CPE301 – SPRING 2019

Design Assignment 3A

Student Name: Mat Tomljenovic

Student #: 1012254710

Student Email: tomljeno@unlv.nevada.edu

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

Directory: https://github.com/matcroatia/DA3A

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

List of components: Internal resistor, Pin C PORT B, Atmega328P

Pins: PB02 and PC02, LED1 and SW1 switch.

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

/\*

\* Assignment 3.c

\*

\* Created: 3/27/2019 6:20:28 PM

\* Author : Mat Tomljenovic

\*/

#define *F\_CPU* 8000000UL

#include <avr/io.h>

#include <util/delay.h>

#include <stdio.h>

#include <avr/interrupt.h>

#define BAUDRATE 9600

#define BAUD\_PRESCALLER (((*F\_CPU* / (BAUDRATE \* 16UL))) - 1)

void USART\_init( unsigned int ubrr ); //function for unsigned integer

void USART\_tx\_string( char \*data ); //function for char data

volatile int Count; //integer variable count

char outs[60]; //number of character to be allowed

int n; //integer declaration

float PI; //floating point number declaration

char str[] = "The Sleeper Has Awakened!"; //string to be displayed

char empty[] = " "; //declaration to print blank space

/\* MAIN FUNCTION \*/

int main(void)

{

Count = 0; //count used for timer

TIMSK0 |= (1 << TOIE0); //interrupt used when timer overflows

sei (); //enable interrupts

TCCR0A = 0x00; //enable normal mode

TCCR0B = 0x05; //set pre-scaler to 1024

USART\_init(BAUD\_PRESCALLER); //initiate baud pre-scaler

USART\_tx\_string("Connected!\r\n"); //displays if serial is connected to output

while(1);

}

/\* INIT USART (RS-232) \*/

void USART\_init( unsigned int ubrr )

{

UBRR0H = (unsigned char)(ubrr>>8); //baud rate register for high byte

UBRR0L = (unsigned char)ubrr; //baud rate register for low byte

UCSR0B = (1 << TXEN0); //Enable RX, TX & RX interrupt

UCSR0C = (3 << UCSZ00); //asynchronous 8 N 1

}

/\* SEND A STRING TO THE RS-232 \*/

void USART\_tx\_string( char \*data )

{

while ((\*data != '\0')) //wait while data transmission is ready

{

while (!(UCSR0A & (1 <<UDRE0)));

UDR0 = \*data; //store data into register UDR0

data++;

}

}

ISR (TIMER0\_OVF\_vect)

{

while (Count < 61) //while loop to count

{

if ((TIFR0 & 0x01) == 1) //sets overflow flag to 1

{

TIFR0 = 0x01;

Count++; //increments count

}

}

if (Count > 60) //while loop to count above 60

{

n = *rand*(); //integer used to generate random number

PI = 3.1415; //declaration for floating point decimal

*dtostrf*(PI, 6, 6, outs); //print PI and declare how many decimal points print

USART\_tx\_string(outs); //prints declared string

USART\_tx\_string(empty); //print a blank space

*snprintf*(outs, sizeof(outs), "%3d\r\n" , n); //converts random number to integer

USART\_tx\_string(outs); //prints out PI

USART\_tx\_string(empty); //prints out blank space

USART\_tx\_string(str); //prints out random number

USART\_tx\_string(empty); //prints out blank space

Count = 0; //resets counter back to 0

}

}

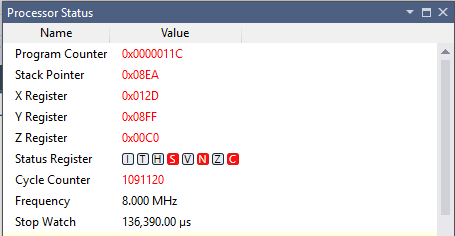
1. **DEVELOPED MODIFIED CODE OF TASK 2/A from TASK 1/A**

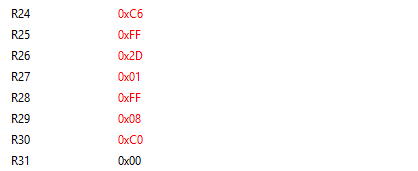
Same as above.

