CPE301 – SPRING 2019

Design Assignment 3A

Student Name: Robert Sander

Student #: 5002102412

Student Email: sander1@unlv.nevada.edu

Primary Github address: <https://github.com/sanderUNLV/submission_DA.git>

Youtube link: <https://youtu.be/Ts_ygjckxOw>

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

// DA3A\_T.1\_C

// Author : Robert Sander

#define *F\_CPU* 16000000UL

#include <avr/io.h>

#include <stdio.h>

#include <avr/interrupt.h>

#define BAUDRATE 9600

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

//Declaration of our functions

void USART\_init(void);

void USART\_send(unsigned char data);

// String

void USART\_putstring(char \*StringPtr);

char String[] = "Hello! "; //String[] is in fact an array but when we put the text between the " " symbols the compiler threats it as a String and automatically puts the null termination character in the end of the text

// Random Integer

void USART\_putrandomint(char \*randint);

volatile int randint = 8;

char randintout[2];

// Floating point number

void USART\_putfloating(char \*floatingnumber);

volatile double floatingnumber = 0.123;

char fltouts[20];

// Space

void USART\_putspace(char \*SpacePtr);

char Space[] = " ";

int main(void) {

OCR1A = 62500; // 16000000/256=62500 - we set the new TOP number to 62500 for 1s delay

TCCR1B |= (1<<CS12); //int\_clk internal clk, 256 prescaler

TCCR1A |= (1<<WGM12); //CTC mode TOP=OCR1A

TIMSK1 = (1<<OCIE1A); //enable Timer1 compare match int.

sei (); //enable interrupts

USART\_init(); //Call the USART initialization code

while (1)

{

//Infinite loop

}

}

void USART\_init(void) {

UBRR0H = (*uint8\_t*)(BAUD\_PRESCALLER >> 8);

UBRR0L = (*uint8\_t*)(BAUD\_PRESCALLER);

UCSR0B = (1 << TXEN0);

UCSR0C = (3 << UCSZ00);

}

void USART\_send(unsigned char data) {

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

UDR0 = data;

}

void USART\_putstring(char\* StringPtr)

{

while (\*StringPtr != 0x00) {

USART\_send(\*StringPtr);

StringPtr++;

}

}

void USART\_putrandomint(char \*randint)

{

while (\*randint != 0x00) {

USART\_send(\*randint);

randint++;

}

}

void USART\_putfloating(char \*floatingnumber)

{

while (\*floatingnumber != 0x00) {

USART\_send(\*floatingnumber);

floatingnumber++;

}

}

void USART\_putspace(char\* SpacePtr)

{

while (\*SpacePtr != 0x00) {

USART\_send(\*SpacePtr);

SpacePtr++;

}

}

ISR (TIMER1\_COMPA\_vect) //ISR for Timer0 compare match

{

USART\_putstring(String); //Pass the string to the USART\_putstring function and sends it over the serial

*snprintf*(randintout,sizeof(randintout),"%d\r\n", randint); // print random integer

USART\_putrandomint(randintout);

USART\_putstring(Space);

