

CPE301 – SPRING 2019
MIDTERM 1

Student Name: Ricky Perez

Student #: 5002297620

Student Email: perezr1@unlv.nevada.edu

Primary Github address: https://github.com/RickyPerez79/submission_da.git

Directory: 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/Midterm, 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

- Xplained mini
- Usb
- ESP32 chip

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

```
/*
 * Midterm_Project.c
 *
 * Created: 4/5/2019 1:46:32 PM
 * Author : perezr1
 */
#define F_CPU 16000000UL
#define BAUD_RATE 9600
#define My_UBRR F_CPU/16/BAUD_RATE-1

/*****Include
Library*****/
#include <avr/io.h>
#include <avr/interrupt.h>
#include <util/delay.h>
#include <stdio.h>
#include <stdlib.h>
/*****
*****/

/*****Prototype
Functions*****/
void read_adc(void); //Read ADC
void USART_tx_string(char *data); //Print String USART
void USART_init(unsigned int UBRR); // Set up the USART Baud Rate Register
/*****
*****/

char results[256]; //array to hold my output
volatile unsigned int ADC_Temperature;
volatile char received_data;
// CALCULATIONS FOR TIMER1:
//TCNT1 = 65535 - ( ((16MHz/1024)*1)-1) = 49911

int main(void) {
    USART_init(My_UBRR); // calls function
    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)| // ADC5 (PC5, PIN28)
    (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 interrupt flag
// Set Prescaler
TCCR1B = 5; // setting the prescalar to 1024
// Set timer
TCNT1 = 49911; // set TCNT1

_delay_ms(1000); // wait a bit
sei(); //interrupt

while(1)
{
    // wait here
}

/* calculates temperature */
void read_adc(void) {
    unsigned char i =4;
    ADC_Temperature = 0; //initialize to zero
    while (i--) {
        ADCSRA |= (1<<ADSC);
        while(ADCSRA & (1<<ADSC));
        ADC_Temperature+= ADC;
        _delay_ms(50);
    }
    ADC_Temperature = ADC_Temperature /8; // gather a few samples
}

/***** Functions *****/

void USART_init( unsigned int ubrr ) {

    UBRRE0H = (unsigned char)(ubrr>>8); // set upper byte 0
    UBRRE0L = (unsigned char)ubrr; // set lower byte to the value of
F_CPU/16/BAUD_RATE-1

    UCSR0B |= (1 << TXEN0) | (1 << RXEN0)| ( 1 << RXCIE0); // Enable receiver,
transmitter & RX interrupt
    UCSR0C |= (1<<UCSZ01) | (1 << UCSZ00);
}

void USART_tx_string( char *data ) {
    while ((*data != '\0')) {
        while (!(UCSR0A & (1 <<UDRE0)));

