CPE301 – SPRING 2019

Design Assignment 3A

Student Name: Allis Hierholzer
Student #: 2000160356
Student Email: hierholz@unlv.nevada.edu
Primary Github address: https://github.com/acexhp/submission_da.git
Directory: Repository/cpe301/DesignAssignment/DA3A

Task:

The goal of the assignment is to modify the above codes to do the following:

1. Write a C AVR program that will display a string, random integer and floating point values on the serial terminal every 1 sec. Use a timer with interrupt for the 1 sec delay. Use a FTDI chip for serial to USB conversion.

Submission:

The following are required for successful completion of the design assignment:

a. AVR C code that has been compiled and working.

b. The C code should be well documented with explanation of every instruction.

c. A word document that contains the flow chart of the assembly code along with the snapshots of the schematics, components connected on the breadboard and screen shoots.

1. COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS

- Breadboard
- Wires
- USB Cables
- ATMEGA328P XPLAINED MINI
- ATMEL STUDIO 7.0

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

```
/*
 * DA3A.c
 */

#include <avr/io.h>
#include <util/delay.h>
#include <avr/interrupt.h>
#include <stdio.h>

#define F_CPU 16000000UL
#define BAUDRATE 9600
#define BAUD_PRESCALLER (((F_CPU / (BAUDRATE * 16UL)))-1)

//Function declarations
volatile int Count;
void USART_init( unsigned int ubrr );                    //calls integere
void USART_TX_string(char *data);                        //calls string
char outs[30];
int random_num;                                          //creates variables
float AVOGADRO_NUM;
char string[] = "x 10^23 is Avogadro's number :)";       //creates string
char space[] = " ";                                      //creates space

int main(void)
{
       Count = 0;
       TIMSK0 |= (1<<TOIE0);                             //sets interrupt when overflow
                                                         occurs
       sei ();
       TCCR0A = 0;                                       //normal mode
       TCCR0B |= (1<<CS02)|(1<<CS00);                    //prescaler = 1024
       USART_init(BAUD_PRESCALLER);                      //baud prescaler
       USART_TX_string("Printing...\r\n");               //shows succesful connection
    while (1);
}
//int USART (RS-232)
void USART_init( unsigned int ubrr ){
       UBRR0H = (unsigned char)(ubrr>>8);
       UBRR0L = (unsigned char)ubrr;
       UCSR0B = (1 << TXEN0);
       UCSR0C = (3 << UCSZ00);
}
//send string to RS-232
void USART_TX_string(char *data) {
       while (*data != '\0') {
              while (!(UCSR0A & (1<<UDRE0)));
              UDR0 = *data;
```

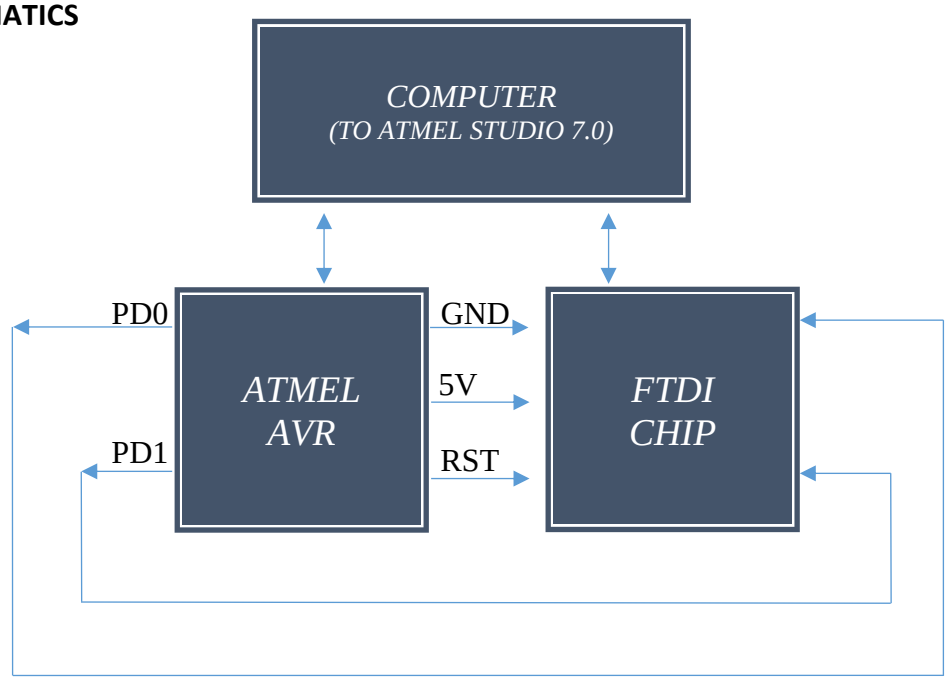
```
                data++;
        }
}
ISR (TIMER0_OVF_vect){                                  //timer0 overlfow interrupt
        while (Count < 61){                             //waiting
                if ((TIFR0 & 0x01) == 1){               //detects overflow
                        TIFR0 = 0X01;                   //reset overlfow flag
                        Count++;
                }
        }
        if (Count > 60){
                USART_TX_string(string);                //prints string
                USART_TX_string(space);                 //prints space

                random_num = rand();                    //creates random number
                AVOGADRO_NUM = 6.022141;                //creates and store float value

                snprintf(outs, sizeof(outs), "%3d\r\n", random_num);
                USART_TX_string(outs);
                USART_TX_string(space);

                sprintf(outs, "%f", AVOGADRO_NUM);
                USART_TX_string(outs);
                USART_TX_string(space);
                Count = 0;
        }
}
```

