CPE301 – SPRING 2019

Design Assignment X

Student Name: Jeb Marinas

Student #:5003376451

Student Email: marinj4@unlv.nevada.edu

Primary Github address:github.com/jebmarinas

Directory:

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**

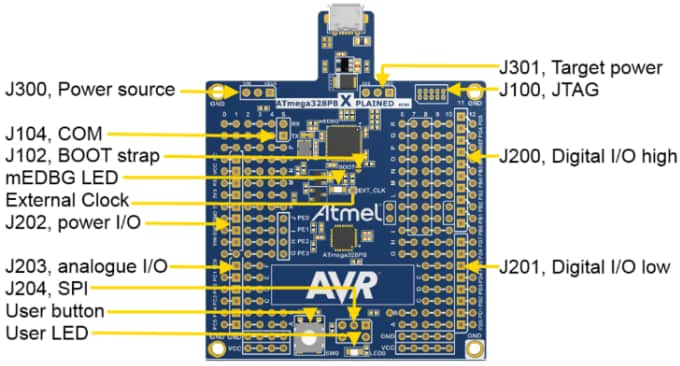
Atmega 328pb

XminiPc

Lm35

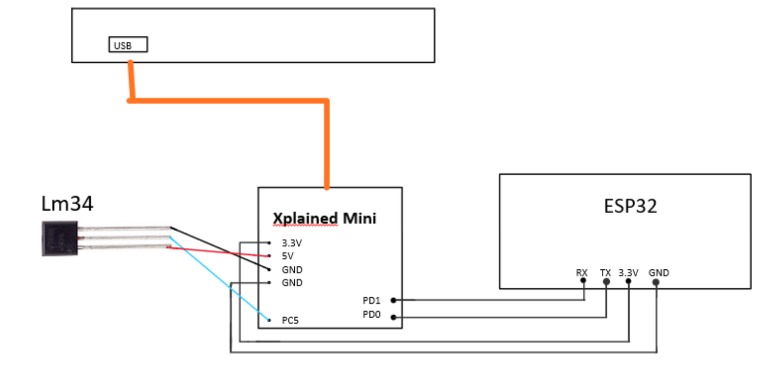
ESP-01

Esp8266

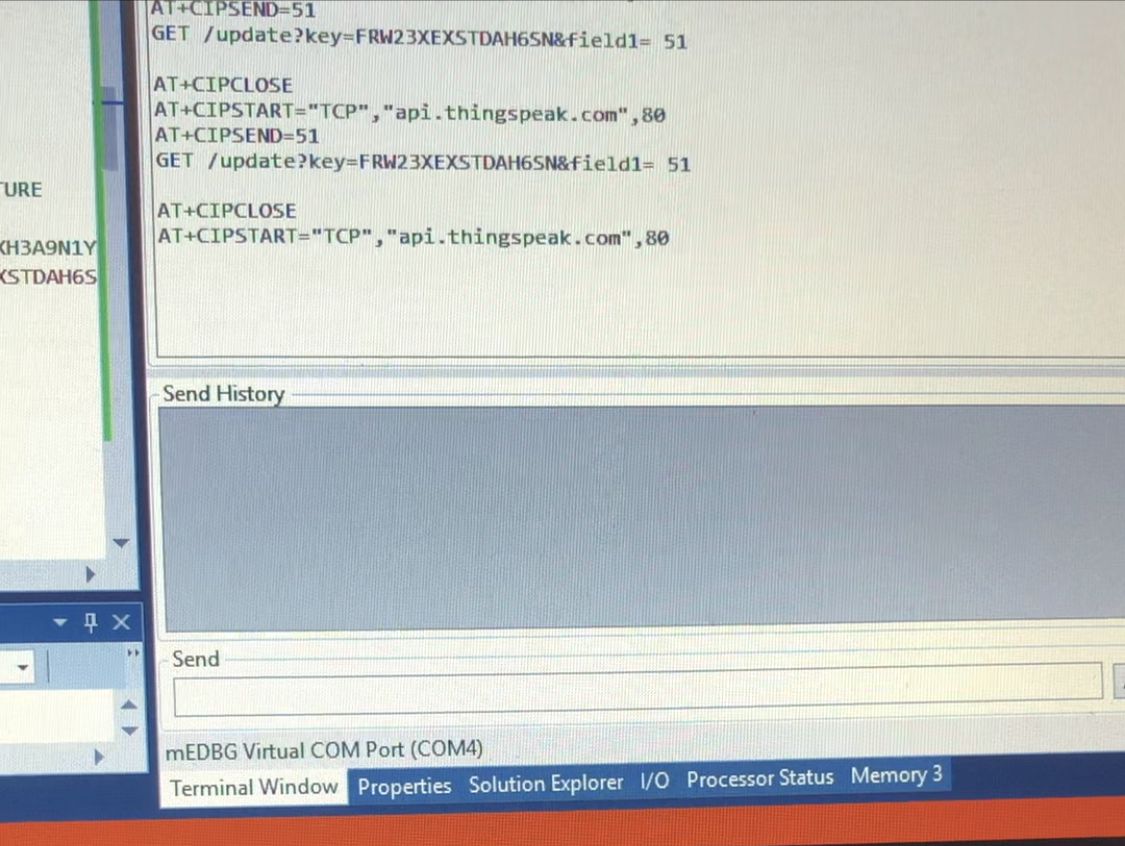


1. **INITIAL/MODIFIED/DEVELOPED CODE OF TASK 1/A**
2. #define *F\_CPU* 16000000UL
3. #define BAUD 9600
4. #include <avr/io.h>
5. #include <util/delay.h>
6. #include <avr/interrupt.h>
7. #include <util/setbaud.h>
8. #include <stdio.h>
9. volatile *uint8\_t* of;
10. volatile float adc\_temp;
11. char direct1[] = "AT";
12. char direct2[] = "AT+CWMODE=1"; //this will enable the wifi
13. char direct3[] = "AT+CWLAP"; // list wireless wifi AP in range
14. char direct4[] = "AT+CWJAP=\"WIFI-NAME\",\"PASSWORD\""; // this will join the network
15. char direct5[] = "AT+CIPSTART=\"TCP\",\"api.thingspeak.com\",80";
16. char direct6[] = "AT+CIPSEND=51 \\";
17. char direct7[] = "AT+CIPCLOSE";
18. void usart\_init(void)
19. {
20. UBRR0H = *UBRRH\_VALUE*; // loads the ubrrh value into high
21. UBRR0L = *UBRRL\_VALUE*;// loads the ubrr0l value into low
22. UCSR0C = \_BV(UCSZ01) | \_BV(UCSZ00); // 8-bit data
23. UCSR0B = \_BV(RXEN0) | \_BV(TXEN0);// enable RX and TX
24. }
25. void set()
26. {
27. TCCR0A = 0; // normal mode is initiated
28. TCCR0B = 0x05;// the prescaler is set to 1024
29. TCNT1 = 0x00;// initiates the counter to 0
30. TIMSK0 = (1<<TOIE0); // enable interrupt
31. sei(); // enable global interrupt
32. }
33. void usart\_tx\_string(char \* info)
34. {
35. while((\*info != '\0'))
36. {
37. while( !(UCSR0A & (1 << UDRE0)));