1. **SCHEMATICS**

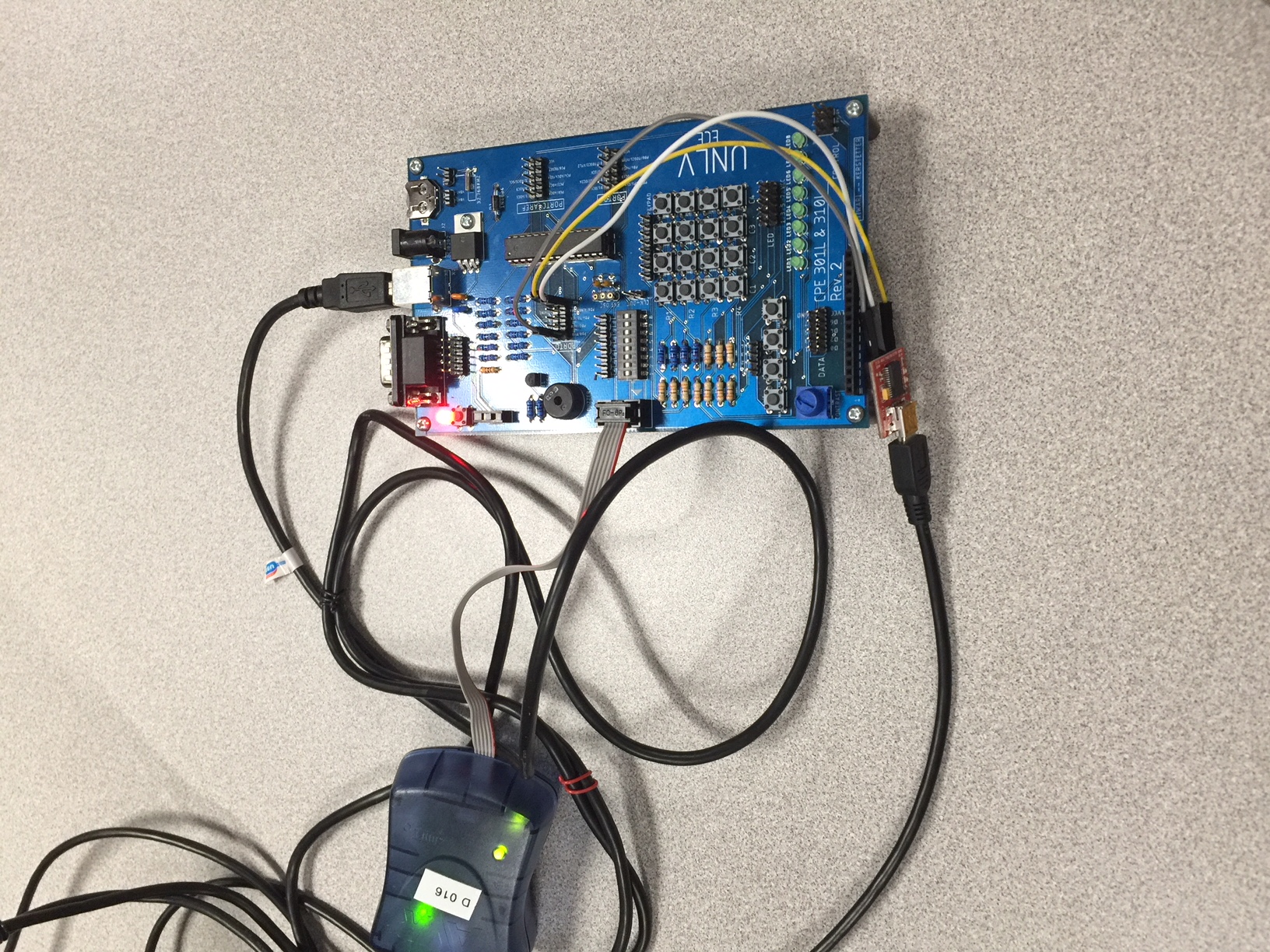
Use fritzing.org

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





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



1. **VIDEO LINKS OF EACH DEMO**

<https://youtu.be/miiUzg9G94Y>

1. **GITHUB LINK OF THIS DA**

https://github.com/matcroatia/DA3A

**Student Academic Misconduct Policy**

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

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

NAME OF THE STUDENT