CPE301 – FALL 2019

Design Assignment 3A

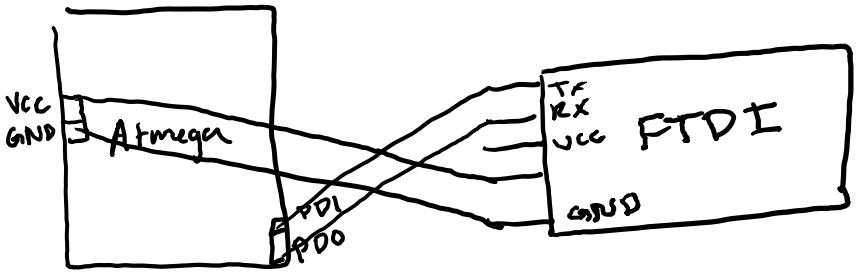
Student Name: Alira Coffman
Student #: 5003236350
Student Email: coffma2@unlv.nevada.edu
Primary Github address: https://github.com/Alira-Coffman
Directory: https://github.com/Alira-Coffman/submission-repo/tree/master/ESD301/DA/DA3A

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

I used the atmega328p board, FTDI chip, and bread board.



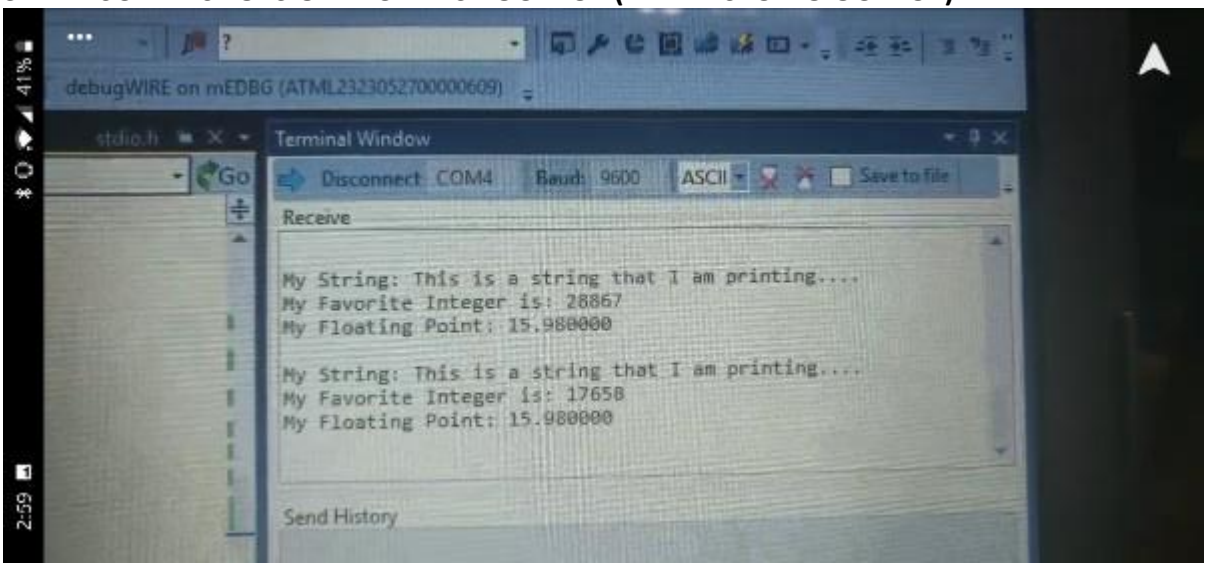
2. INITIAL/MODIFIED/DEVELOPED CODE OF TASK 1/A

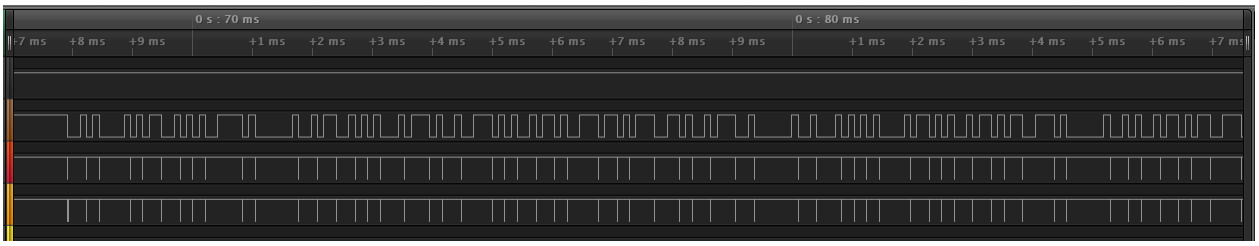
```
/*UART TestAtmega328 DIP TX PD1 (pin3)Atmega328 DIP RX PD0 (pin2) */
#define BAUD 9600
#define F_CPU 16000000UL
#include <time.h>
#include <avr/io.h>
#include <stdio.h>
#include <util/delay.h>
#include <util/setbaud.h>
#include <avr/interrupt.h>
#include <stdlib.h>

//srand(time(NULL));


void USART_send(char data);
void USART_putstring(char* StringPtr);
void USART_init(void);
int randomValue;
char stringtype[] = "My String: ";
char inttype[] = "My Favorite Integer is: ";
char floattype[] = "My Floating Point: ";
char Space[] = "\n";

char String[] = "This is a string that I am printing....";
char floating[15];
volatile float my_float = 1006.3452; // Sets the float value

int main(void)
{
	USART_init(); // Initializes the analog to digital functions as well as
OVF interrupt
	srand(time(NULL));   // Initialization, should only be called once.
```

```
	int r = rand();		// Returns a pseudo-random integer between 0 and
RAND_MAX.
	char arr[10] = "";
	randomValue = r;

	while(1)
	{
		// main loop
	}
}

ISR (TIMER1_OVF_vect)