38. UDR0 = \*info;
39. info++;
40. }
41. }
42. ISR(TIMER0\_OVF\_vect)
43. {
44. of++; // this will increment the overflow
45. }
46. void adc\_init(void)
47. {
48. ADMUX = (0<<REFS1)| // reference selection bit
49. (1<<REFS0)| //AVCC EXTERNAL CAP AT REF
50. (0<<ADLAR)| //ADC LEFT ADJUST RESULT
51. (1<<MUX2)| //ANALOG CHANNEL SELECTION BITS
52. (0<<MUX1)| //ADC
53. (0<<MUX0);
54. ADCSRA = (1<<ADEN)| // ADC ENABLE
55. (0<<ADSC)| // ADC START CONVERSION
56. (0<<ADATE)| // ADC AUTO TRIGGER ENABLE
57. (0<<ADIF)| // ADC INTERRUPT ENABLE
58. (0<<ADIE)|// ADC PRESCALER SELECT BIT
59. (1<<ADPS2)|
60. (1<<ADPS1)|
61. (1<<ADPS0); // SELECT CHANNEL
62. }
63. void read\_adc(void)
64. {
65. unsigned char i = 4;
66. adc\_temp = 0;
67. while(i--)
68. {
69. ADCSRA |=(1<<ADSC);
70. while(ADCSRA & (1<<ADSC));
71. adc\_temp += ADC;
72. *\_delay\_ms*(50);
73. }
74. adc\_temp = (adc\_temp/4);
75. }
76. int main(void)
77. {
78. usart\_init();
79. set();
80. adc\_init();
82. char c[256];
83. usart\_tx\_string(direct1);
84. *\_delay\_ms*(6000);
85. usart\_tx\_string(direct2);
86. *\_delay\_ms*(6000);
87. usart\_tx\_string(direct3);
88. *\_delay\_ms*(6000);
89. usart\_tx\_string(direct4);
90. *\_delay\_ms*(6000);
91. usart\_tx\_string(direct5);
92. *\_delay\_ms*(6000);
93. usart\_tx\_string(direct6);
94. *\_delay\_ms*(6000);
95. usart\_tx\_string(direct7);
96. *\_delay\_ms*(6000);
97. read\_adc(); // this will read the adc
98. *snprintf*(c,sizeof(c),"GET//api.thingspeak.com/update?api\_key=FRW23XEXSTDAH6SN&field1=%3d\r\n", adc\_temp); // this will print out the float
99. usart\_tx\_string(c); // this will output on the terminal
100. TCNT1 =0x00; // this will reset the timer
101. usart\_init();
102. set();
103. adc\_init();

