CPE301 – FALL 2019

Design Assignment 5

Student Name: Worku Tafara
Student #: 2001245644
Student Email: tafarw1@unlv.nevada.edu
Primary Github address: https://github.com/WorkuT1226/CPE301.git
Directory:

Submit the following for all Labs:

- In the document, for each task submit the modified or included code (only) with highlights and justifications of the modifications. Also, include the comments.

- 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.

- If multiple asm/c files or other libraries are used, create a folder LabXX-TYY and place these files inside the folder.

- 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).

- **COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS**

  Atmega328p
  NRF24L01
  LM35

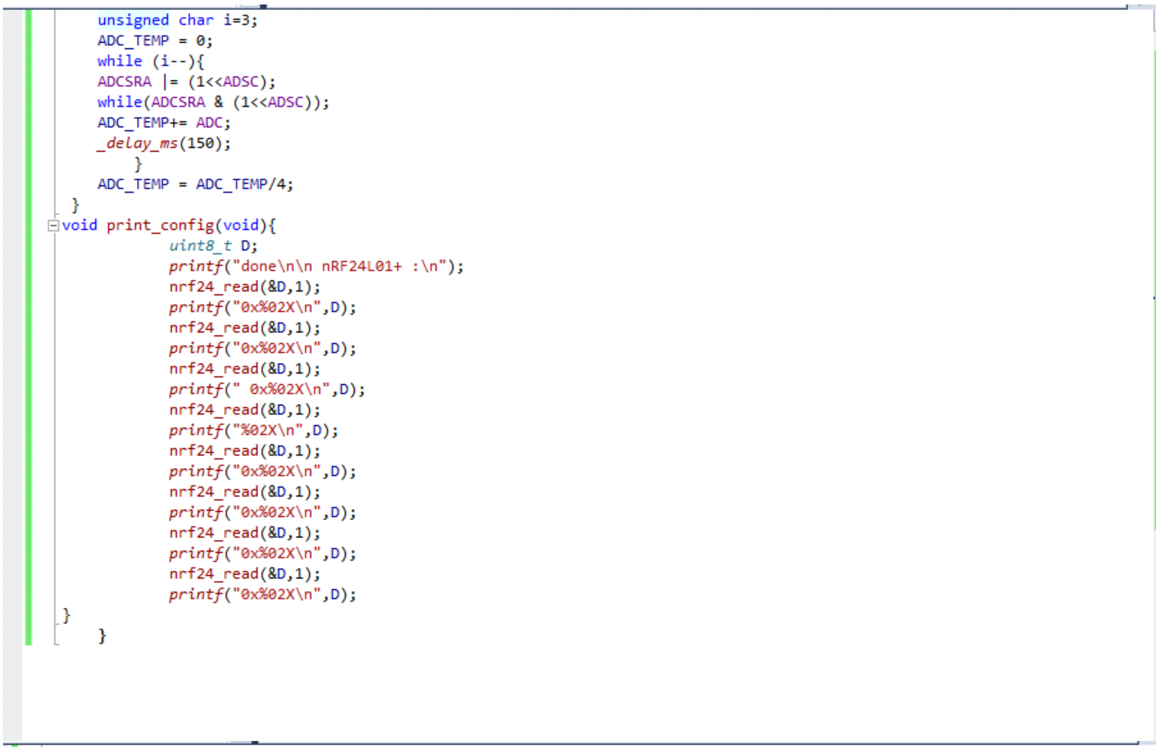
- **INITIAL/MODIFIED/DEVELOPED CODE OF TASK 1/A**

```
#define BAUD 9600
#define F_CPU 16000000UL
#include <avr/io.h>
#include <util/delay.h>
#include <avr/interrupt.h>
#include <stdbool.h>
#include <stdio.h>
#include <string.h>
unsigned int ADC_TEMP;
//#include "nrf24l01.h"
//#include "nrf24l01-mnemonics.h"
//#include "spi.h"
void print_config(void);
void ADC_INIT(void);
void READ_ADC(void);
volatile bool message_received = false;
volatile bool status = false;
int main(void){
	char tx_message[32];
```