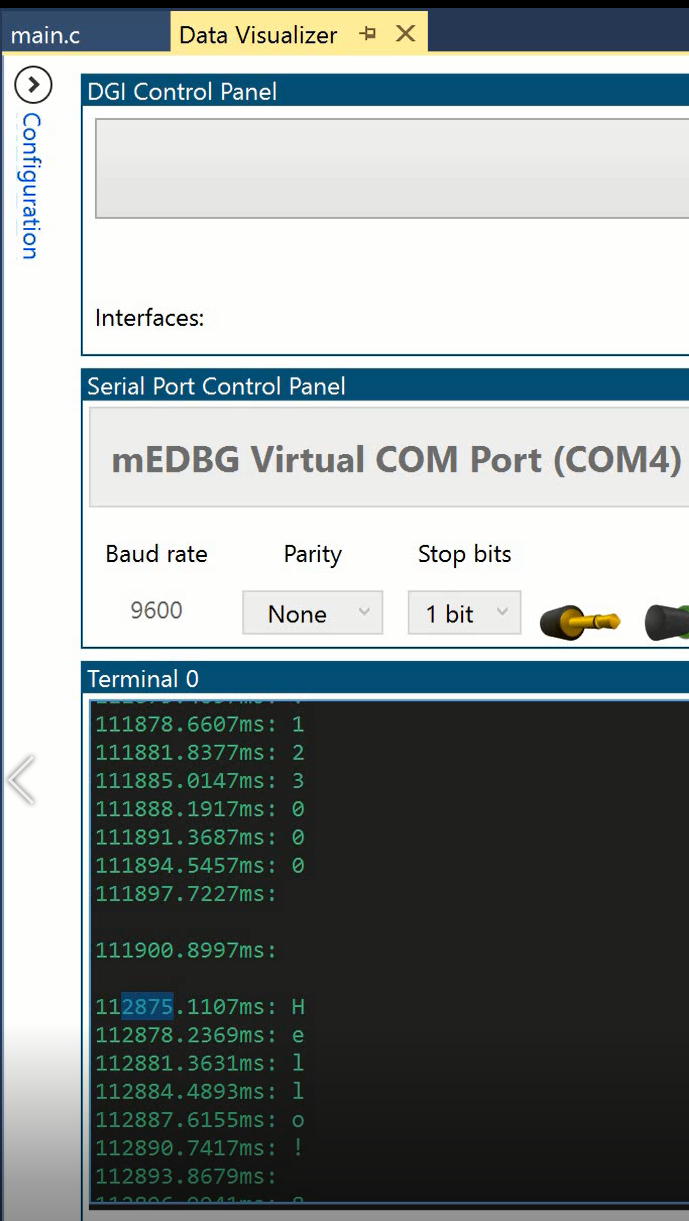
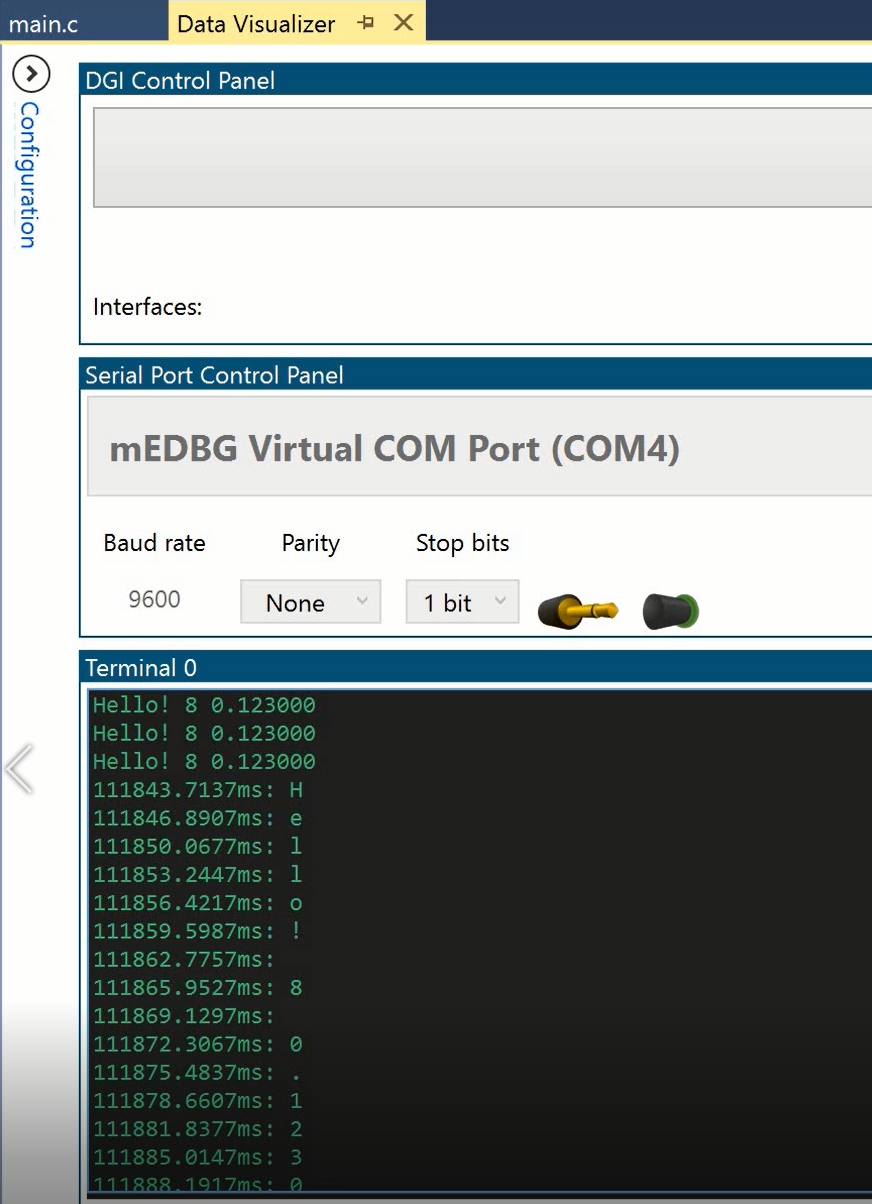
*snprintf*(fltouts,sizeof(fltouts),"%f\r\n", floatingnumber); // print floating point number

