CPE301 – SPRING 2019

Design Assignment 3B

Student Name: Chris Barr

Student #: 2000682859

Student Email: barrc1@unlv.nevada.edu

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

Directory: https://github.com/BarrChris/submission\_da.git

Submit the following for all Labs:

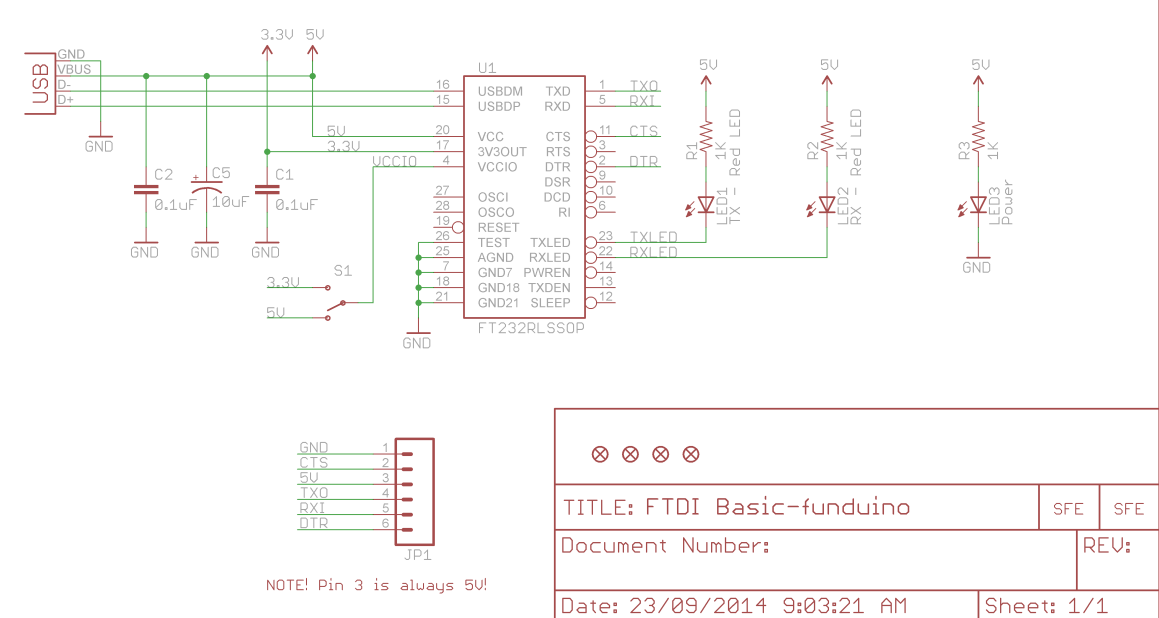
1. In the document, for each task submit the modified or included code (only) with highlights and justifications of the modifications. Also, include the comments.
2. Use the previously create a Github repository with a random name (no CPE/301, Lastname, Firstname). Place all labs under the root folder 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**

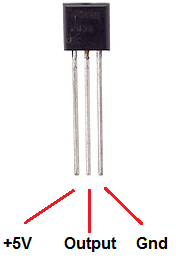
List of Components used

* Atmega328P
* Multi-Function Shield
* FTDI Basic
* 4 M-M wires
* 3 F-F wires
* LM34

Block diagram with pins used in the Atmega328P



This is the diagram for the FTDI



LM34

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

/\*

\* DA3B.c

\*

\* Created: 3/29/2019 1:44:28 PM

\* Author : Chris

\*/

#define *F\_CPU* 16000000UL

#define BAUD\_RATE 9600

#include <avr/io.h>

#include <util/delay.h>

#include <avr/interrupt.h>

void usart\_init ();

void adc\_init();

void timer\_init();

void usart\_send (unsigned char ch);

int main (void)

{

timer\_init ();

usart\_init ();

adc\_init ();

while (1)

{

/\*

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

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

ADCSRA |= (1<<ADIF);

int a = ADCL;

a = a | (ADCH<<8);

a = (a/1024.0) \* 5000/10;

usart\_send((a/100)+'0');

a = a % 100;

usart\_send((a/10)+'0');

a = a % 10;

