CPE301 – SPRING 2020

Design Assignment 4A

Student Name: Jason Villanueva

Student #: 2001255420

Student Email: villaj21@unlv.nevada.edu

Primary Github address: <https://github.com/jasonvillanuevagit/submission_designAssignments->

Directory: <https://github.com/jasonvillanuevagit/submission_designAssignments-/tree/master/DesignAssignment4A>

1. **COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS**

Atmel Studio 7.0 Atmega328PB-Xmini PC MD08A DC-Motor

- Assembler

- Simulator

- Debugger

A screenshot of a computer

Description automatically generatedA circuit board

Description automatically generated

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

#define F\_CPU 16000000UL

#include <avr/io.h>

#include <util/delay.h>

#include <stdio.h>

#include <avr/interrupt.h>

//global variable

volatile float adc\_value;

//function prototypes

void adc\_init(void);

void read\_adc(void);

//main

int main(void){

adc\_init();//initialize adc

DDRC |= (0<<3);//port c3 input

PORTC |= (1<<3);//activate pullup resistors

PORTC |= (1<<0);

DDRB |= (1<<DDB1);//port pb1 output

PCICR = (1<<PCIE1);//pin change interrupt control register

PCMSK1 = (1<<PCINT11);//enable mask register c3

sei ();//enable global interrupts

ICR1 = 0XFFFF;

TCCR1A |= (1<<COM1A1)|(1<<COM1B1);

TCCR1A |= (1<<WGM11);

TCCR1B |= (1<<WGM12)|(1<<WGM13);

TCCR1B |= (1<<CS10);

while (1 ){

read\_adc();

\_delay\_ms(50);

//PWM at 95% max

if ((adc\_value >= 62258) && (adc\_value < 65535 )){

OCR1A = 62258;

\_delay\_ms(20);

}

//increasing PWM as resistor value increases

else if ((adc\_value < 62257) && (adc\_value >= 3277)){

OCR1A = adc\_value + 30000;

\_delay\_ms(20);

}

//PWM is 0%

else

OCR1A = 0x00;

}

return 0;

}

//interrupt subroutine

ISR(PCINT1\_vect){

DDRC ^= (1<<0);

}

//initializes adc

void adc\_init(void){

ADMUX = (0<<REFS1) | (1<<REFS0) | (1<<ADLAR) | (0<<MUX2) | (0<<MUX1) | (0<<MUX0);

ADCSRA = (1<<ADEN) | (0<<ADSC) | (0<<ADATE) | (0<<ADIF) | (0<<ADIE) | (1<<ADPS2) | (0<<ADPS1) | (1<<ADPS0);

}

//read adc pins

void read\_adc(void){

unsigned char i = 10;//variable for sample amounts

adc\_value = 0;//initialize adc\_value

while(i--){

ADCSRA |= (1<<ADSC);//enable start conversion

while(ADCSRA & (1<<ADSC));//wait until enable & start conversion

adc\_value += ADC;//PC0 value added & stored into adc\_value

}

adc\_value = adc\_value/10;//average of values

}

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

N/A

1. **SCHEMATICS**

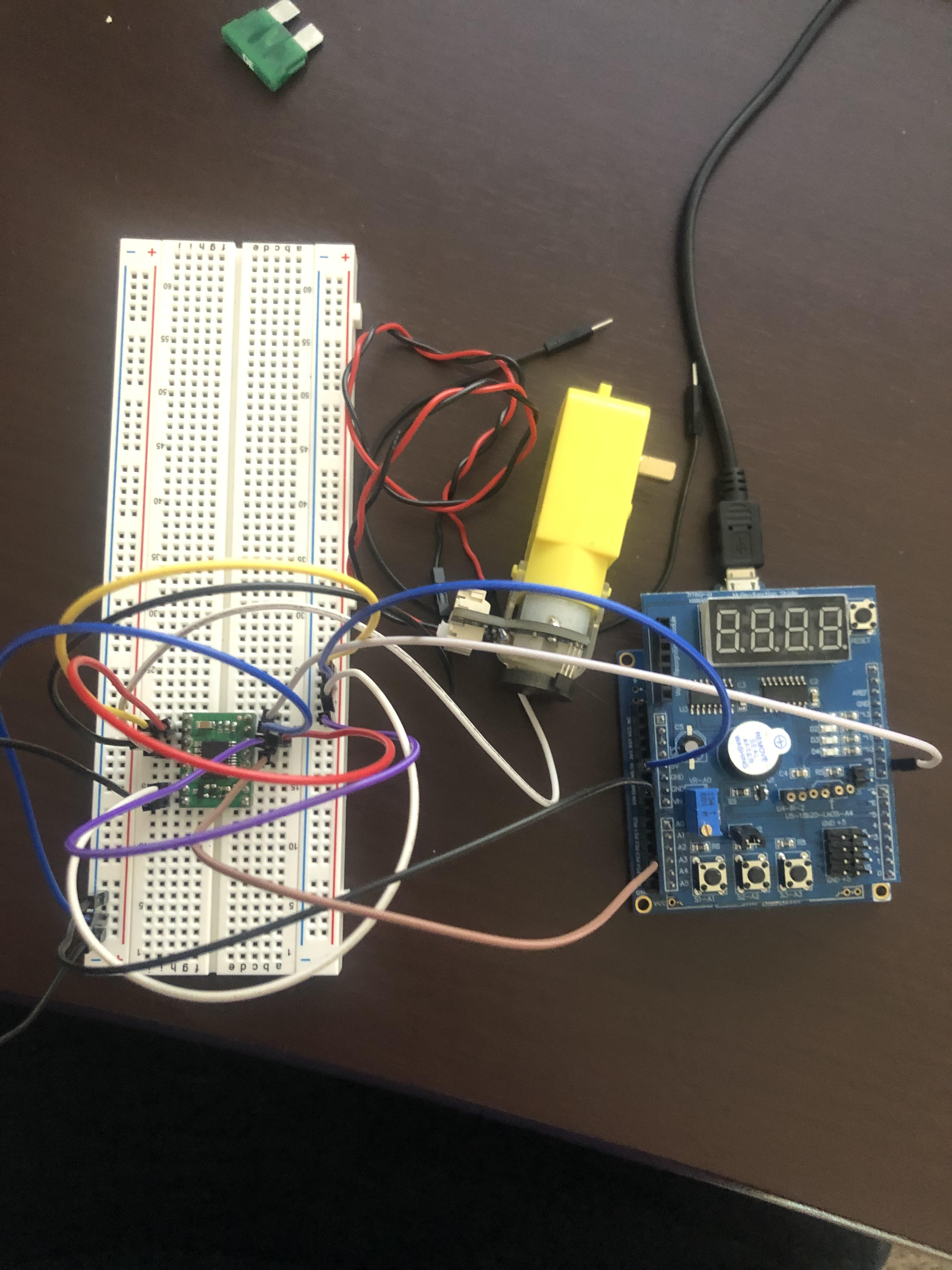
A picture containing screenshot

Description automatically generated

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

N/A

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



1. **VIDEO LINKS OF EACH DEMO**

YouTube Video Demo - <https://youtu.be/BRc20GITtR8>

1. **GITHUB LINK OF THIS DA**

<https://github.com/jasonvillanuevagit/submission_designAssignments-/tree/master/DesignAssignment4A>

**Student Academic Misconduct Policy**

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

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

Jason Villanueva