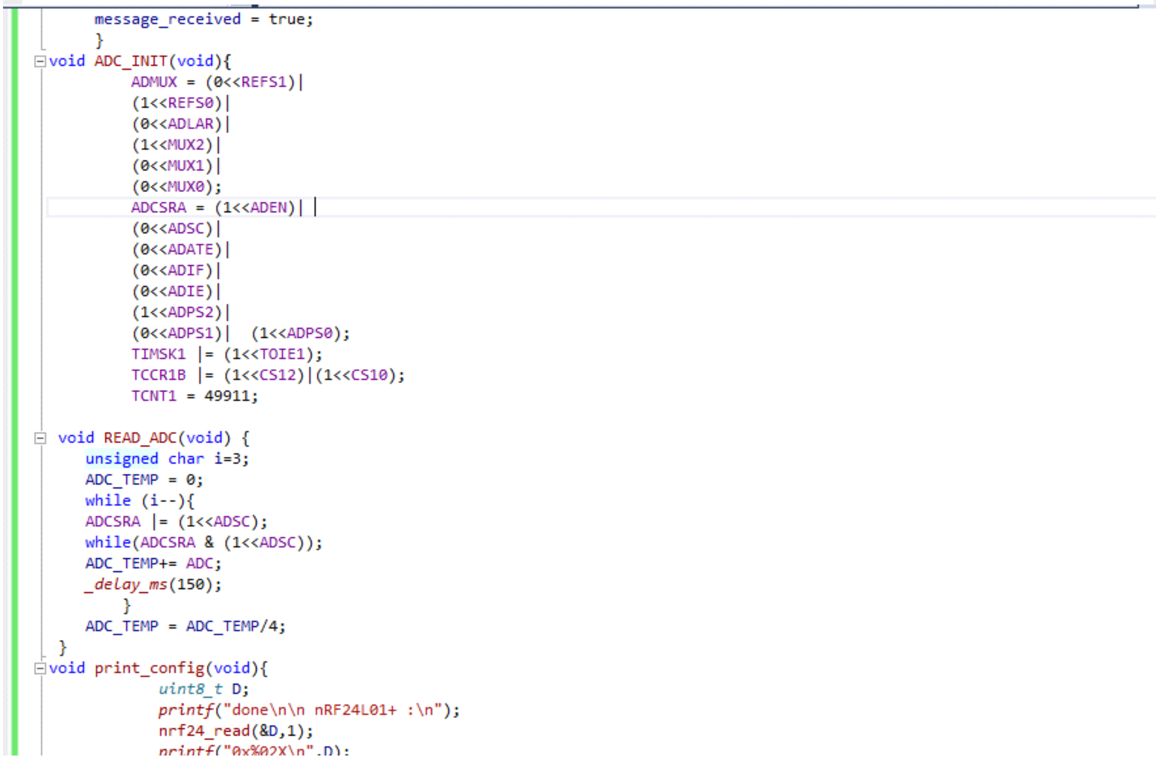
```
       char *tx_ptr = tx_message;
       ADC_INIT();
       printf("begin");
       printf("end");
       ADC_TEMP = 0;
       while (1){
              tx_ptr = tx_message;
       if (message_received){
              printf("hello");
              message_received = false;
              _delay_ms(500);
              if (status == true)
              printf("sent");
               }
       }
}
ISR(INT0_vect){
        message_received = true;
        }
void ADC_INIT(void){
               ADMUX = (0<<REFS1)|
               (1<<REFS0)|
               (0<<ADLAR)|
               (1<<MUX2)|
               (0<<MUX1)|
               (0<<MUX0);
               ADCSRA = (1<<ADEN)|
               (0<<ADSC)|
               (0<<ADATE)|
               (0<<ADIF)|
               (0<<ADIE)|
               (1<<ADPS2)|
               (0<<ADPS1)|  (1<<ADPS0);
               TIMSK1 |= (1<<TOIE1);
               TCCR1B |= (1<<CS12)|(1<<CS10);
               TCNT1 = 49911;

 void READ_ADC(void) {
       unsigned char i=3;
       ADC_TEMP = 0;
       while (i--){
       ADCSRA |= (1<<ADSC);
       while(ADCSRA & (1<<ADSC));
       ADC_TEMP+= ADC;
       _delay_ms(150);
              }
       ADC_TEMP = ADC_TEMP/4;
 }
void print_config(void){
                    uint8_t D;
                    printf("done\n\n nRF24L01+ :\n");
                    nrf24_read(&D,1);
                    printf("0x%02X\n",D);
                    nrf24_read(&D,1);
                    printf("0x%02X\n",D);
                    nrf24_read(&D,1);
                    printf(" 0x%02X\n",D);
                    nrf24_read(&D,1);
                    printf("%02X\n",D);
                    nrf24_read(&D,1);
                    printf("0x%02X\n",D);
                    nrf24_read(&D,1);
                    printf("0x%02X\n",D);
                    nrf24_read(&D,1);
```

```
                              printf("0x%02X\n",D);
                              nrf24_read(&D,1);
                              printf("0x%02X\n",D);
}
          }
```