3. SCHEMATICS



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

main | int main(void)

```
/*
 * DA3A.c
 */

#define F_CPU 16000000UL
#include <avr/io.h>
#include <util/delay.h>
#include <avr/interrupt.h>
#include <stdio.h>

#define BAUDRATE 9600
#define BAUD_PRESCALLER (((F_CPU / (BAUDRATE * 16UL)))-1)

//Function declarations
volatile int Count;
void USART_init( unsigned int ubrr );
void USART_TX_string(char *data);
char outs[30];
int random_num;                                 //creates variable
float AVOGADRO_NUM;                             //creates variable
char str[] = "x 10^23 is Avogadro's number :)"; //creates string
char empty[] = " ";                             //creates space

int main(void)
{
    Count = 0;
    TIMSK0 |= (1<<TOIE0);                       //sets interrupt when
    sei ();
    TCCR0A = 0;                                 //normal mode
    TCCR0B |= (1<<CS02)|(1<<CS00);              //prescaler = 1024
    USART_init(BAUD_PRESCALLER);                //baud prescaler
    USART_TX_string("Printing...\r\n");         //shows succesful con
    while (1);
}
//int USART (RS-232)
void USART_init( unsigned int ubrr ){
    UBRR0H = (unsigned char)(ubrr>>8);
    UBRR0L = (unsigned char)ubrr;
    UCSR0B = (1 << TXEN0);
    UCSR0C = (3 << UCSZ00);
}
//send string to TEH RS-232
void USART_TX_string(char *data) {
    while (*data != '\0') {
        while (!(UCSR0A & (1<<UDRE0)));
        UDR0 = *data;
        data++;
    }
}
ISR (TIMER0_OVF_vect){                          //timer0 overlfow interrupt
    while (Count < 61){                         //waiting
        if ((TIFR0 & 0x01) == 1){               //detects overflow
            TIFR0 = 0X01;                       //reset overlfow flag
```

