CPE301 – SPRING 2019

Design Assignment 3B

Student Name: Brian West

Student #: 5003032874

Student Email: westb2@unlv.nevada.edi

Primary Github address: <https://github.com/westbrian2/Spring2019>

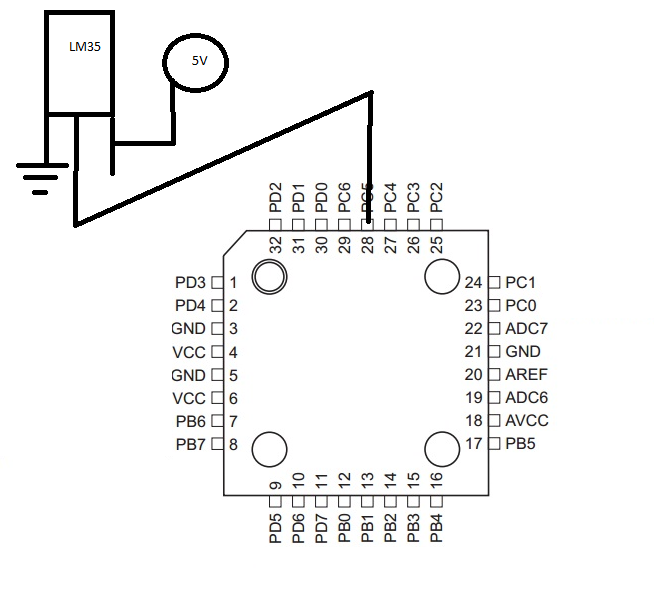
Directory: Spring2019/DesignAssignments

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**

Mini Xplained, LM35



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

#define F\_CPU 16000000UL

#define UBRR\_9600 103 // for 16Mhz with .2% error

#include <avr/io.h>

#include <util/delay.h>

#include <stdio.h>

void read\_adc(void); // Function Declarations

void adc\_init(void);

void USART\_init( unsigned int ubrr );

void USART\_tx\_string( char \*data );

volatile unsigned int adc\_temp;

char outs[20];

int main(void) {

adc\_init(); // Initialize the ADC (Analog / Digital Converter)

USART\_init(UBRR\_9600); // Initialize the USART (RS232 interface)

USART\_tx\_string("Start\r\n"); // we're alive!

\_delay\_ms(125); // wait a bit

while(1)

{

read\_adc();

snprintf(outs,sizeof(outs),"%3d\r\n", adc\_temp); // print it

USART\_tx\_string(outs);

\_delay\_ms(1000); // wait a bit

}

}

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

(1<<MUX2)| // ANalog Channel Selection Bits

(0<<MUX1)| // ADC2 (PC5)

(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); // Select Channel

}

void read\_adc(void) {

unsigned char i =4;

adc\_temp = 0;

while (i--) {

ADCSRA |= (1<<ADSC);

while(ADCSRA & (1<<ADSC));

adc\_temp+= ADC;

\_delay\_ms(50);

}

adc\_temp = (adc\_temp / 4); // Average a few samples

adc\_temp =(5\*adc\_temp\*100)/1024; //gets C temp

adc\_temp=(adc\_temp\*1.8)+32.0; //coverts to F

}

void USART\_init( unsigned int ubrr ) {

UBRR0H = (unsigned char)(ubrr>>8);

UBRR0L = (unsigned char)ubrr;

UCSR0B = (1 << TXEN0); // Enable receiver, transmitter & RX interrupt

UCSR0C = (1 << UCSZ01) | (1 << UCSZ00); //asynchronous 8 N 1

}

void USART\_tx\_string( char \*data ) {

while ((\*data != '\0')) {

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

UDR0 = \*data;

data++;

}

}

1. **SCHEMATICS**



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



1. **GITHUB LINK OF THIS DA**

https://github.com/westbrian2/Spring2019/DesignAssignments/DA3B

**Student Academic Misconduct Policy**

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

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

Brian West