{
	USART_putstring(Space);
	USART_putstring(stringtype);
	USART_putstring(String);
	USART_putstring(Space);
	USART_putstring(inttype);
	USART_send(randomValue);
	USART_putstring(Space);
	USART_putstring(floattype);
	snprintf(floating, sizeof(floating), "%f\r\n", my_float);
	USART_putstring(floating);
	USART_putstring(Space);
	TCNT1 = 49911;
}

void USART_init( void )
{
	UBRR0H = 0;
	UBRR0L = F_CPU/16/BAUD - 1;
	UCSR0C = _BV(UCSZ01) | _BV(UCSZ00);
	UCSR0B = _BV(RXEN0) | _BV(TXEN0);
	TCCR1B |= 5;
	TIMSK1 = (1 << TOIE1);
	TCNT1 = 49911;
	sei();
}

void USART_send(char data)
{
	while (!(UCSR0A & (1 << UDRE0)));
	UDR0 = data; // UDR0 register grabs the value given from the parameter

}

void USART_putstring(char *StringPtr)
{
	while ((*StringPtr != '\0')){ // Until it reaches the end of the line, it
will keep looping
		while (!(UCSR0A & (1 << UDRE0))); // Until UDRE0 goes high, it will
keep looping
```

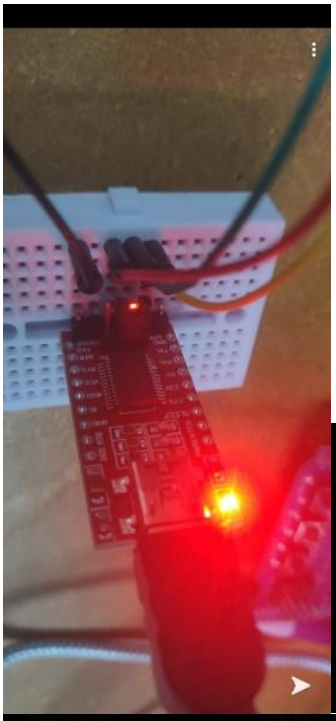
```
            UDR0 = *StringPtr; // UDR0 register grabs the value given from the
parameter
            StringPtr++; // but it does it by every character as shown here
      }
}
```

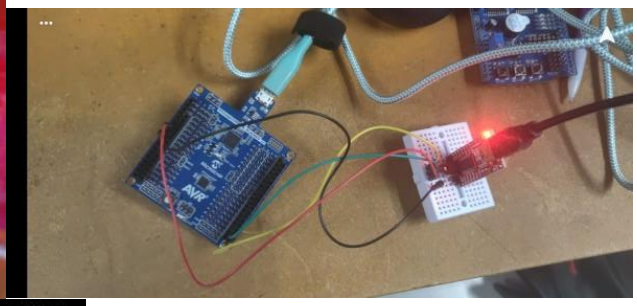
3. SCREENSHOTS OF EACH TASK OUTPUT (ATMEL STUDIO OUTPUT)

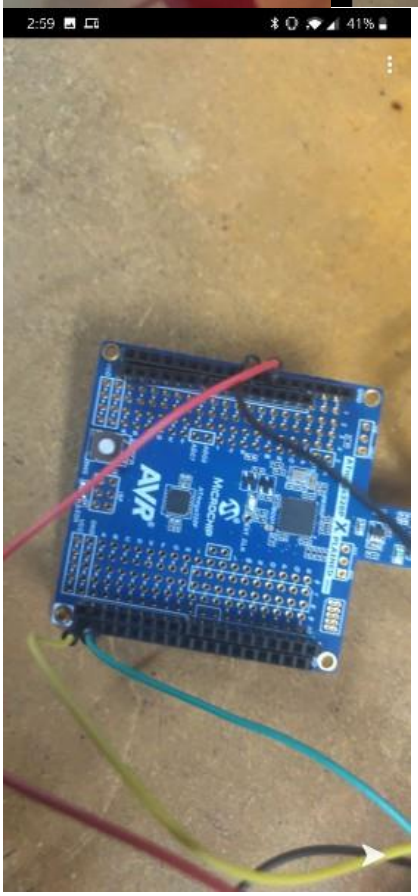




4. SCREENSHOT OF EACH DEMO (BOARD SETUP)







5. VIDEO LINKS OF EACH DEMO

https://youtu.be/gplkwuhdv78

6. GITHUB LINK OF THIS DA

https://github.com/Alira-Coffman/submission-repo/tree/master/ESD301/DA/DA3A

Student Academic Misconduct Policy
http://studentconduct.unlv.edu/misconduct/policy.html