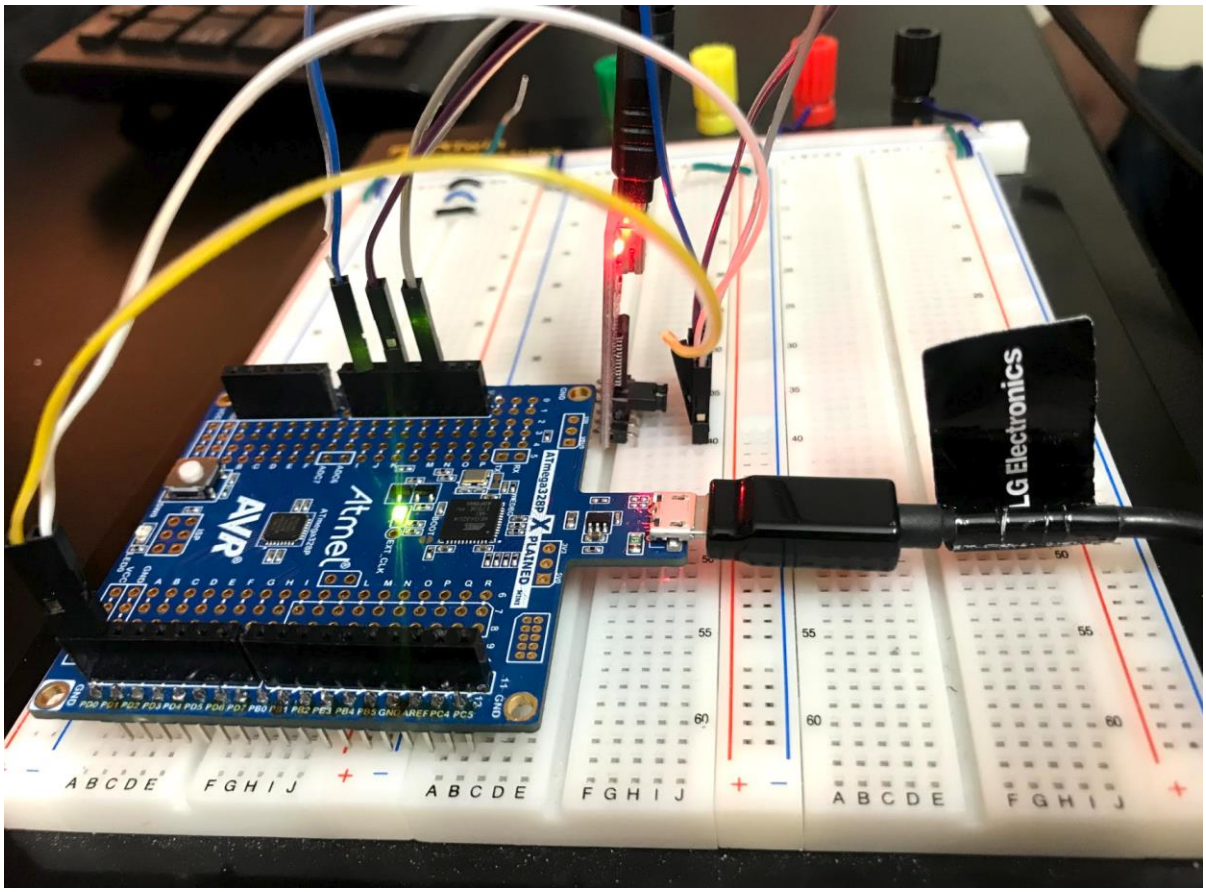
I/O

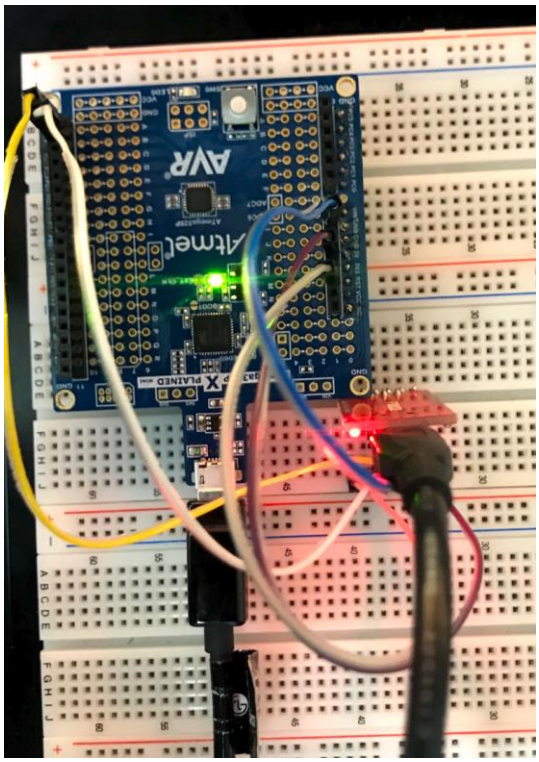
Filter:

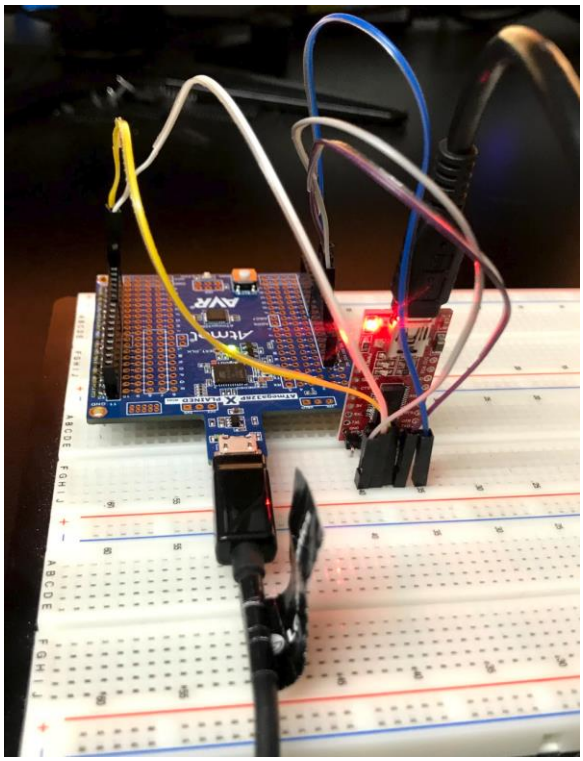
Name	Value
I/O Port (PORTB)	
I/O Port (PORTC)	
I/O Port (PORTD)	
Serial Peripheral Interface (...	
Timer/Counter, 16-bit (TC1)	
Timer/Counter, 8-bit (TC0)	
Timer/Counter, 8-bit Async...	
Two Wire Serial Interface (...	
USART (USART0)	
Watchdog Timer (WDT)	

Name	Address	Value	Bits
UCSR0A	0xC0	0x20	
UCSR0B	0xC1	0x08	
UCSR0C	0xC2	0x06	
UBRR0	0xC4	0x0067	
UDR0	0xC6	0x00	

5. SCREENSHOT OF EACH DEMO (BOARD SETUP)







6. VIDEO LINKS OF EACH DEMO

https://youtu.be/de81e_pFMLY

7. GITHUB LINK OF THIS DA

https://github.com/acexhp/submission_da.git

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

"*This assignment submission is my own, original work*".
Allis Hierholzer