```

```

        UDR0 = *data;
        data++;
    }
}

/*****
*****/

ISR(TIMER1_OVF_vect) //timer overflow interrupt to delay for 1 second
{
    char TEMP[256];
    unsigned char AT_COMMANDS[] = "AT\r\n"; //AT Commands
    unsigned char CWMODE[] = "AT+CWMODE=1\r\n"; //Set the mode
    unsigned char CWJAP[] = "AT+CWJAP=\"Itsa_Me_Ricky\", \"xzft3981\"\r\n"; // WIFI
    username and password
    unsigned char CIPMUX[] = "AT+CIPMUX=0\r\n";
    unsigned char CIPSTART[] = "AT+CIPSTART=\"TCP\", \"api.thingspeak.com\", 80\r\n";
    unsigned char CIPSEND[] = "AT+CIPSEND=100\r\n";

    _delay_ms(2000);
    USART_tx_string(AT_COMMANDS); //send commands

    _delay_ms(5000);
    USART_tx_string(CWMODE); //set mode

    _delay_ms(5000);
    USART_tx_string(CWJAP); //connect to Wifi

    _delay_ms(5000);
    USART_tx_string(CIPMUX); //select MUX

    _delay_ms(5000);
    USART_tx_string(CIPSTART); //connect TCP

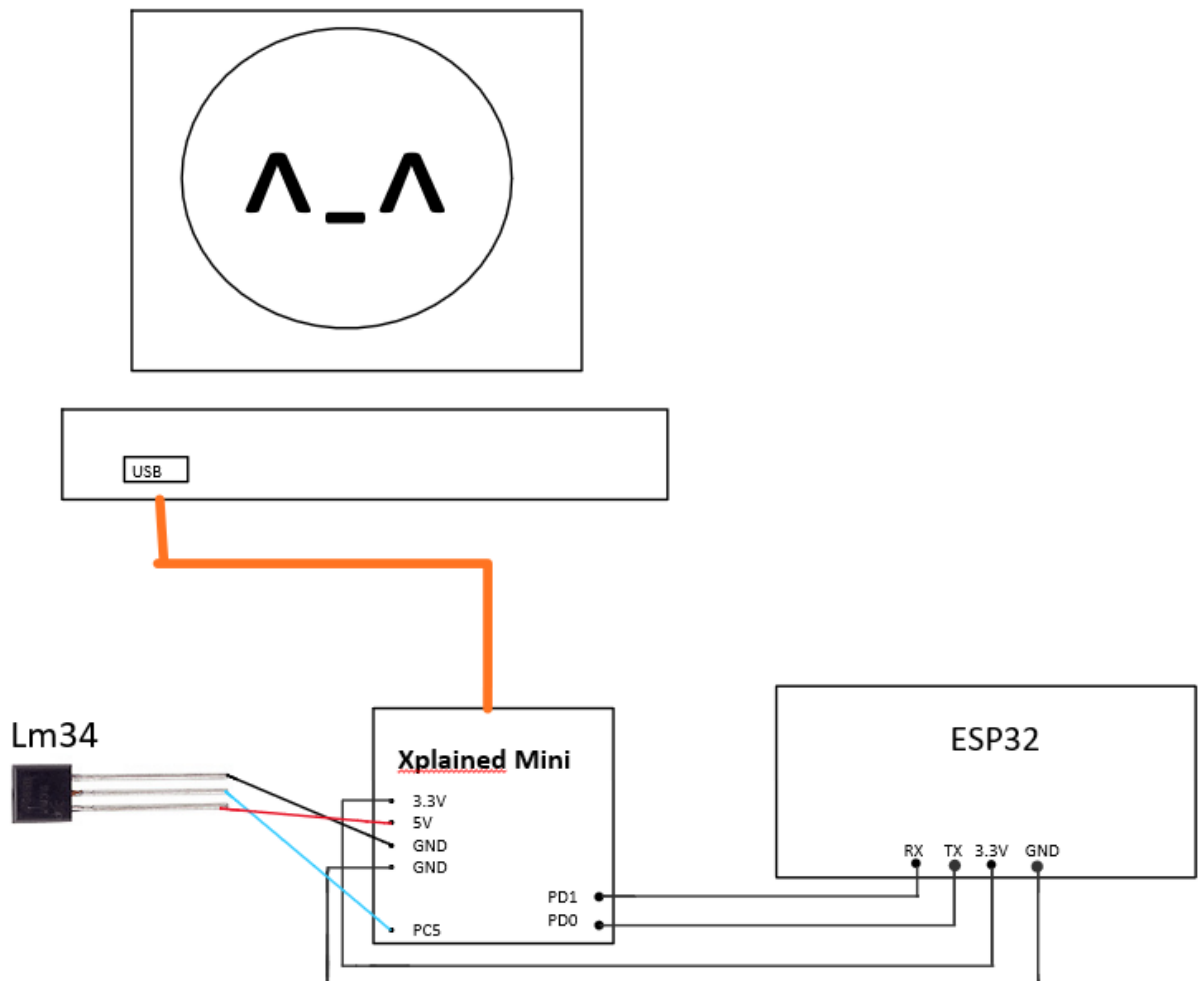
    _delay_ms(5000);
    USART_tx_string(CIPSEND); //send size

    _delay_ms(5000);

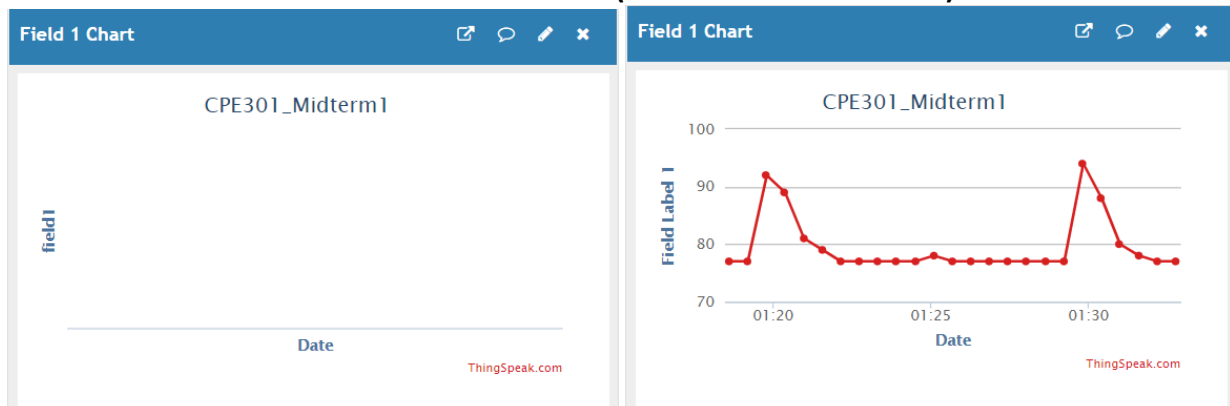
    read_adc(); //read ADC
    snprintf(results, sizeof(results), "GET
https://api.thingspeak.com/update?api_key=PXJ5Q3YZQDNNE9FS&field1=%3d\r\n",
ADC_Temperature); // print
    USART_tx_string(results); //send result of the data gathered
    _delay_ms(3000); // lets it give it some time
    TCNT1 = 49911; //resets timer
}

