CPE301 – SPRING 2019

Design Assignment 3A

Student Name: Geovanni Portillo

Student #: 8000603824

Student Email: [portig1@unlv.nevada.edu](mailto:portig1@unlv.nevada.edu)

Primary Github address: <https://github.com/portig1/submissions_E>

Directory: submissions\_E/DA/LAB3A/

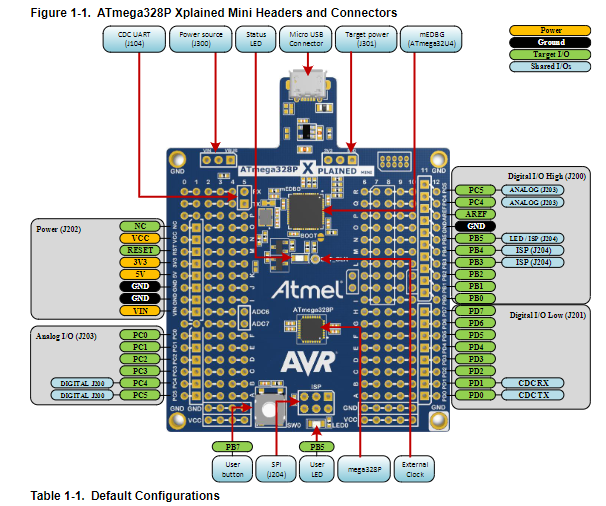
Submit the following for all Labs:

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

ATmega328PB Xplained mini



1. **INITIAL CODE OF TASK 1**

#define *F\_CPU* 16000000UL

#include <avr/io.h>

#include <util/delay.h>

#define BAUDRATE 9600

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

//Declaration of our functions

void USART\_init(void);

unsigned char USART\_receive(void);

void USART\_send( unsigned char data);

void USART\_putstring(char\* StringPtr);

char String[] = "Hello world!!"; //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

int main(void) {

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

while (1) { //Infinite loop

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

*\_delay\_ms*(5000); //Delay for 5 seconds so it will re-send the string every 5 seconds

}

return 0;

}

void USART\_init(void) {

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

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

UCSR0B = (1 << RXEN0) | (1 << TXEN0);

UCSR0C = (3 << UCSZ00);

}

unsigned char USART\_receive(void) {

while (!(UCSR0A & (1 << RXC0)));

return UDR0;

}

void USART\_send( unsigned char data) {

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

UDR0 = data;

}

void USART\_putstring(char\* StringPtr) {

while (\*StringPtr != 0x00) {

USART\_send(\*StringPtr);

StringPtr++;

}

}

1. **MODIFIED CODE OF TASK1**

#include <stdio.h> //included for snprintf

#include <stdlib.h> //included for rand

#define *F\_CPU* 16000000UL

#include <avr/io.h>

#include <avr/interrupt.h>

#define BAUDRATE 9600

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

//Declaration of our functions

void USART\_init(void);

unsigned char USART\_receive(void);

void USART\_send( unsigned char data);

void USART\_putstring(char\* StringPtr);

//Declaration of global variables

volatile float randomFlt;

volatile int randomInt;

char String1[] = "The random integer is "; //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

char String2[] = ", and the random float is ";

char randomIntString[20];

char randomFltString[20];

int main(void) {

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

TIMSK1 = (1 << OCIE1A);

sei();

OCR1A = 62499; //Using TCNT = clk\*delay/prescaler - 1 to find OCR1A given clk = 16MHz, OCR1A was calculated to 62,499

TCCR1A = 0; // COM1A/B Normal Operation, OC1A/B Disconnected

TCCR1B = (1 << WGM12) | (1 << CS12); //WGM CTC Mode, Prescaler = 256

while (1) { //Infinite loop

}

return 0;

}

void USART\_init(void) {

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

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

UCSR0B = (1 << RXEN0) | (1 << TXEN0);

UCSR0C = (3 << UCSZ00);

}

unsigned char USART\_receive(void) {

while (!(UCSR0A & (1 << RXC0)));

return UDR0;

}

void USART\_send( unsigned char data) {

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

UDR0 = data;

}

void USART\_putstring(char\* StringPtr) {

while (\*StringPtr != 0x00) {

USART\_send(\*StringPtr);

StringPtr++;

}

}

ISR (TIMER1\_COMPA\_vect) // timer1 compare interrupt

{

randomInt = *rand*();

randomFlt = (randomInt / 13); //Generate random integer then divide by 13 to get a different floating point number

*snprintf*(randomIntString, sizeof(randomIntString), "%d", randomInt);

*snprintf*(randomFltString, sizeof(randomFltString), "%f\r\n", randomFlt); //convert numbers to strings

