CPE301 – SPRING 2019

MIDTERM 1

Student Name: Henry Mesa

Student #: 1008099273

Student Email: mesah1@unlv.nevada.edu

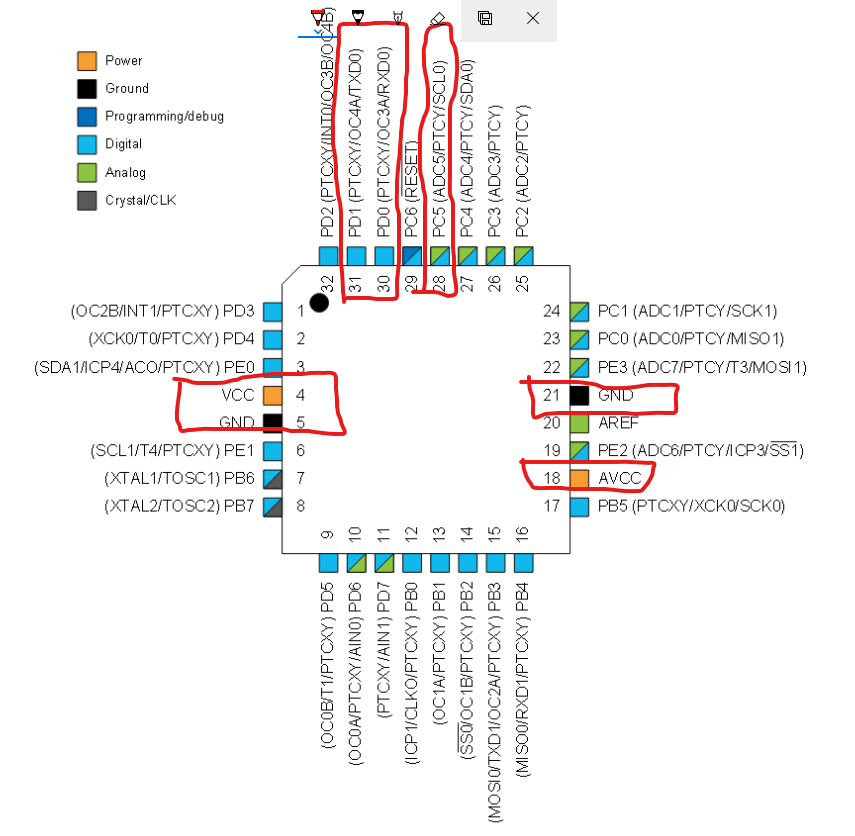
Primary Github address: https://github.com/mesah1

Directory: https://github.com/mesah1/submissions

1. **COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS**

List of Components used:

Atmega328P, LM35, ESP01



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

/\*

\* Midterm.c

\*

\* Created: 10/26/2019 2:06:31 PM

\* Author : Henry Mesa

\* TOOL: LM35, EXPLAINED-MINI(ATMEGA328PB), ESP01

\* THIS CODE WILL UPLOAD TEMPERATURE VALUES TO THINGSPEAK.COM THROUGHT USART

\*

\*/

#define *F\_CPU* 16000000UL

#define UBRR\_9600 103

#include <avr/io.h>

#include <util/delay.h>

#include <stdio.h>

#include <stdlib.h>

#include <avr/interrupt.h>

//Function declaration

void read\_adc(void);

void adc\_init(void);

void USART\_init( );

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

volatile float adc;

char ar[255];

int main(void)

{

adc\_init(); //intialize ADC

read\_adc();

USART\_init(UBRR\_9600); //initialize USART

USART\_tx\_string("Connected!\n");

*snprintf*(ar, sizeof(ar),"%f\n\r",adc/128); //print

USART\_tx\_string(ar);

while (1)

{

}

}

/\* INITIALIZING ADC \*/

void adc\_init(void){

//ADC ENABLE AND SET-UP

ADMUX = (0 << REFS1)| // external cap at AREF

(1 << REFS0)| // AVcc - Reference Selection INTERNAL

(1 << ADLAR)| // ADC Left Adjust Result

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

(0 << MUX1)| // ADC0 (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

(0 << ADPS1)|

(1 << ADPS0);

// Timer/Counter1 Interrupt Mask Register

TIMSK1 |= (1<<TOIE1); // enable interrupt flag

// Set Prescalar

TCCR1B = 5; // setting the prescalar to 1024

// Set timer

TCNT1 = 7000; // set TCNT1

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

sei(); //interrupt

}

/\* READ ADC PINS \*/

void read\_adc(void){

unsigned char i = 4;

adc = 0;

while (i--)

