CPE301 – Fall 2019

Design Assignment 4A

Student Name: Jacob Patrick Reed

Student #: 1008448895

Student Email: reedj35@unlv.nevada.edu

Primary Github address: <https://github.com/reedjacobp>

Directory: <https://github.com/reedjacobp/submission_da>

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**

* ATmega328PB Xplained Mini
* Multifunction Shield
* TB6612FNG Motor Driver
* DC Motor
* Logic Analyzer

1. **DEVELOPED CODE OF TASK 1**

/\*

\* DA4A.c

\*

\* Created: 10/31/2019 4:26:26 PM

\* Author : jreed

\*/

#define *F\_CPU* 16000000UL // 16MHz

#include <avr/io.h> //

#include <avr/interrupt.h> //

#include <util/delay.h> //

void init\_adc(void);

int control; // VARIABLE FOR ADC VALUE (NOT NEEDED)

int toggle = 0; // USED TO DETERMINE IF BUTTON SHOULD BE ON OR OFF

int main()

{

DDRB |= (1 << DDRB1) | (1 << DDRB2); //make PB1 and PB2 as output

DDRD |= (1 << DDRD6); //make PD6 as output for PWM

DDRC &= ~(1 << DDRC0); //PC0 is an input

DDRC &= ~(1 << DDRC1); //PC1 is an input

PORTC |= (1 << PORTC1); // ENABLE PULL-UP RESISTOR

TCCR0A=0x83; //set up for fast PWM and clear OC0A on compare match

TCCR0B=0x05; //prescalar of 1024

PCICR = 0x02; //enable PCIE1 for PCMSK1 to work

PCMSK1 = 0x02; //enable pin changes on PC1 (PCINT9)

sei(); //enable interrupts

ADMUX = (1<<REFS0); //reference voltage at AREF

ADCSRA = (1<<ADEN)|(1<<ADSC)|(1<<ADATE)|(1<<ADPS2)|(1<<ADPS1)|(1<<ADPS0); //set up for ADC conversion

while (1)

{

//wait here

}

}

ISR(PCINT1\_vect){

if(!(PINC & (1<<PINC1))){

if(toggle == 0){

OCR0A = 0;

PORTB &= ~(1<<PORTB2);

*\_delay\_ms*(1000);

}

if (toggle == 1){

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

control = ADC; // ADC Conversion

OCR0A = control; // Output to converted value to 0CR0A

PORTB |= (1 << PORTB2);

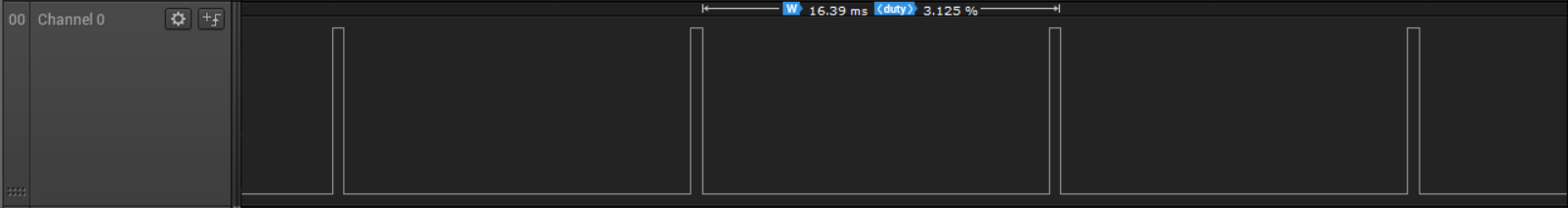
*\_delay\_ms*(1000);