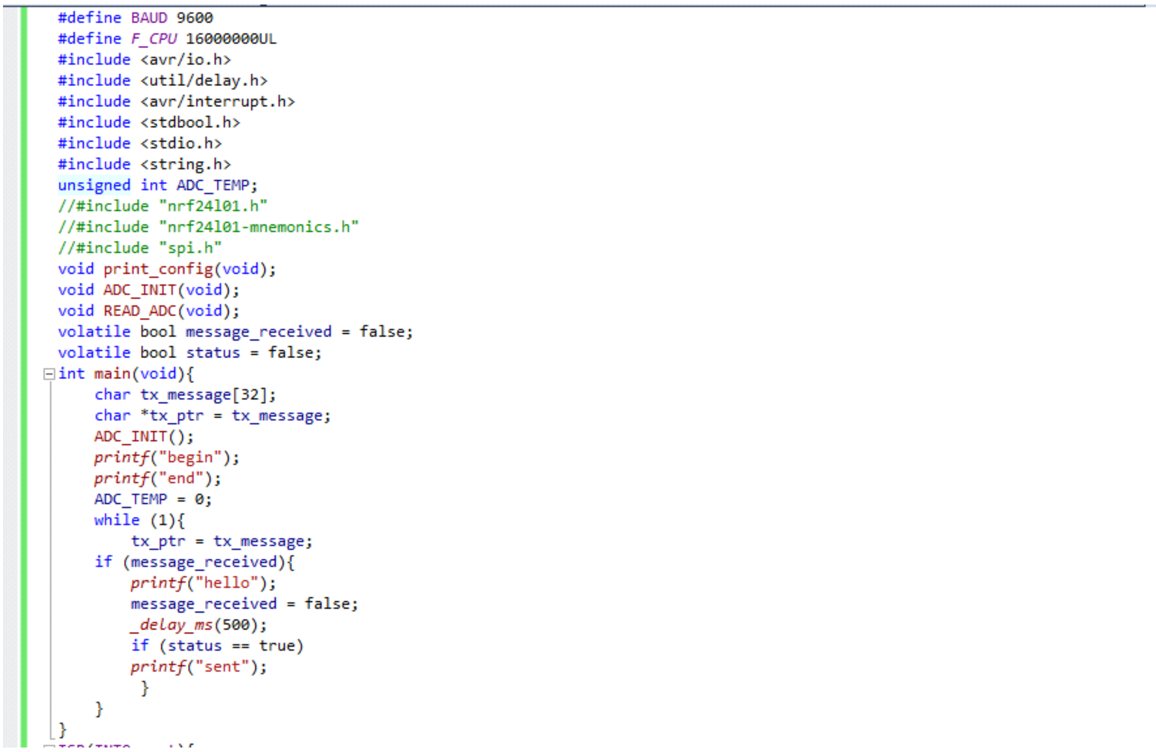
- **SCREENSHOTS OF EACH TASK OUTPUT (ATMEL STUDIO OUTPUT)**



```
        unsigned char i=3;
        ADC_TEMP = 0;
        while (i--){
        ADCSRA  |= (1<<ADSC);
        while(ADCSRA & (1<<ADSC));
        ADC_TEMP+= ADC;
        _delay_ms(150);
            }
        ADC_TEMP = ADC_TEMP/4;
    }
void print_config(void){
                uint8_t D;
                printf("done\n\n nRF24L01+ :\n");
                nrf24_read(&D,1);
                printf("0x%02X\n",D);
                nrf24_read(&D,1);
                printf("0x%02X\n",D);
                nrf24_read(&D,1);
                printf(" 0x%02X\n",D);
                nrf24_read(&D,1);
                printf("%02X\n",D);
                nrf24_read(&D,1);
                printf("0x%02X\n",D);
                nrf24_read(&D,1);
                printf("0x%02X\n",D);
                nrf24_read(&D,1);
                printf("0x%02X\n",D);
                nrf24_read(&D,1);
                printf("0x%02X\n",D);
  }
        }
```

