Guillermo Gálvez

CPE301 – SPRING 2018

Design Assignment 3

**DO NOT REMOVE THIS PAGE DURING SUBMISSION:**

The student understands that all required components should be submitted in complete for grading of this assignment.

|  |  |  |  |
| --- | --- | --- | --- |
| **NO** | **SUBMISSION ITEM** | **COMPLETED (Y/N)** | **MARKS**  **(/MAX)** |
| 1 | COMPONENTS LIST AND CONNECTION BLOCK DIAGRAM w/ PINS |  |  |
| 2. | CODE OF TASK 1/A |  |  |
| 3. | SCHEMATICS |  |  |
| 4. | Flow Chart |  |  |
| 5. | SCREENSHOTS OF EACH TASK OUTPUT/BOARD SETUP |  |  |
| 6. | VIDEO LINKS OF EACH DEMO |  |  |
| 7. | Git Link |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

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

LM34 Temperature sensor

Bread Board

ESP8266-01

328P Xplained Mini

Jumper wires

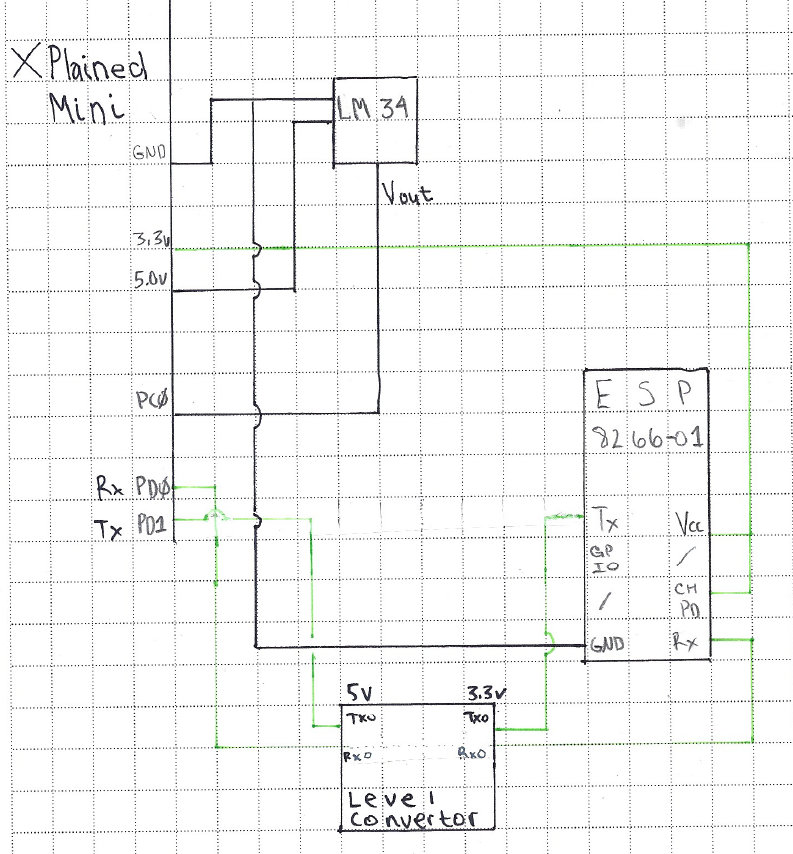
1. **DEVELOPED CODE OF TASK 1/A**
2. */\**
3. *\* Midterm\_galveg1.c*
4. *\**
5. *\* Created: 4/7/2018 12:23:56 PM*
6. *\* Author : Guillermo Gálvez*
7. *\*/*
8. #include <stdlib.h>
9. #include <avr/io.h>
10. #include <avr/interrupt.h>
11. #include <util/delay.h>
12. #include <stdio.h>
13. #include <stdint.h>
14. #define F\_CPU 8000000UL
15. #define FOSC 16000000 *//Frequency*
16. #define BAUD 9600 *//Baud Rate*
17. #define MYUBRR FOSC/16/BAUD-1 *//Automatic BAUD rate calculation*
18. **volatile** uint8\_t adcValue;
19. */\*~1.7min timer1 Counter 65536\*12 \* T = ~1.7 minutes\*/*
20. **volatile** uint8\_t fifteenPlus = 12; *//*
21. */\*Function Declarations\*/*
22. void USART\_Init();
23. void ADC\_Init();
24. void outputChr(unsigned char c);
25. void outputStr(char \*c);
26. void readTemp();
27. void AT\_Tx(char \*t);
28. *//void append(unsigned char \*one, unsigned char \*two);*
29. */\*Interrupt Service Routines\*/*
30. ISR(ADC\_vect)
31. {
32. ADCSRA |= (1 << ADIF); *//Reset flag*
33. adcValue = ADCH; *//MSB 8-bits of ADC form left shift of ADLAR*
34. }*//end ISR(ADC\_vect(*
35. ISR(TIMER1\_OVF\_vect)
36. {
37. TIFR1 |= (1 << TOV1); *//Clr Flag*
38. fifteenPlus++;
40. char seeTemp[8];
41. float lm34\_0; *//For ASCII Temp output*
43. **while**((ADCSRA & (1 << ADIF)) == 0); *//Wait for conversion to finish*
45. */\*Conversion to °F\*/*
46. lm34\_0 = (adcValue \* 5.0 / 0x100) \* 100.0; *//(ADC \* 5 = 200 /256) \* 100*
48. dtostrf(lm34\_0, 5, 2, seeTemp); *//Float to char conversion*