USART\_putfloating(fltouts);

TIFR0 |= (1<<OCF0A); // clear the compare A match flag

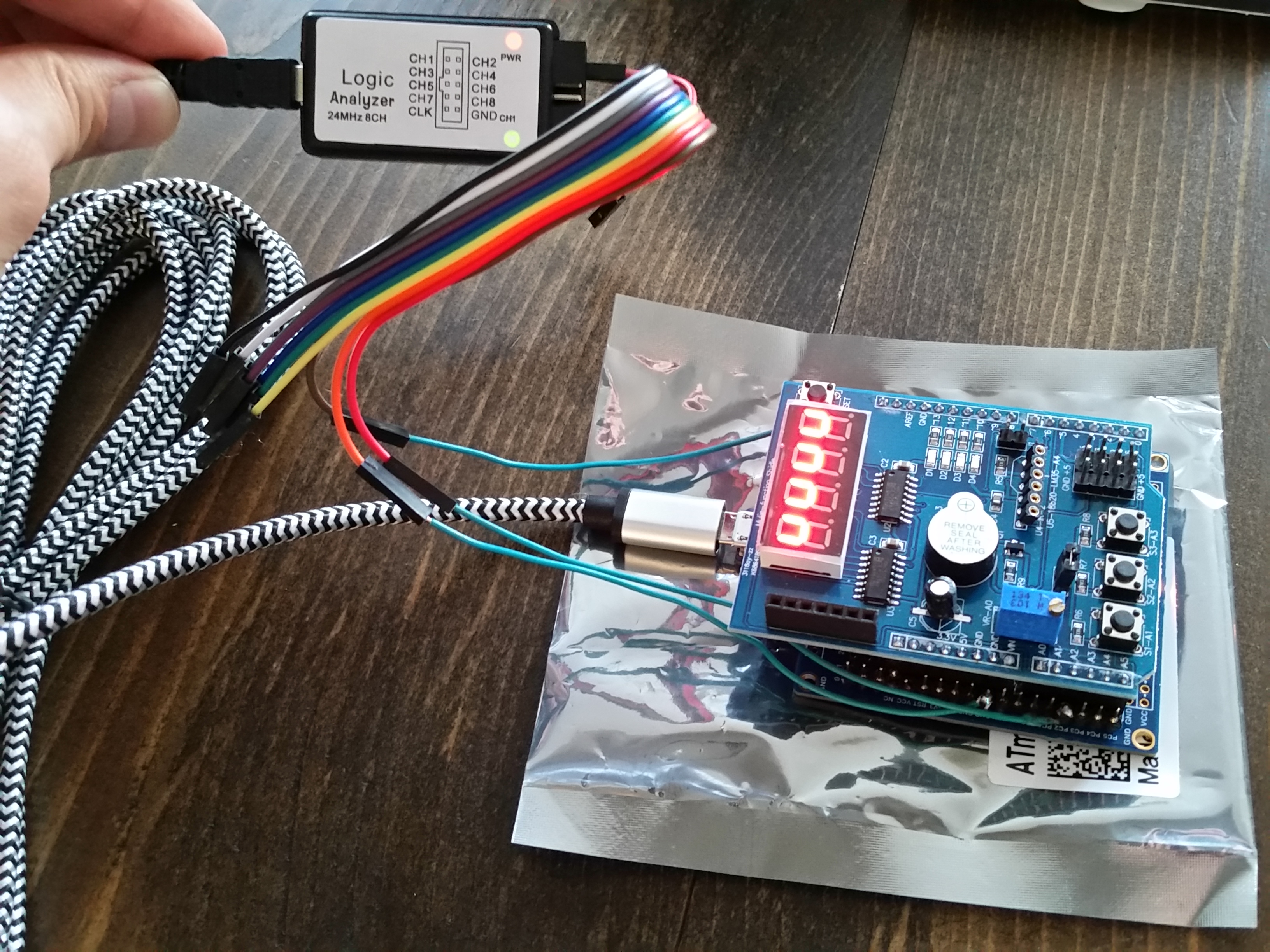
}

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



*Figure 1 – Data Visualizer Output, 1000ms delay*

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



1. **VIDEO LINKS OF EACH DEMO**

<https://youtu.be/Ts_ygjckxOw>

1. **GITHUB LINK OF THIS DA**

<https://github.com/sanderUNLV/submission_DA.git>

**Student Academic Misconduct Policy**

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

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

-Robert Sander