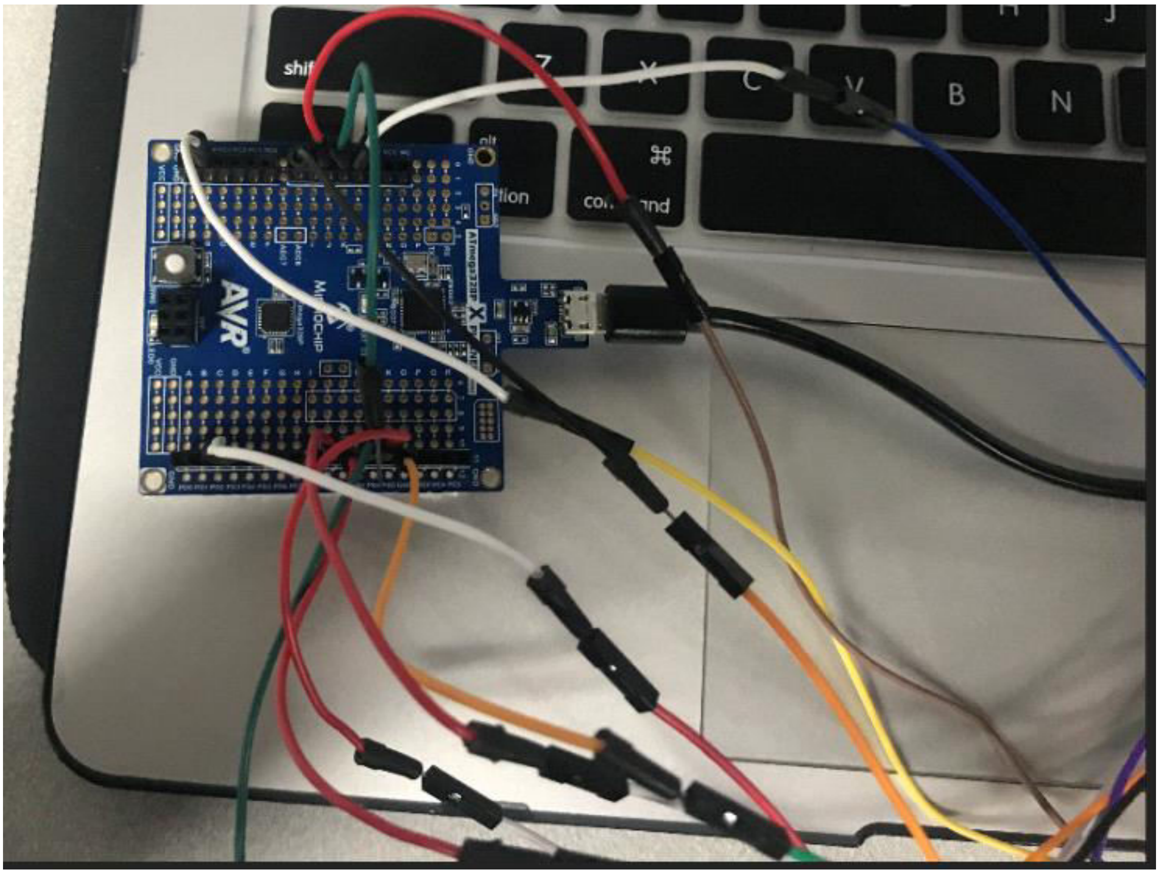


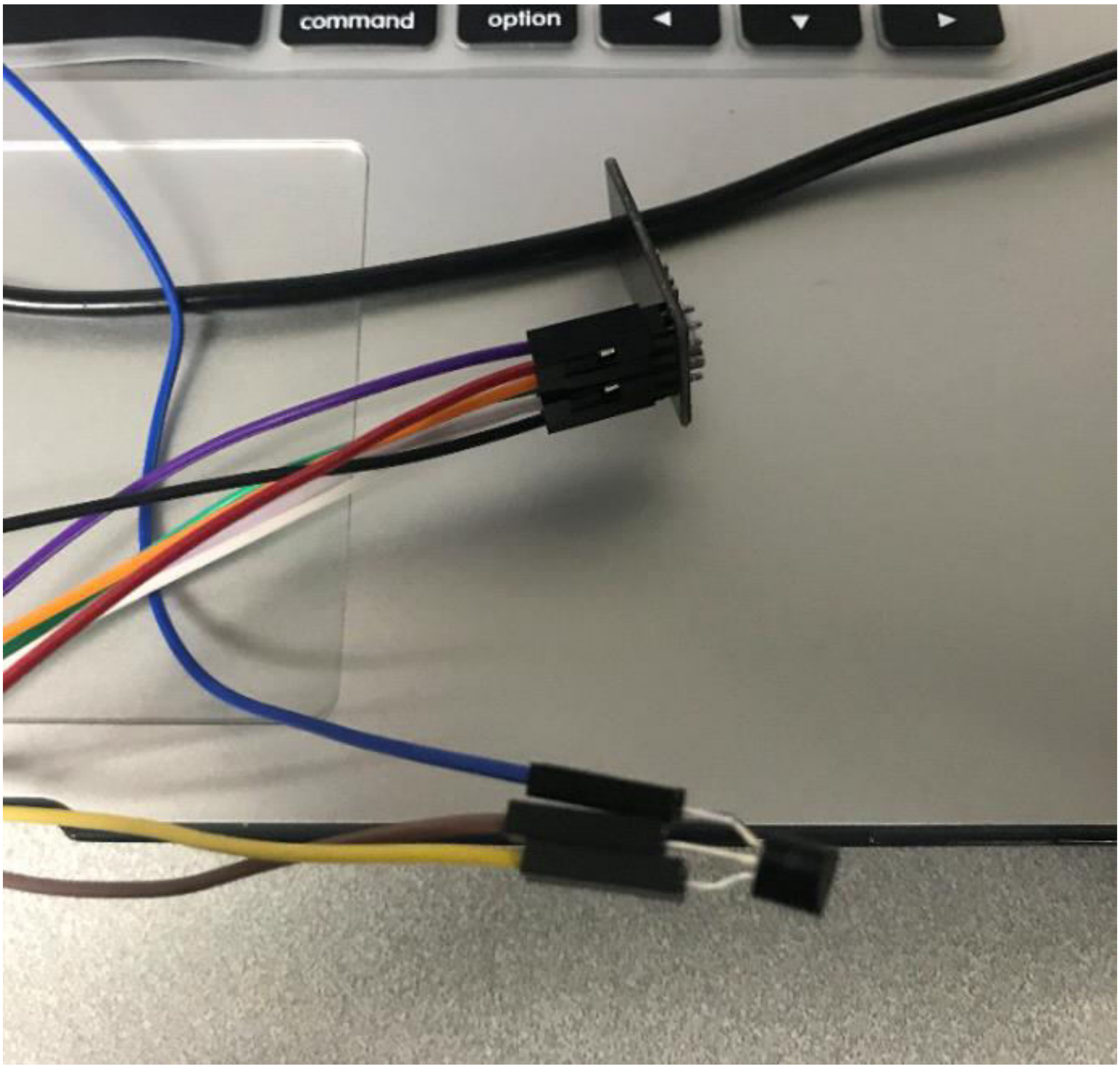
```
        message_received = true;
        }
void ADC_INIT(void){
            ADMUX = (0<<REFS1)|
            (1<<REFS0)|
            (0<<ADLAR)|
            (1<<MUX2)|
            (0<<MUX1)|
            (0<<MUX0);
            ADCSRA = (1<<ADEN)|
            (0<<ADSC)|
            (0<<ADATE)|
            (0<<ADIF)|
            (0<<ADIE)|
            (1<<ADPS2)|
            (0<<ADPS1)|  (1<<ADPS0);
            TIMSK1 |= (1<<TOIE1);
            TCCR1B |= (1<<CS12)|(1<<CS10);
            TCNT1 = 49911;

 void READ_ADC(void) {
      unsigned char i=3;
      ADC_TEMP = 0;
      while (i--){
      ADCSRA  |= (1<<ADSC);
      while(ADCSRA & (1<<ADSC));
      ADC_TEMP+= ADC;
      _delay_ms(150);
          }
      ADC_TEMP = ADC_TEMP/4;
  }
void print_config(void){
              uint8_t D;
              printf("done\n\n nRF24L01+ :\n");
              nrf24_read(&D,1);
              printf("0x%02X\n",D);
```



```
#define BAUD 9600
#define F_CPU 16000000UL
#include <avr/io.h>
#include <util/delay.h>
#include <avr/interrupt.h>
#include <stdbool.h>
#include <stdio.h>
#include <string.h>
unsigned int ADC_TEMP;
//#include "nrf24l01.h"
//#include "nrf24l01-mnemonics.h"
//#include "spi.h"
void print_config(void);
void ADC_INIT(void);
void READ_ADC(void);
volatile bool message_received = false;
volatile bool status = false;
int main(void){
    char tx_message[32];
    char *tx_ptr = tx_message;
    ADC_INIT();
    printf("begin");
    printf("end");
    ADC_TEMP = 0;
    while (1){
        tx_ptr = tx_message;
    if (message_received){
        printf("hello");
        message_received = false;
        _delay_ms(500);
        if (status == true)
        printf("sent");
         }
    }
}
```

- **SCREENSHOT OF EACH DEMO (BOARD SETUP)**





- **VIDEO LINKS OF EACH DEMO**
- **GITHUB LINK OF THIS DA**

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

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