**CPE301 – SPRING 2019**

# Midterm 1

Student Name: Brian West
Student #: 5003032874
Student Email: westb2@unlv.nevada.edu
Primary Github address: https://github.com/westbrian2/Spring2019
Directory: Spring2019/Midterms/Midterm1

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.

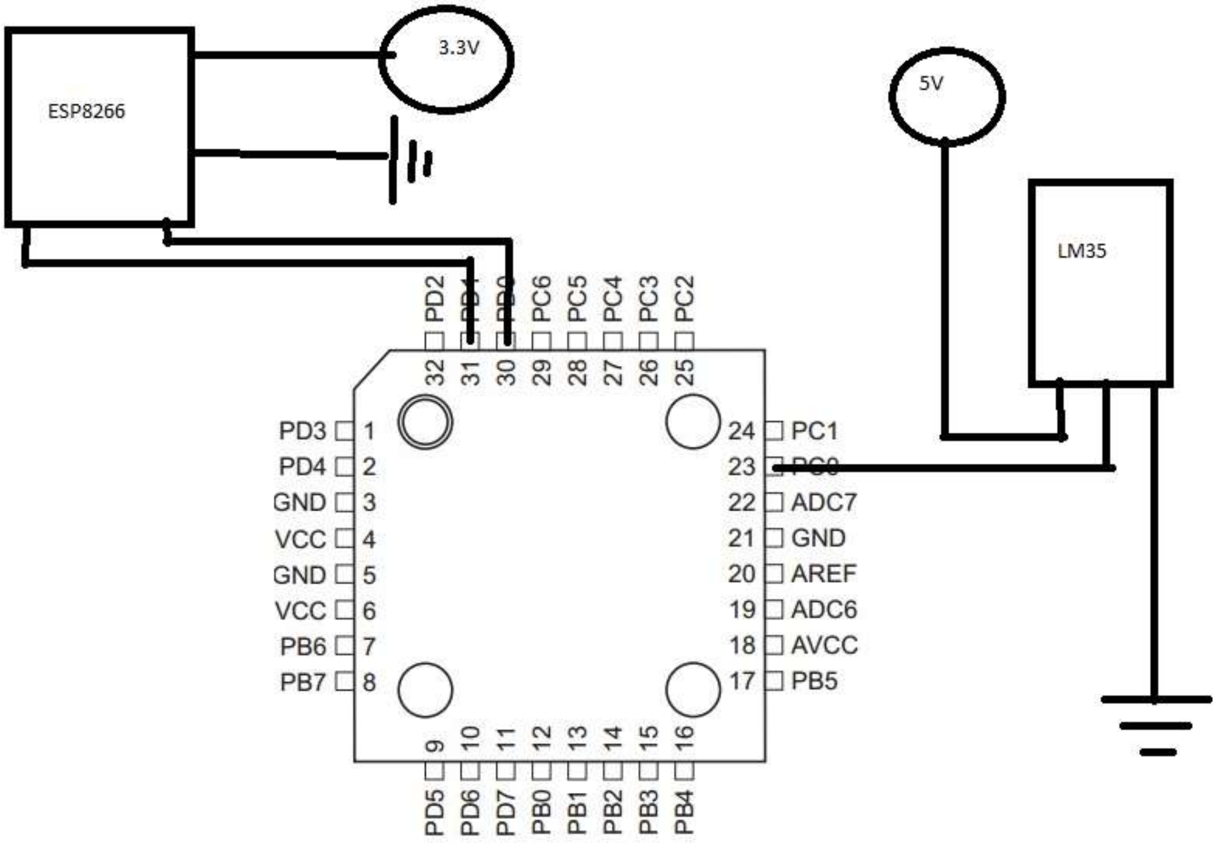
2. Use the previously create a Github repository with a random name (no CPE/301, Lastname, Firstname). Place all labs under the root folder ESD301/DA, sub-folder named LABXX, with one document and one video link file for each lab, place modified asm/c files named as LabXX-TYY.asm/c.