//a = a \* (9/5) + 32; // celsius to fahrenheit

usart\_send((a)+'0');

usart\_send('\r');

\_delay\_ms(100);

\*/

}

return 0;

}

ISR (TIMER1\_OVF\_vect)

{

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

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

ADCSRA |= (1<<ADIF);

int a = ADCL;

a = a | (ADCH<<8);

a = (a/1024.0) \* 5000/10;

usart\_send((a/100)+'0');

a = a % 100;

usart\_send((a/10)+'0');

a = a % 10;

//a = a \* (9/5) + 32; // celsius to fahrenheit

usart\_send((a)+'0');

usart\_send('\r');

*\_delay\_ms*(100);

TCNT1 = 49911; // Reset timer

}

void usart\_init (void)

{

UCSR0B = (1<<TXEN0);

UCSR0C = (1<< UCSZ01)|(1<<UCSZ00);

UBRR0L = *F\_CPU*/16/BAUD\_RATE-1;

}

void adc\_init (void)

{

/\*\* Setup and enable 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)| // ADC4 (PC4 PIN27)

(1<<MUX0);

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 timer\_init (void)

{

TCCR1B |= 5; //(1 << CS12) | (1 << CS10); // Sets prescaler to 1024

TIMSK1 = (1 << TOIE1); // Enables overflow flag

TCNT1 = 49911; // 1 second delay = (0xFFFF) - TCNT = 65535 - 15624 = 49911

sei();

}

void usart\_send (unsigned char ch)

{

while (! (UCSR0A & (1<<UDRE0))); //wait until UDR0 is empty

UDR0 = ch; //transmit ch

}

void usart\_print(char\* str)

{

int i = 0;

while(str[i] != 0)

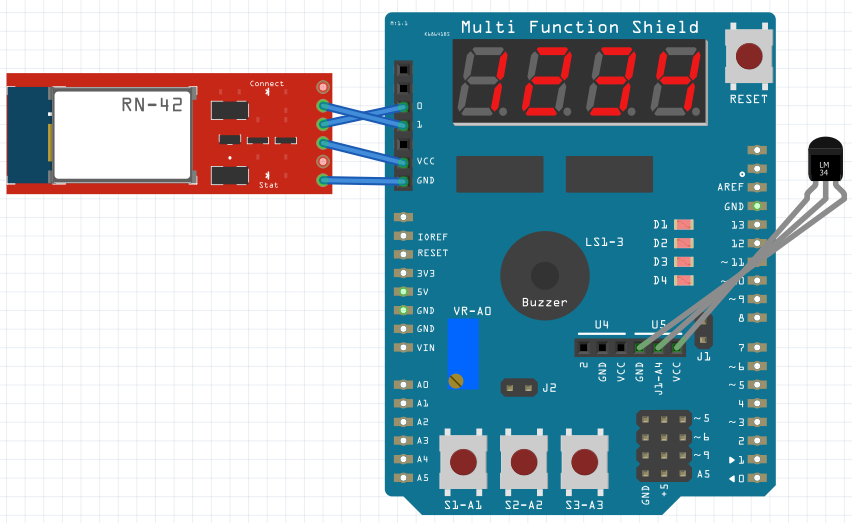
usart\_send(str[i]);