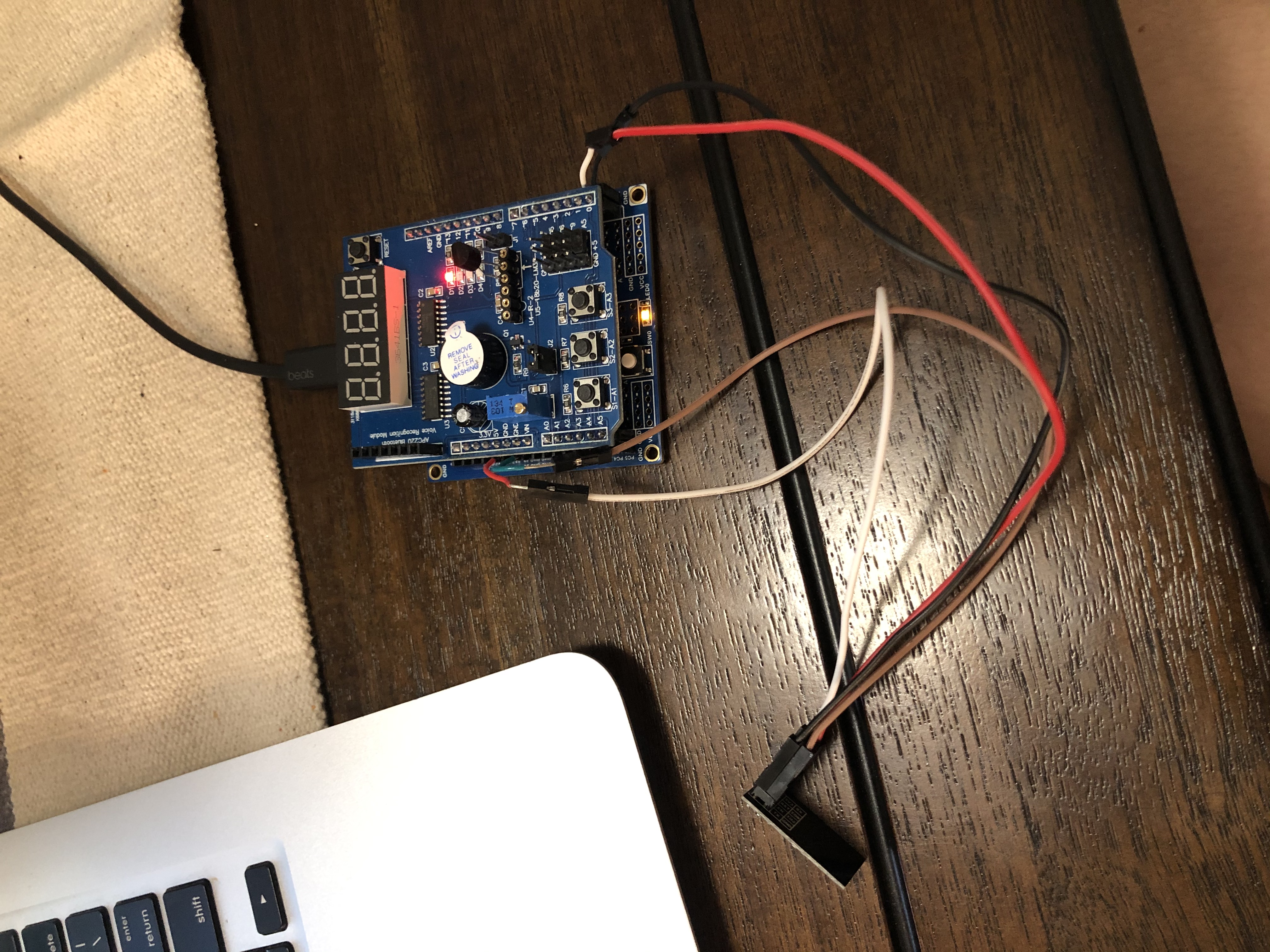
106. while(1)
107. {
108. while(of < 4);
109. read\_adc();
110. *snprintf*(c,sizeof(c), "%f\r\n", adc\_temp);
111. usart\_tx\_string(c);
112. usart\_tx\_string("\n");
113. of = 0;
114. }
115. }
116. **SCHEMATICS**



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



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



1. **VIDEO LINKS OF EACH DEMO**
2. **GITHUB LINK OF THIS DA**

**Student Academic Misconduct Policy**

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

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

NAME OF THE STUDENT