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

COMPONENTS: Xplained Mini board, LM35, ESP8266, ESP01 to USB module

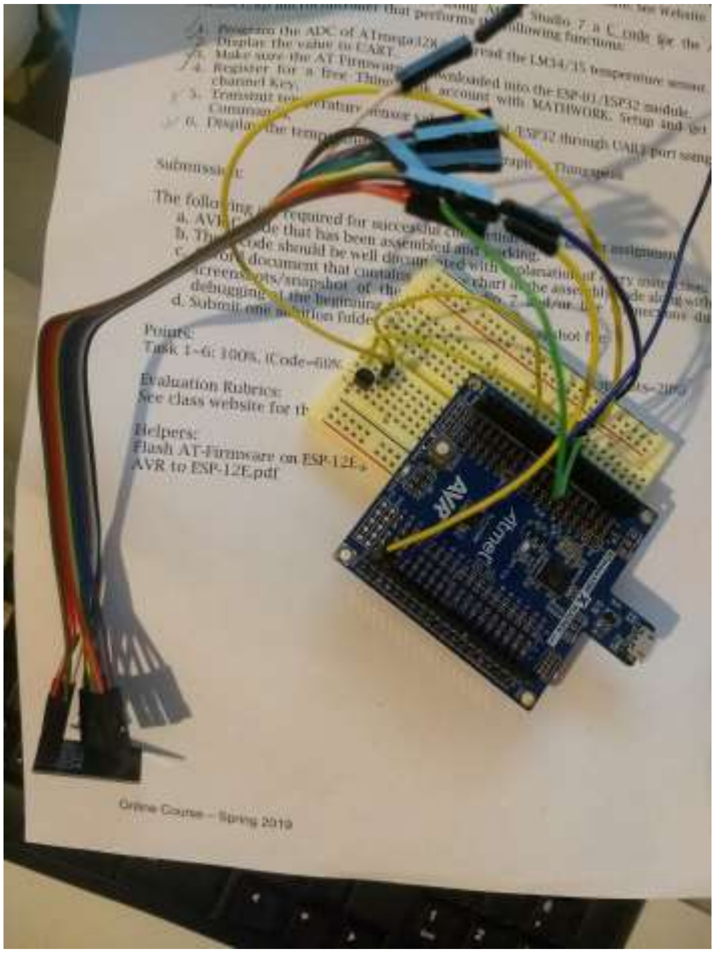


## 2. INITIAL/MODIFIED/DEVELOPED CODE OF TASK 1/A

```
#define F_CPU 16000000UL
#define UBRR_9600 103 //Baud rate for 16MHz

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

void read_adc(void);
void adc_init(void);

void USART_init( unsigned int ubrr ); //Sets up usart for use
void USART_tx_string( char *data ); //function that outputs data (usart)



char output[50]; //used to output value
volatile unsigned int adc_temp; //holds temp value

int main(void){
```