51. **if**(fifteenPlus>=12) *//Timer*
52. {
54. */\*Build Strings for AT+ commands\*/*
55. unsigned char CIPStart[] = "AT+CIPSTART=**\"**TCP**\"**,**\"**api.thingspeak.com**\"**,80**\r\n**";
56. unsigned char CIPSend[] = "AT+CIPSEND=51**\r\n**";
57. unsigned char Data[] = "GET /update?api\_key=X5GNOJ6AFIEO0XRP&field1=";
58. unsigned char temp0 = "t";
59. unsigned char temp1 = "**\r\n**";
60. unsigned char CIPClose[] = "AT+CIPCLOSE**\r\n**";

63. \_delay\_ms(2000);
64. outputStr(CIPStart); *//Send Start String*
66. \_delay\_ms(2000);
67. outputStr(CIPSend); *//Number of Char being sent*
69. \_delay\_ms(2000);
70. outputStr(Data); *//Get command sent*
71. outputStr(seeTemp); *//Temperature string added to end of Get command*
72. outputStr("**\n\r**"); *//Enter key*
74. fifteenPlus = 0; *//Reset timer*
76. }*//end if*
78. }*//end ISR(TIMER1\_OVF\_vect)*
79. */\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**
80. *\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**
81. *\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**
82. *MAIN\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**
83. *\*/*
84. int main(void)
85. {
86. TCCR1B |= (1 << CS12) | (1 << CS10); *//Set prescale 1024*
87. TIMSK1 |= (1 << TOIE1); *//enable OVF interrupt*
89. unsigned int oneFive = 0;
91. ADC\_Init(); *//initialize ADC*
92. USART\_Init(); *//UART initialization*

95. **while** (1)
96. {
98. }*//end while(1)*
100. }*//end main*
101. */\*USART initialization function\*/*
102. void USART\_Init()
103. {
105. */\*Set Baud Rate\*/*
106. UBRR0H = (MYUBRR>>8); *//Shift MSB "top" of UBRR0H 0100 0100 >> 8 -> UBRR0H 0000 0000*
107. UBRR0L = MYUBRR; *//UBRR0L 0100 0100*
109. UCSR0B |= (1 << RXEN0) | (1 << TXEN0); *//Enable Rec and Trans*
110. UCSR0B |= (1 << RXCIE0); *//Enable Rec INT*
111. UCSR0C |= (1 << UCSZ01) | (1 << UCSZ00); *//Set frame 8-bit, 1 STP*
112. }*//end USART\_int*
113. */\*ADC initialization function\*/*
114. void ADC\_Init()
115. {
116. DDRC = 0; *//Set PORTC as input for adc*
117. DIDR0 = 0x01; *//Disable Digi input on ADC0*
119. ADMUX = 0; *//Sets Mux selection bits to 0 ADC0 used*
120. ADMUX |= (1 << REFS0); *//Use Vcc Ref voltage selectin 01*
121. ADMUX |= (1 << ADLAR); *//Left adjust ADC Reg, ADCH 8-bit Resolution*
123. *//Enable ADC, Start Conv, Set Auto Trig Enable*
124. ADCSRA |= (1 << ADEN); *//Enable ADC*
125. ADCSRA |= (1 << ADATE); *//Set ADC Auto Trig*
126. ADCSRA |= (1 << ADIE); *//Enable Interrupts*
127. ADCSRA |= (1 << ADPS2) | (1 << ADPS1) | (0 << ADPS0); *//8MHz, Pre-Scale 64 = 125KHz*
128. ADCSRA |= (1 << ADSC); *//Start Conversion*
130. ADCSRB = 0; *//Free running mode*
131. sei(); *//Enable interrupts*
133. }*//end ADC\_Init*
134. */\*Character print function\*/*
135. void outputChr(unsigned char c)
136. {
137. UDR0 = c; *//Display Char on Serial*
138. \_delay\_ms(800);
139. }*//end outputChr*
140. */\*String print function\*/*
141. void outputStr(char \*c)
142. {
143. unsigned int i = 0; *//loop control*
144. **while**(c[i] != 0)
145. outputChr(c[i++]);
146. }*//end outpuStr*
147. **SCHEMATICS**