{

ADCSRA |= (1 << ADSC);

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

adc+= ADC;

*\_delay\_ms*(50);

}

adc = adc / 4; // Average a few samples

}

/\* INITIALIZING USART (RS-232) \*/

void USART\_init(unsigned int ubrr){

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

UBRR0L = (unsigned char)ubrr;

UCSR0B = (1 << TXEN0)|(1 << RXEN0)| ( 1 << RXCIE0); //transreciver enabled and RX interrupt

UCSR0C = (1 << UCSZ01) | (1 << UCSZ00); //5-bit characters

}

/\* SEND STRING TO RS-232 \*/

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

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

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

UDR0 = \*data;

data++;

}

}

void UART\_sendString(volatile unsigned char AT[]){

volatile unsigned char len = 0;

volatile unsigned char i;

while (AT[len] != 0){

len++;

}

for (i = 0x00; i < len; i++)

{

//Wait for the transmitter to finish

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

{

UDR0 = AT[i];

}

}

}

ISR(TIMER1\_OVF\_vect){

unsigned char AT\_COMMMANDS[] = "AT\n\r"; //AT Commands

unsigned char CWMODE[] ="AT+CWMODE=1\n\r"; //Set the mode

unsigned char CWJAP[] ="AT+CWJAP=\"AquaMan\",\"AMI@6280#life\"\r\n"; // WIFI username and password

unsigned char CIPMUX[] ="AT+CIPMUX=0\n\r";

unsigned char CIPSTART[] ="AT+CIPSTART=\"TCP\",\"api.thingspeak.com\",80\n\r";

unsigned char CIPSEND[] ="AT+CIPSEND=51\n\r";

unsigned char DATA[]="GET /update?api\_key=0DSHG7MOCFW2VGFI&field1=";

unsigned char CLOSE[]="AT+CIPCLOSE\n\r";

read\_adc();

*snprintf*(ar, sizeof(ar),"%f\n\r",adc/128); //print

UART\_sendString(AT\_COMMMANDS);

*\_delay\_ms*(3000);

UART\_sendString(CWMODE);

*\_delay\_ms*(2000);

UART\_sendString(CWJAP);

*\_delay\_ms*(3000);

UART\_sendString(CIPMUX);

*\_delay\_ms*(2000);

UART\_sendString(CIPSTART);

*\_delay\_ms*(3000);

UART\_sendString(CIPSEND);

*\_delay\_ms*(3000);

UART\_sendString(DATA);

UART\_sendString(ar);

*\_delay\_ms*(3000);

UART\_sendString(CLOSE);

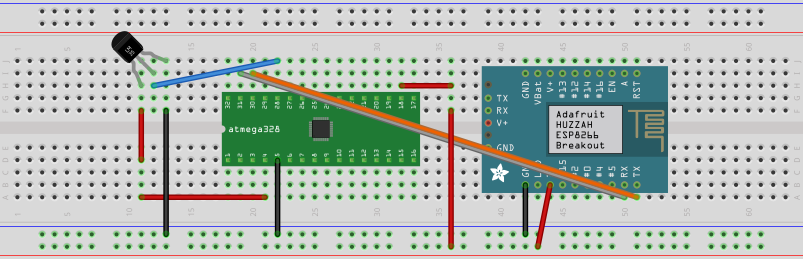
*\_delay\_ms*(2000);

USART\_tx\_string(ar);

