CPE301 – SPRING 2019

Design Assignment DA3B

Student Name: Saul Alejandro Mendoza Guzman

Student #: 2000540481

Student Email: mendos1@unlv.nevada.edu

Primary Github address: <https://github.com/mendos1>

Directory: subnission\_da/DA3B

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**

Atmel Studio 7, ATmega328p Xplained mini, FTDI chip, breadboard, two wires, two usb cables, LM34, LIGHTER.

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

#define F\_CPU 16000000UL //16MHz

#define BAUD 9600 //Baud Rate

#define MYUBRR F\_CPU/16/BAUD-1 //calculate Baud

#include <avr/io.h>

#include <util/delay.h>

#include <stdio.h>

#include <avr/interrupt.h>

void READ\_ADC(void); //Read ADC

void ADC\_INIT(void); //initialize ADC

void USART\_INIT( unsigned int ubrr ); //initialize USART

void USART\_TX\_STRING(char \*data); //Print String USART

volatile unsigned int ADC\_TEMP;

char outs[20]; //array

int main(void) {

ADC\_INIT(); // Initialize the ADC

USART\_INIT(MYUBRR); // Initialize the USART

USART\_TX\_STRING("Connected!\r\n"); // connection with USART

\_delay\_ms(125); // wait a bit

sei(); //interrupt

while(1)

{

}

}

void USART\_TX\_STRING( char \*data ) {

while ((\*data != '\0')) {

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

UDR0 = \*data;

data++;

}

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

void USART\_INIT( unsigned int ubrr ) {

UBRR0H = (unsigned char)(ubrr>>8);

UBRR0L = (unsigned char)ubrr;

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

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

}

void ADC\_INIT(void)

{

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)| //

(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);

// Timer/Counter1 Interrupt Mask Register

TIMSK1 |= (1<<TOIE1); // enable overflow interrupt

TCCR1B |= (1<<CS12)|(1<<CS10); // clock

TCNT1 = 49911; //((16MHz/1024)\*1)-1 = 15624

}

}

ISR(TIMER1\_OVF\_vect) //timer overflow interrupt to delay for 1 second

{

READ\_ADC();//read ADC

snprintf(outs,sizeof(outs),"%3d F \r\n", ADC\_TEMP);// print it

USART\_TX\_STRING(outs);

TCNT1 = 49911; //reset tcnt

}

void READ\_ADC(void) {

unsigned char i =4;

ADC\_TEMP = 0; //initialize

while (i--) {

ADCSRA |= (1<<ADSC);

while(ADCSRA & (1<<ADSC));

ADC\_TEMP+= ADC;

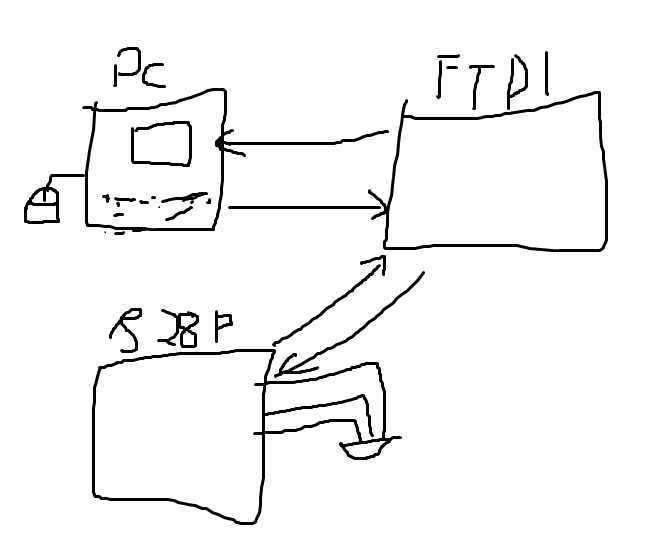
\_delay\_ms(50);

