Midterm1: IoT Due Date: See Website

Q: Write, simulate, and demonstrate using AVRStudio6/7 an C code for the AVR ATMEGA328p microcontroller that performs the following functions:

1. Program the ADC of ATmega328/p to read the LM34/35 temperature sensor.
2. Display the value to UART.
3. Make sure the AT Firmware is downloaded into the ESP8266-01 module.
4. Register for a free Thingspeak account with MATHWORK. Setup and get the channel Key.
5. Transmit temperature sensor value to ESP8266-01 through UART port using AT Commands.
6. Display the temperature sensor value as a graph in Thingspeak

Submission:

The following are required for successful completion of the design assignment:

a. AVR C code that has been assembled and working.
b. The C code should be well documented with explanation of every instruction.
c. A word document that contains the flow chart of the assembly code along with the screenshots/snapshot of the AVRStudio6/7 and/or live connections during debugging at the beginning and end of Task 1-6.
d. Submit one solution folder, with doc and video/snapshot file

Points:
Task 1: 25%, Task 2:15%, Task 3-6: 60%.

Helpers:
Flash AT-Firmware on ESP-12E.pdf
AVR to ESP-12E.pdf

Code for internal temperature sensor:

```
unsigned int Ctemp;
unsigned int Ftemp;

int main(void)
{

    /* Setup ADC to use int 1.1V reference
    and select temp sensor channel */
    ADMUX = (1<<REFS1) | (1<<REFS0) | (0<<ADLAR) | (1<<MUX3) | (0<<MUX2) | (0<<MUX1)
| (0<<MUX0);

    /* Set conversion time to
    112usec = [(1/(8Mhz / 64)) * (14 ADC clocks  per conversion)]
     and enable the ADC*/
    ADCSRA = (1<<ADPS2) | (1<<ADPS1) | (1<<ADEN);

    /* Perform Dummy Conversion to complete ADC init */
    ADCSRA |= (1<<ADSC);

    /* wait for conversion to complete */
    while ((ADCSRA & (1<<ADSC)) != 0);

    /* Scan for changes on A/D input pin in an infinite loop */
    while(1)
    {
        /* start a new conversion on channel 8 */
        ADCSRA |= (1<<ADSC);

        /* wait for conversion to complete */
        while ((ADCSRA & (1<<ADSC)) != 0)
        ;

        /* Calculate the temperature in C */
        Ctemp = (ADC - 247)/1.22;
        Ftemp = (Ctemp * 1.8) + 32;
    }

    return -1;

}
```

Code for UART - ESP8266-01 interface:

```
…
   unsigned char AT[] = "AT\r\n";
   unsigned char CIPMUX[] = "AT+CIPMUX=1\r\n";
   unsigned char CIPSTART[] =  "AT+CIPSTART=0,\"TCP\",\"api.thingspeak.com\",80\r\n";
   unsigned char CIPSEND[] = "AT+CIPSEND=0,110\r\n";
    unsigned char GET_DATA[] = "GET https://api.thingspeak.com/apps/thinghttp/send_request?
api_key=xxxxxxxxxxxxxxxx\r\n";
          unsigned char SEND_DATA[] = "GET https://api.thingspeak.com/update?
api_key=xxxxxxxxxxxxxxxxxxxxxxx=50\r\n";

   //_delay_ms(200);
   usart_init(); // initialize usart
   _delay_ms(500);
   sei();

   _delay_ms(200);
   send_AT(AT);

   _delay_ms(2000);
   send_AT(CIPMUX);

   _delay_ms(2000);
   send_AT(CIPSTART);

   while(1)
   {

   }//while(1) close

…
```