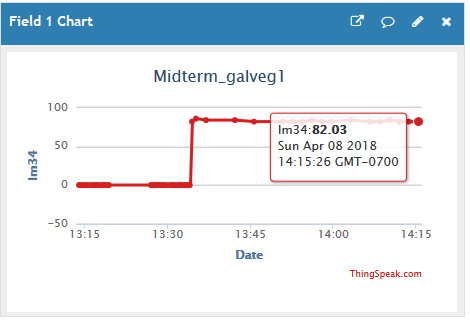
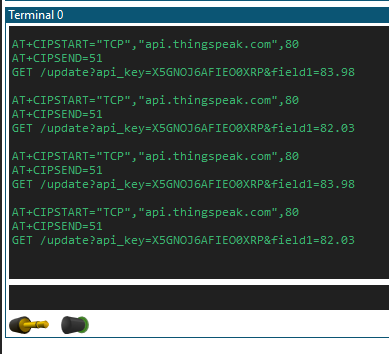
Use fritzing.org 🡨 Freezes

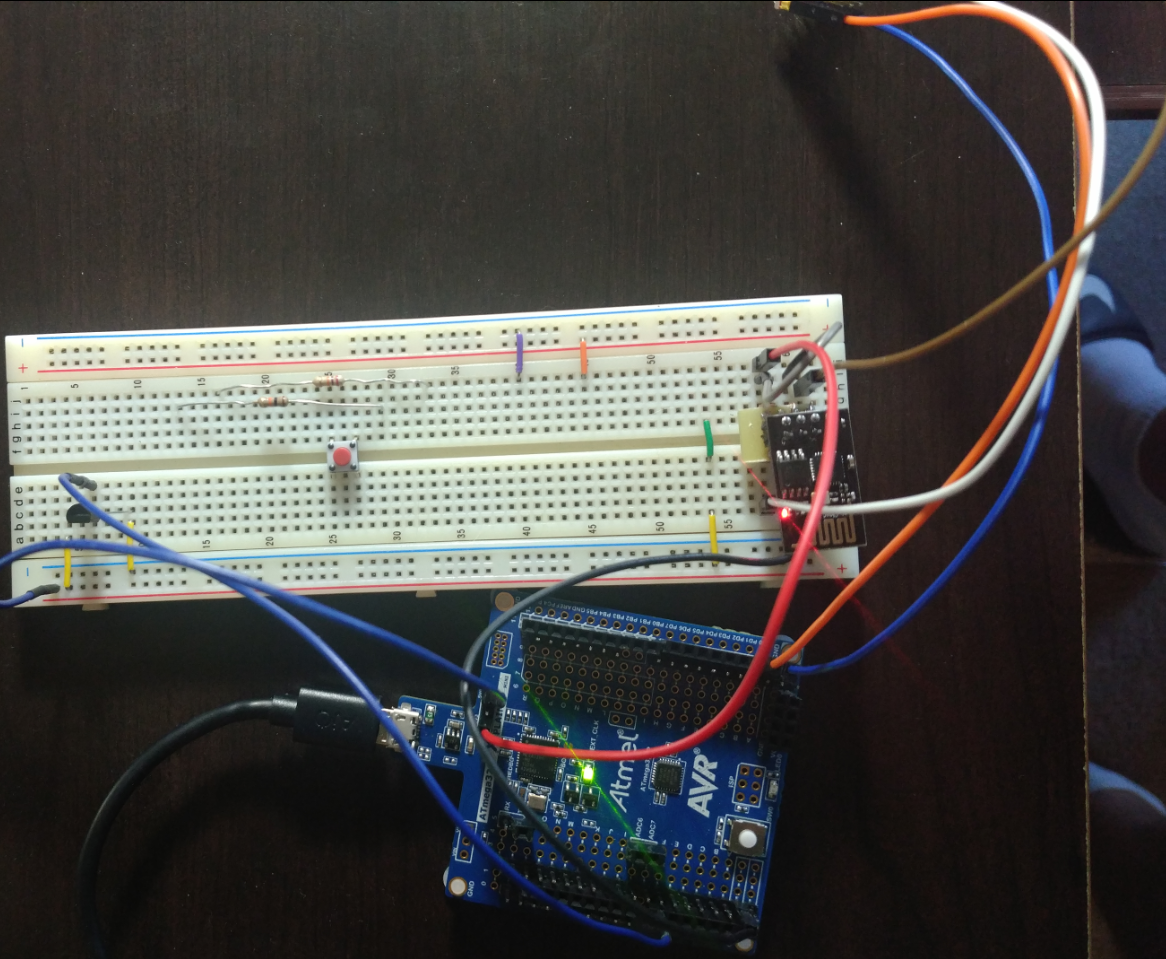


1. **FLOWCHART**

Wait for OVF

1. **SCREENSHOT OF (ATMEL STUDIO OUTPUT)/ (BOARD SETUP)**





1. **VIDEO LINKS OF EACH DEMO**

For video modified code to update faster changed back

<https://youtu.be/Rp3zeCU4gQw>

1. **GITHUB LINK OF THIS DA**

**Student Academic Misconduct Policy**

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

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

Guillermo Gálvez