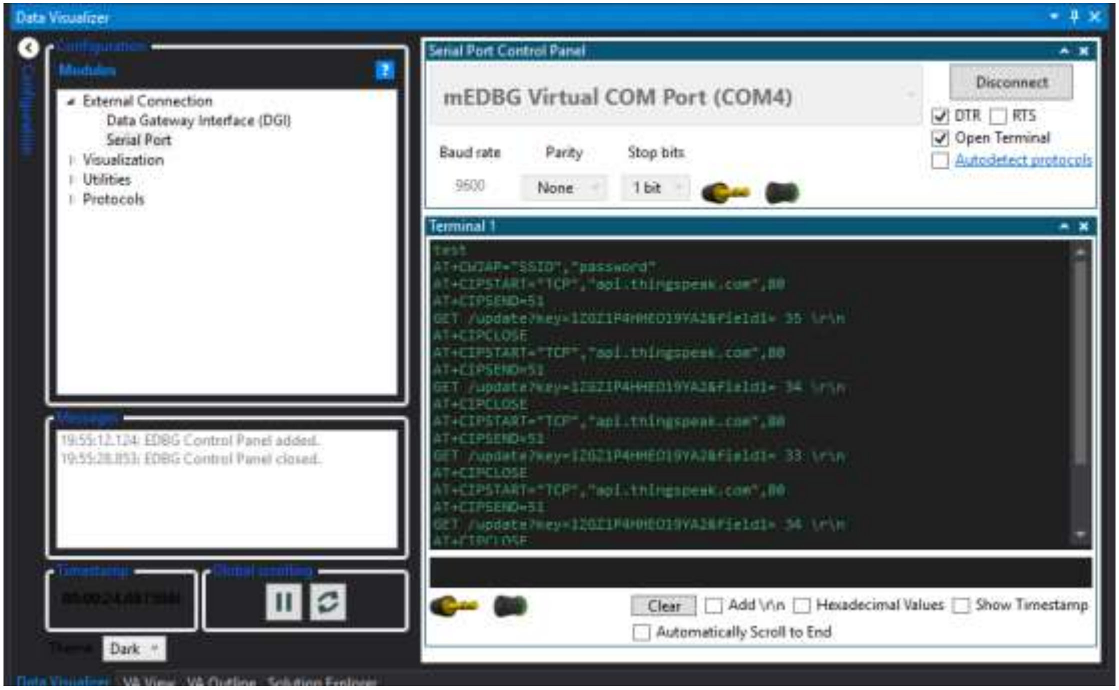
```
adc_init(); //setting up ADC
USART_init(UBRR_9600);
USART_tx_string("test\r\n");//testing usart
snprintf(output,sizeof(output),"AT+CWJAP=\"SSID\",\"password\"\r\n"); //connects to
network
USART_tx_string(output);
_delay_ms(150);
        while(1){
                read_adc();

        snprintf(output,sizeof(output),"AT+CIPSTART=\"TCP\",\"api.thingspeak.com\",80\r\n"
); //starts session
                USART_tx_string(output);//connecting to thingspeak

                _delay_ms(300);
                snprintf(output,sizeof(output),"AT+CIPSEND=51\r\n"); //prepares to send
data
                USART_tx_string(output);//
                _delay_ms(300);
                snprintf(output,sizeof(output),"GET /update?key=1ZGZ1P4HHEO19YA2&field1=%3d
\\r\\n\r\n",adc_temp);
                USART_tx_string(output);//send temp value
                _delay_ms(300);
                snprintf(output,sizeof(output),"AT+CIPCLOSE\r\n");
                USART_tx_string(output); //close session
                _delay_ms(1000);
        }
}


void USART_init(unsigned int ubrr){
        UBRR0H=(unsigned char)(ubrr>>8); //Setting up
        UBRR0L=(unsigned char)(ubrr);
        UCSR0B=(1<<TXEN0)|(1<<RXEN0);//Enabling reciever, transmitter, and rx interrupt
        UCSR0C=(1<<UCSZ01)|(1<<UCSZ00); //async  8 n 1
}

void USART_tx_string(char *data){ //sends string
        while((*data!= '\0')){
                while(!(UCSR0A&(1<<UDRE0)));
                UDR0=*data;
                data++; //gets next part of data
        }
}
void adc_init(void)
{
        /** Setup and enable ADC **/
        ADMUX = (0<<REFS1)| // Reference Selection Bits
        (1<<REFS0)| // AVcc - external cap at AREF
        (0<<ADLAR)| // ADC Left Adjust Result
        (0<<MUX2)| // setting input to PC0
        (0<<MUX1)|
        (0<<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); // Select Channel
}
void read_adc(void) {
        unsigned char i =4; //to get 4 samples
        adc_temp = 0;
        while (i--) {
                ADCSRA |= (1<<ADSC); //start conversion
                while(ADCSRA & (1<<ADSC)); //waiting for coversion to finish
                adc_temp+= ADC;

                _delay_ms(50);
        }
        adc_temp = (adc_temp / 4)-20; // Average a few samples and adjusts for slight
offset
}
```

## 3. SCHEMATICS



Picture of the board wired

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



mEDBG Virtual COM Port (COM4)