USART\_putstring(String1); //Pass the string to the USART\_putstring function and sends it over the serial

USART\_putstring(randomIntString);

USART\_putstring(String2);

USART\_putstring(randomFltString);

}

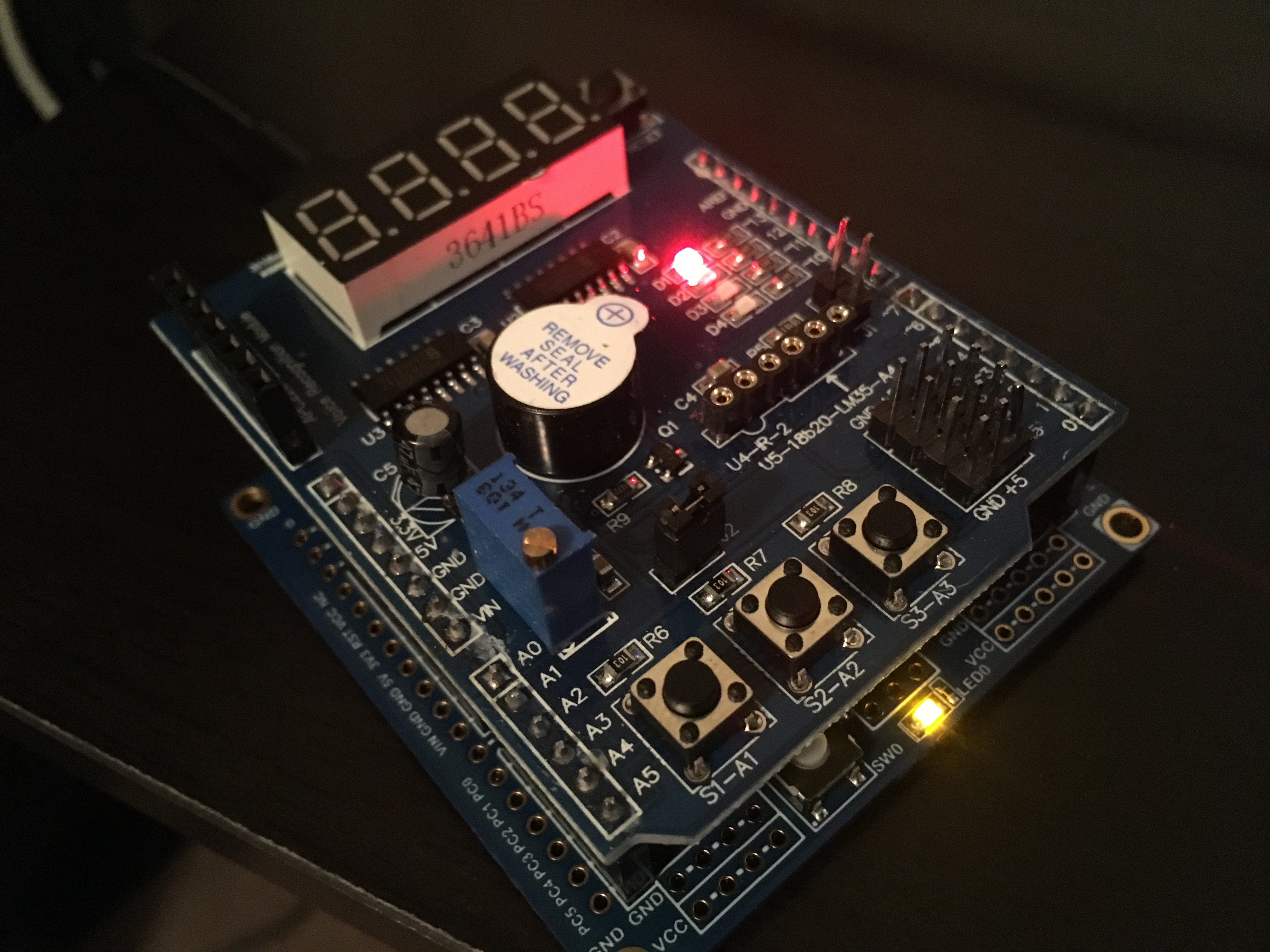
1. **SCHEMATICS**

No schematic needed, no external components used

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

Not asked for in instructions

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

****

Board setup for Task 1

1. **VIDEO LINKS OF EACH DEMO**

<https://youtu.be/0QmDsDwE4xA>

1. **GITHUB LINK OF THIS DA**

https://github.com/portig1/submissions\_E/tree/master/DA/LAB3A

**Student Academic Misconduct Policy**

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

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

Geovanni Portillo