```

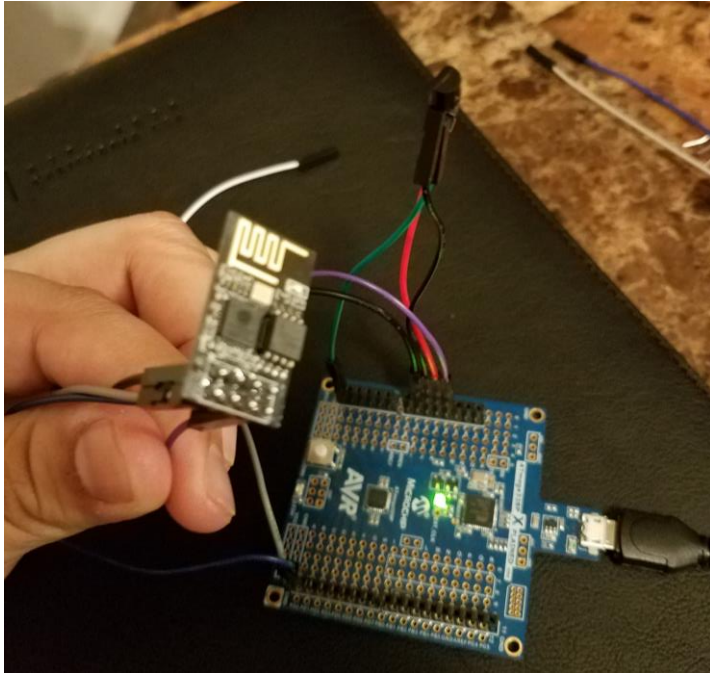
3. SCHEMATICS



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



5. SCREENSHOT OF EACH DEMO (BOARD SETUP)



6. VIDEO LINKS OF EACH DEMO

https://youtu.be/p_E59KMaKIs

7. GITHUB LINK OF THIS DA

https://github.com/RickyPerez79/submission_da.git

Student Academic Misconduct Policy

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

"This assignment submission is my own, original work".

RICKY PEREZ