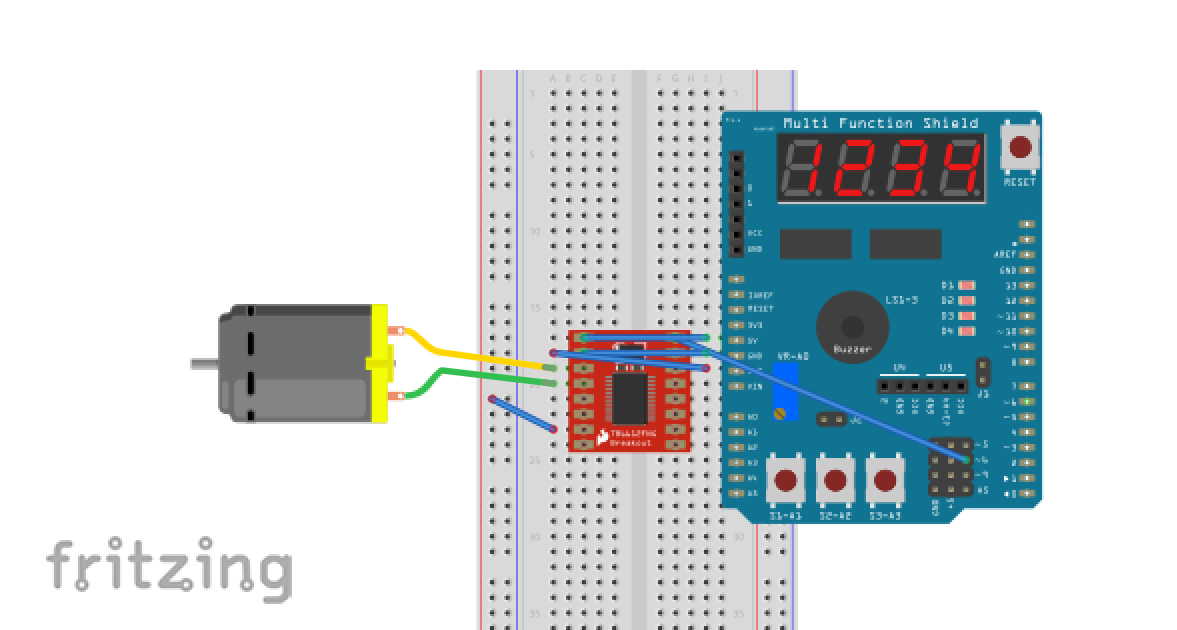
// Timer/Counter1 Interrupt Mask Register

}

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

**N/A**

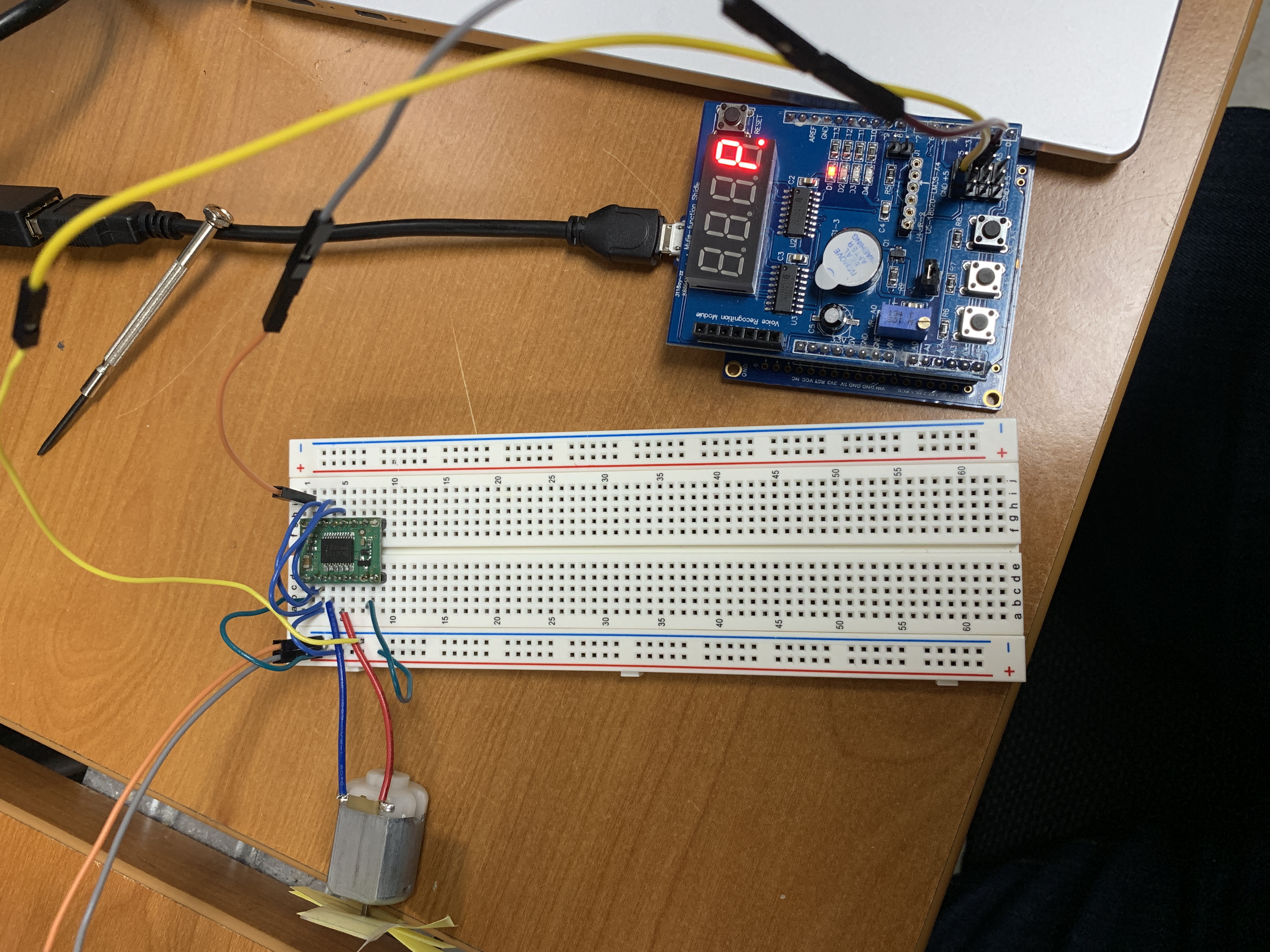
1. **SCHEMATICS**



1. **SCREENSHOTS OF EACH TASK OUTPUT (ATMEL STUDIO OUTPUT)**

N/A

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



1. **VIDEO LINKS OF EACH DEMO**

<https://youtu.be/Bv9w_7TFTfo>

1. **GITHUB LINK OF THIS DA**

https://github.com/martiv6/submissions\_da/tree/master/DesignAssignment/DA4

**Student Academic Misconduct Policy**

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

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

VICTOR MARTINEZ