}

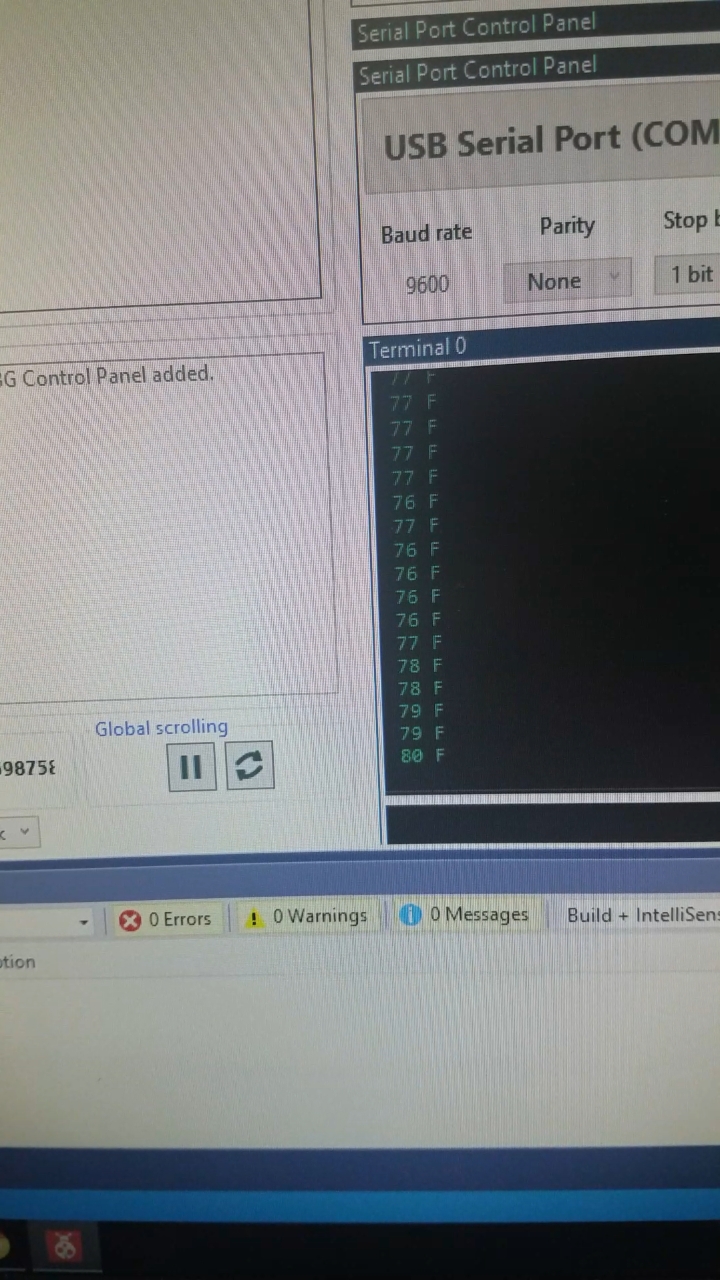
ADC\_TEMP = ADC\_TEMP/8 ; // Average

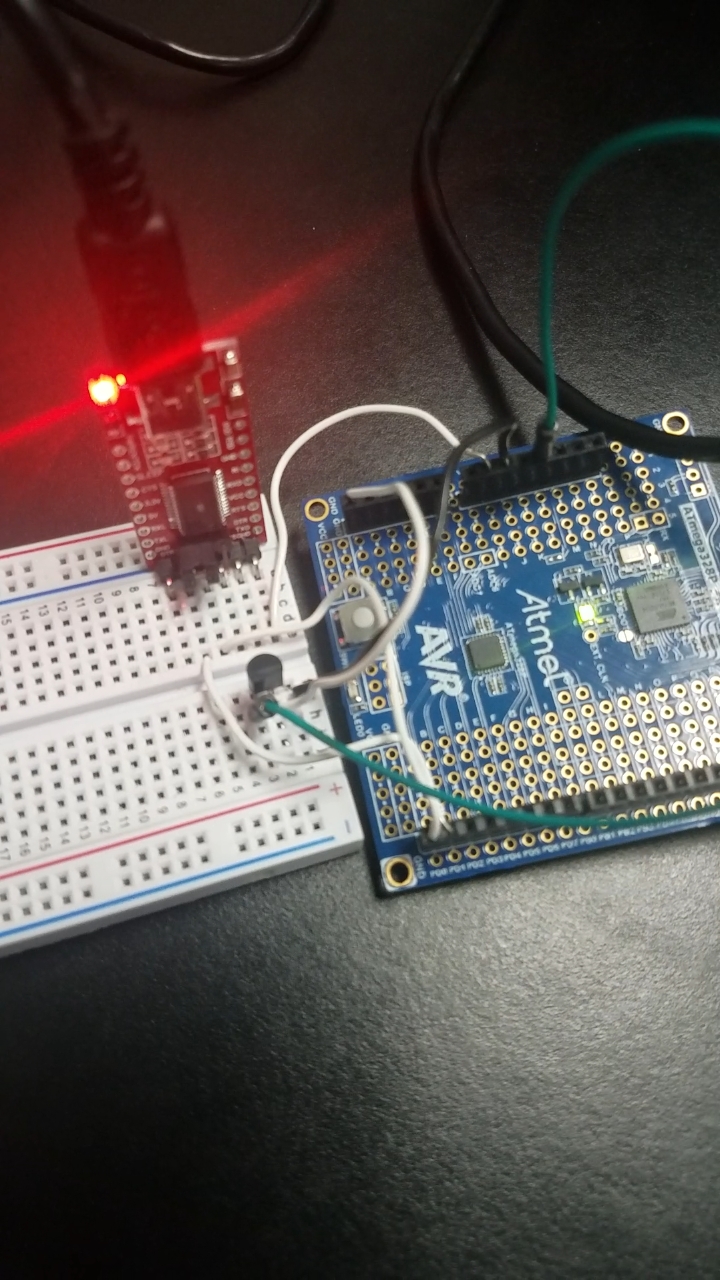
}

1. **schematic**

****

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





1. **VIDEO LINKS OF EACH DEMO**

<https://www.youtube.com/watch?v=xRrjuo9wgkI>

1. **GITHUB LINK OF THIS DA**

<https://github.com/mendos1/subnission_da>

**Student Academic Misconduct Policy**

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

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

NAME OF THE STUDENT