}

toggle ^= 1; //update state of motor to on

}

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

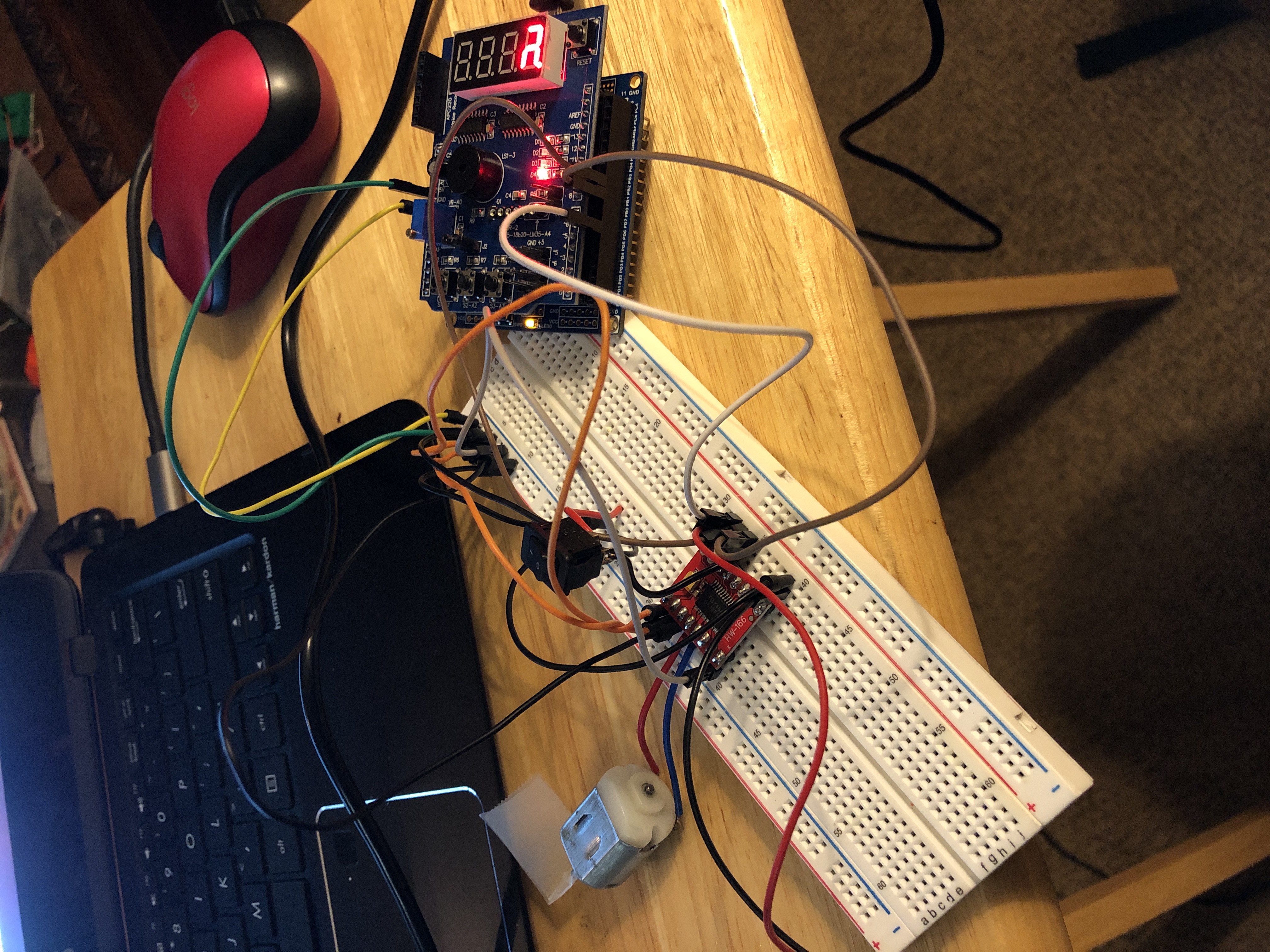


With the potentiometer decreased to minimum



With the potentiometer increase to maximum

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



1. **VIDEO LINKS OF EACH DEMO**

https://youtu.be/\_vvTdJz7MJ8

1. **GITHUB LINK OF THIS DA**

**Student Academic Misconduct Policy**

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

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

NAME OF THE STUDENT