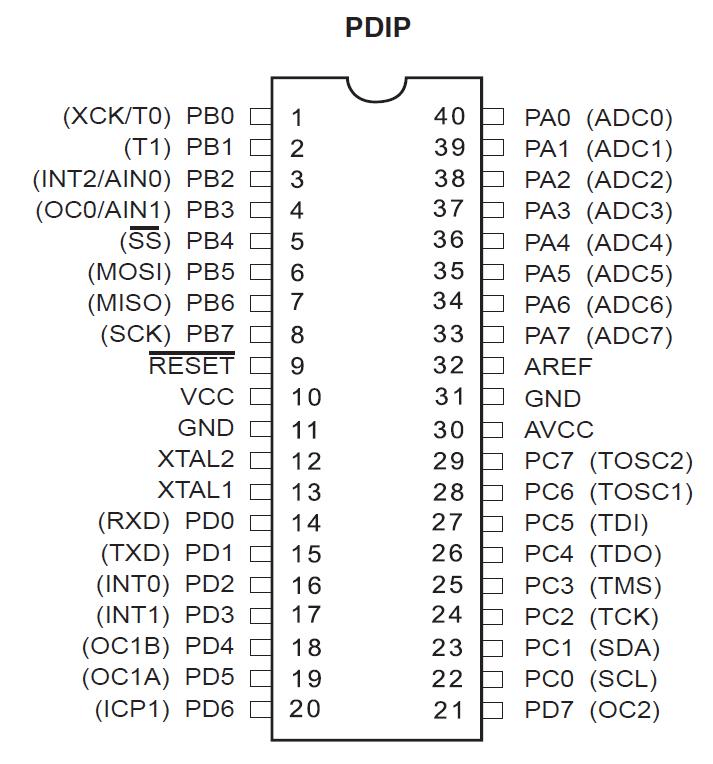
}

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

None modified

1. **SCHEMATICS**





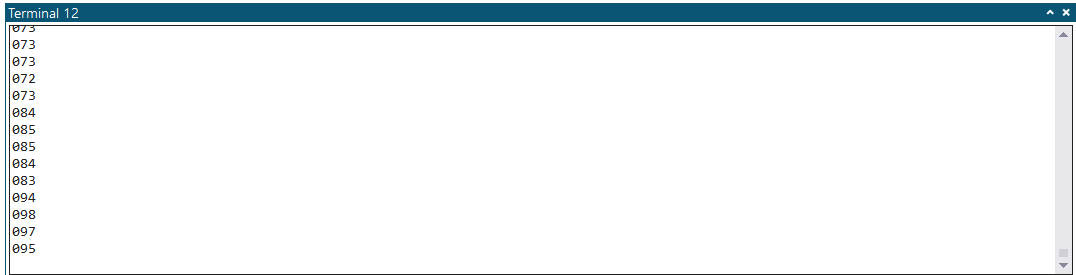
PD0 uses RX

PD1 uses TX

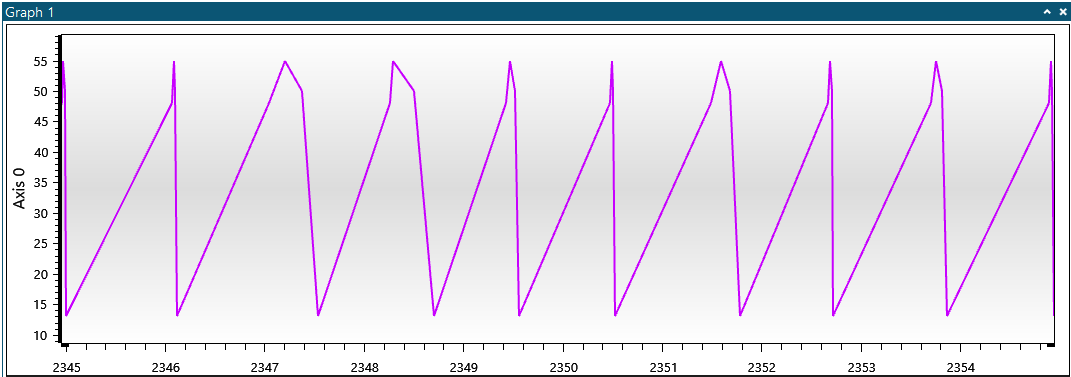
5V to 5V

GND to GND

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

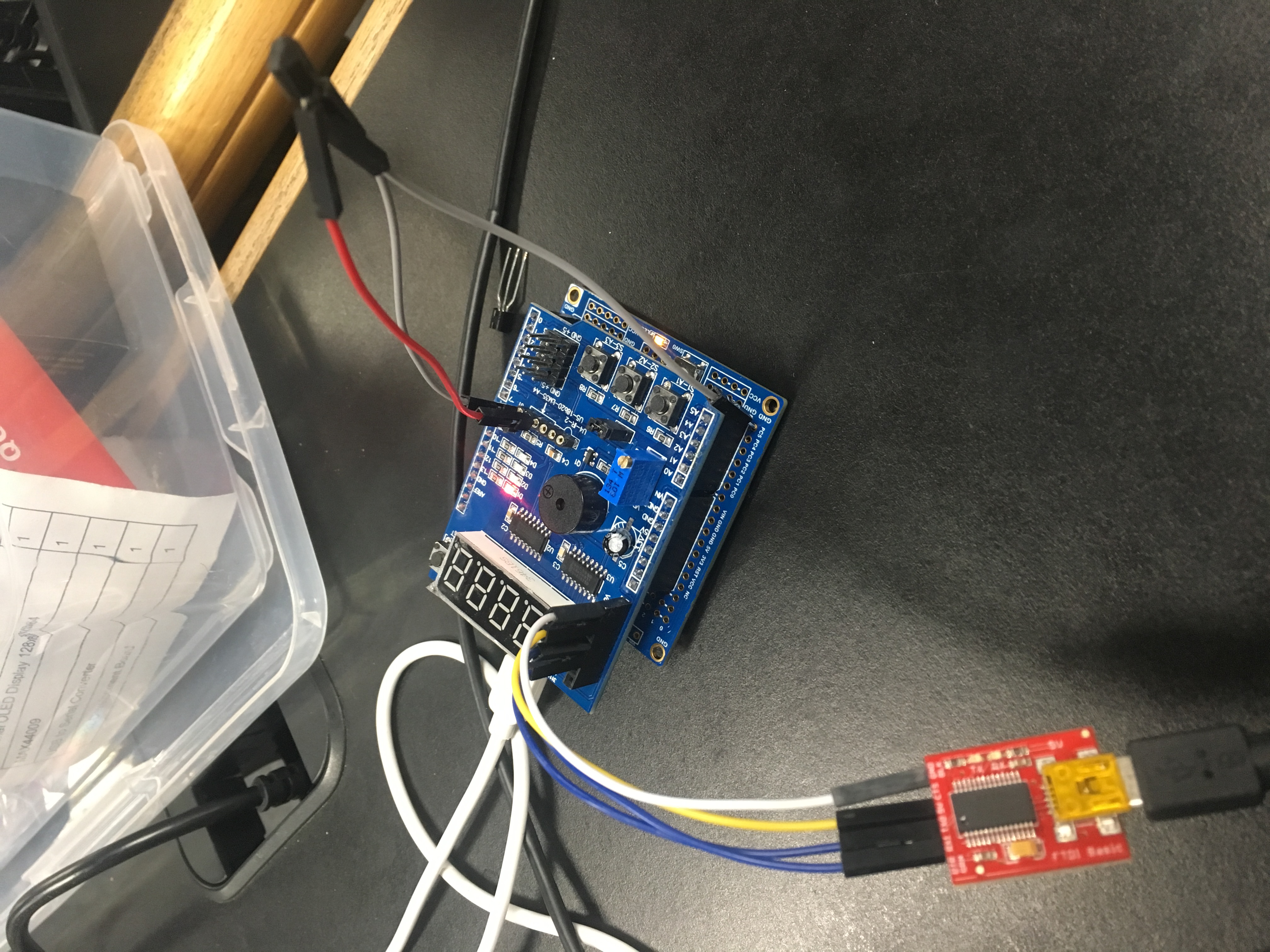


Temperature readings after lighting a lighter under the LM34



Time on the X-Axis, and the temperature reading on Y-Axis

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



Screenshot of the board

The wire on the bottom is sticking to PC5

The wires on the top right is the LM34

The wires on the bottom left is the FTDI connected to the blutooth headers

1. **VIDEO LINKS OF EACH DEMO**

<https://www.youtube.com/watch?v=zUFWhA-y1yY>

1. **GITHUB LINK OF THIS DA**

<https://github.com/BarrChris/submission_da/tree/master/DesignAssignments/DA3B>

**Student Academic Misconduct Policy**

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

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

Chris Barr