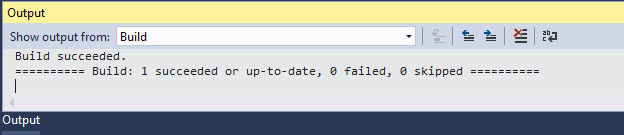
TCNT1 = 7000; //resets timer

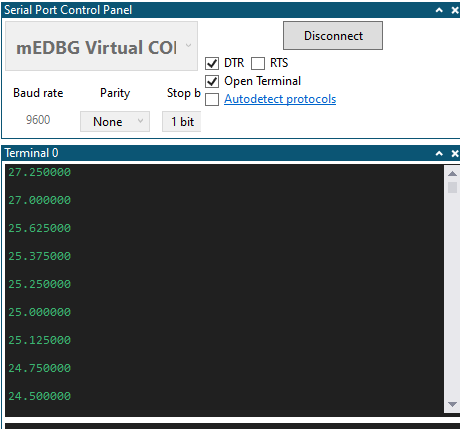
}

1. **SCHEMATICS**

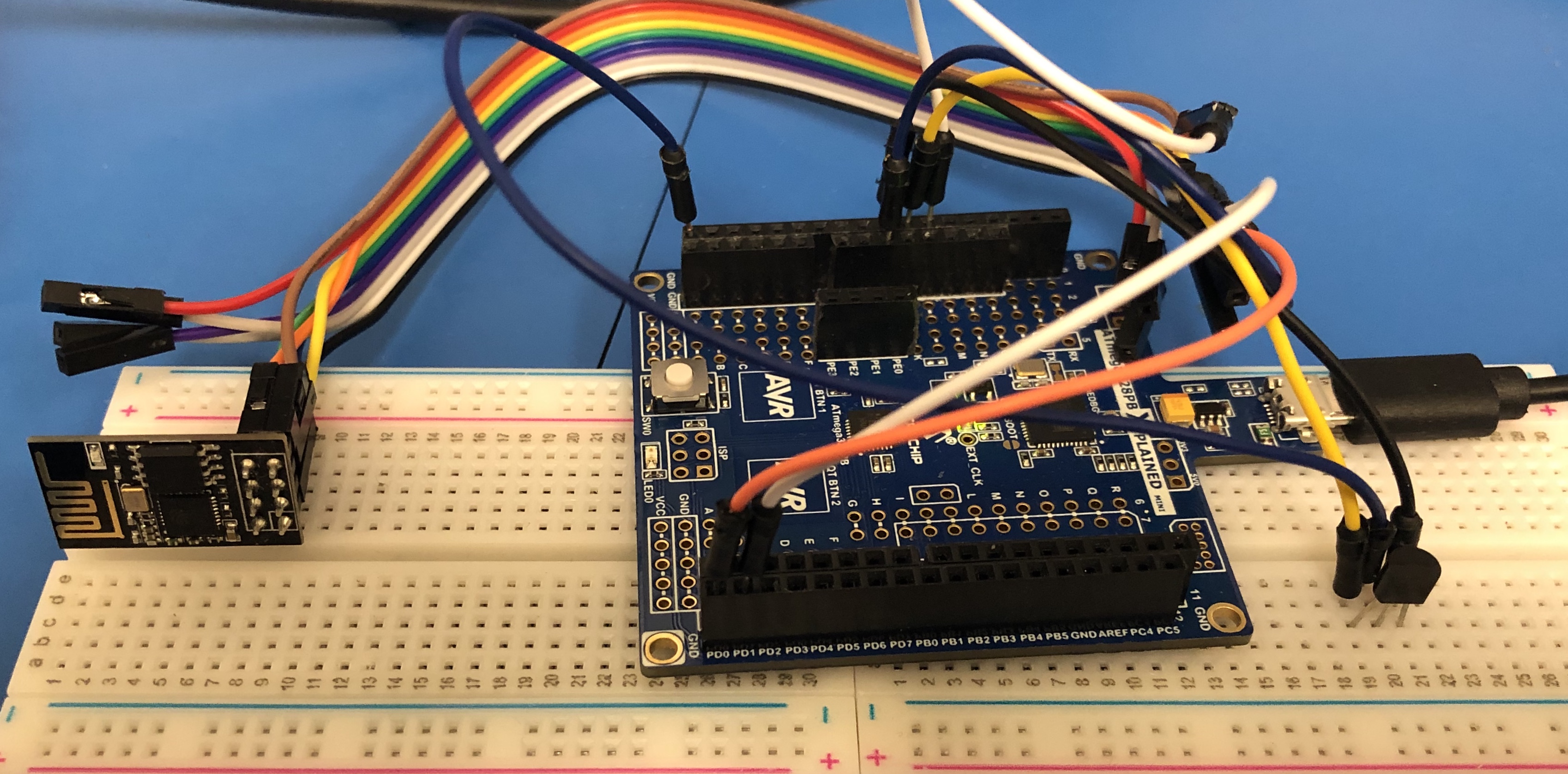


1. **SCREENSHOTS OF EACH TASK OUTPUT (ATMEL STUDIO OUTPUT)**





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



1. **VIDEO LINKS OF EACH DEMO**

<https://youtu.be/xVnoOiG9VTE>

1. **THINGSPEAK LINK**

<https://thingspeak.com/channels/895725>

1. **GITHUB LINK OF THIS DA**

<https://github.com/mesah1/submissions/tree/master/Midterm>

**Student Academic Misconduct Policy**

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

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

HENRY MESA