☑ DTR □ RTS
☑ Open Terminal
□ Autodetect protocols

Baud rate: 9600 Parity: None Stop bits: 1 bit

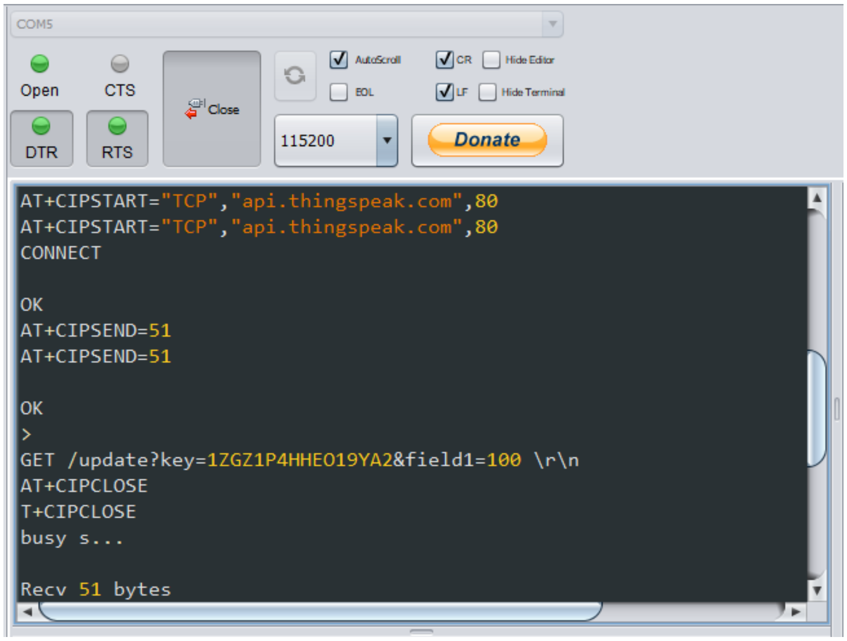
□ Add \r\n □ Hexadecimal Values □ Show Timestamp
□ Automatically Scroll to End

Theme: Dark

```
19:55:12.124: EDBG Control Panel added.
19:55:28.853: EDBG Control Panel closed.
```

```
test
AT+CWJAP="SSID","password"
AT+CIPSTART="TCP","api.thingspeak.com",80
AT+CIPSEND=51
GET /update?key=1ZGZ1P4HHEO19YA2&field1= 35 \r\n
AT+CIPCLOSE
AT+CIPSTART="TCP","api.thingspeak.com",80
AT+CIPSEND=51
GET /update?key=1ZGZ1P4HHEO19YA2&field1= 34 \r\n
AT+CIPCLOSE
AT+CIPSTART="TCP","api.thingspeak.com",80
AT+CIPSEND=51
GET /update?key=1ZGZ1P4HHEO19YA2&field1= 33 \r\n
AT+CIPCLOSE
AT+CIPSTART="TCP","api.thingspeak.com",80
AT+CIPSEND=51
GET /update?key=1ZGZ1P4HHEO19YA2&field1= 34 \r\n
AT+CIPCLOSE
```

Screenshot of UART output working.



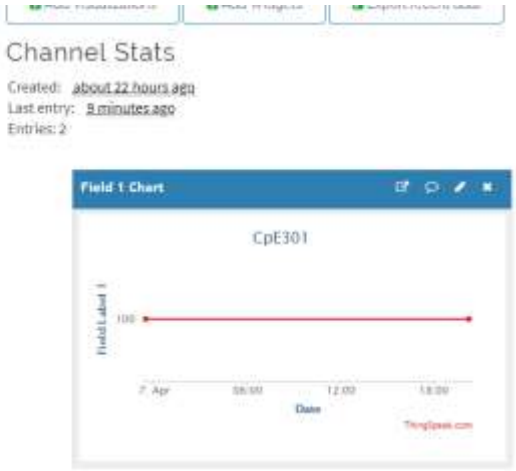
COM5

☑ AutoScroll □ EOL ☑ CR ☑ LF □ Hide Editor □ Hide Terminal

115200

```
AT+CIPSTART="TCP","api.thingspeak.com",80
AT+CIPSTART="TCP","api.thingspeak.com",80
CONNECT

OK
AT+CIPSEND=51
AT+CIPSEND=51

OK
>
GET /update?key=1ZGZ1P4HHEO19YA2&field1=100 \r\n
AT+CIPCLOSE
T+CIPCLOSE
busy s...

Recv 51 bytes
```

Picture showing the ESP8266 was programmed correctly when programmed from computer
I fixed the issue I was having with the flasher program by changing to an older version of JDK.



Channel Stats

Created: about 22 hours ago
Last entry: 9 minutes ago
Entries: 2

Picture showing that I was able to get the ESP to send data to thingspeak using ESPlorer.

## 5. VIDEO LINKS OF EACH DEMO

I really don’t know what you would want videos of.

## 6. GITHUB LINK OF THIS DA

https://github.com/westbrian2/Spring2019/Midterms/Midterm1

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

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