Yannick Kengne Tatcha

CPE301 – SPRING 2018

Midterm

**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 |  |  |
| 2. | TASK 1 & 2 COMPLETE CODE W/ COMMENTS AND SCREENSHOTS |  |  |
| 3. | AT FIRMWARE DOWNLOAD AND VERSION CONFIRMATION |  |  |
| 4. | THINGSPEAK ACCOUNT AND CHANNEL KEY |  |  |
| 5. | FLASH AND TRANSMISSION |  |  |
| 6. | THINGSPEAK GRAPH |  |  |

1. **COMPONENTS LIST**

Needed:

FTDI232R chip

ATMEGA328P Microcontroller

LM34 Sensor

ESP8266-01

1. **TASK 1 & 2 COMPLETE CODE W/ COMMENTS AND SCREENSHOTS**

/\*

\* midterm1.c

\*

\* Created: 4/11/2018 11:09:31 AM

\* Author : YKengne

\*/

#include <stdlib.h>

#include <avr/io.h>

#include <avr/interrupt.h>

#include <util/delay.h>

#include <stdio.h>

#include <stdint.h>

#define *F\_CPU* 8000000UL

#define FOSC 16000000 //Frequency

#define BAUD 9600 //Baud Rate

#define MYUBRR FOSC/16/BAUD-1 //Automatic BAUD rate calculation

volatile *uint8\_t* adcValue;

volatile *uint8\_t* fifteenPlus = 120; //

/\*Function Declarations\*/

void USART\_Init();

void ADC\_Init();

void outputChr(unsigned char c);

void outputStr(char \*c);

void readTemp();

void AT\_Tx(char \*t);

ISR(ADC\_vect)

{

ADCSRA |= (1 << ADIF); //Reset flag

adcValue = ADCH; //MSB 8-bits of ADC form left shift of ADLAR

}

ISR(TIMER1\_OVF\_vect)

{

TIFR1 |= (1 << TOV1); //Clr Flag

fifteenPlus++;

}

int main(void)

{

TCCR1B |= (1 << CS12) | (1 << CS10); //Set prescale 1024

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

unsigned int oneFive = 0;

ADC\_Init(); //initialize ADC

USART\_Init(); //UART initialization

while (1)

{

if(fifteenPlus>=120)

{

readTemp();

fifteenPlus = 0;

}

}

}

void USART\_Init()

{

UBRR0H = (MYUBRR>>8); //Shift MSB "top" of UBRR0H 0100 0100 >> 8 -> UBRR0H 0000 0000

UBRR0L = MYUBRR; //UBRR0L 0100 0100

UCSR0B |= (1 << RXEN0) | (1 << TXEN0); //Enable Rec and Trans

UCSR0B |= (1 << RXCIE0); //Enable Rec INT

UCSR0C |= (1 << UCSZ01) | (1 << UCSZ00); //Set frame 8-bit, 1 STP

}

void ADC\_Init()

{

DDRC = 0; //Set PORTC as input for adc

DIDR0 = 0x01; //Disable Digi input on ADC0

ADMUX = 0; //Sets Mux selection bits to 0 ADC0 used

ADMUX |= (1 << REFS0); //Use Vcc Ref voltage selectin 01

ADMUX |= (1 << ADLAR); //Left adjust ADC Reg, ADCH 8-bit Resolution

ADCSRA |= (1 << ADEN); //Enable ADC

ADCSRA |= (1 << ADATE); //Set ADC Auto Trig

ADCSRA |= (1 << ADIE); //Enable Interrupts

ADCSRA |= (1 << ADPS2) | (1 << ADPS1) | (0 << ADPS0); //8MHz, Pre-Scale 64 = 125KHz

ADCSRA |= (1 << ADSC); //Start Conversion

ADCSRB = 0; //Free running mode

sei(); //Enable interrupts

}

void outputChr(unsigned char c)

{

UDR0 = c; //Display Char on Serial

*\_delay\_ms*(800);

}

void outputStr(char \*c)

{

unsigned int i = 0; //loop control

while(c[i] != 0)

outputChr(c[i++]);

}

void readTemp()

{

char seeTemp[8];

float lm34\_0; //For ASCII Temp output

float lm34\_1; //For showing valued of ADC

while((ADCSRA & (1 << ADIF)) == 0); //Wait for conversion to finish

/\*Conversion to °F\*/

lm34\_0 = (adcValue \* 5.0 / 0x100) \* 100.0; //(ADC \* 5 = 200 /256) \* 100

*dtostrf*(lm34\_0, 5, 2, seeTemp); //Float to char conversion

AT\_Tx(seeTemp);

}

void AT\_Tx(char \*t)

{

/\*Build Strings for AT+ commands\*/

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

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

unsigned char Data[] = "GET /update?api\_key=X5GNOJ6AFIEO0XRP&field1=";

unsigned char temp0 = "t";

unsigned char temp1 = "\r\n";

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

*\_delay\_ms*(2000);

outputStr(CIPStart); //Send Start String

*\_delay\_ms*(2000);

outputStr(CIPSend); //Number of Char being sent

*\_delay\_ms*(2000);

outputStr(Data); //Get command sent

outputStr(t); //Temperature string added to end of Get command

outputStr("\n\r"); //Enter key

1. **Schematic**

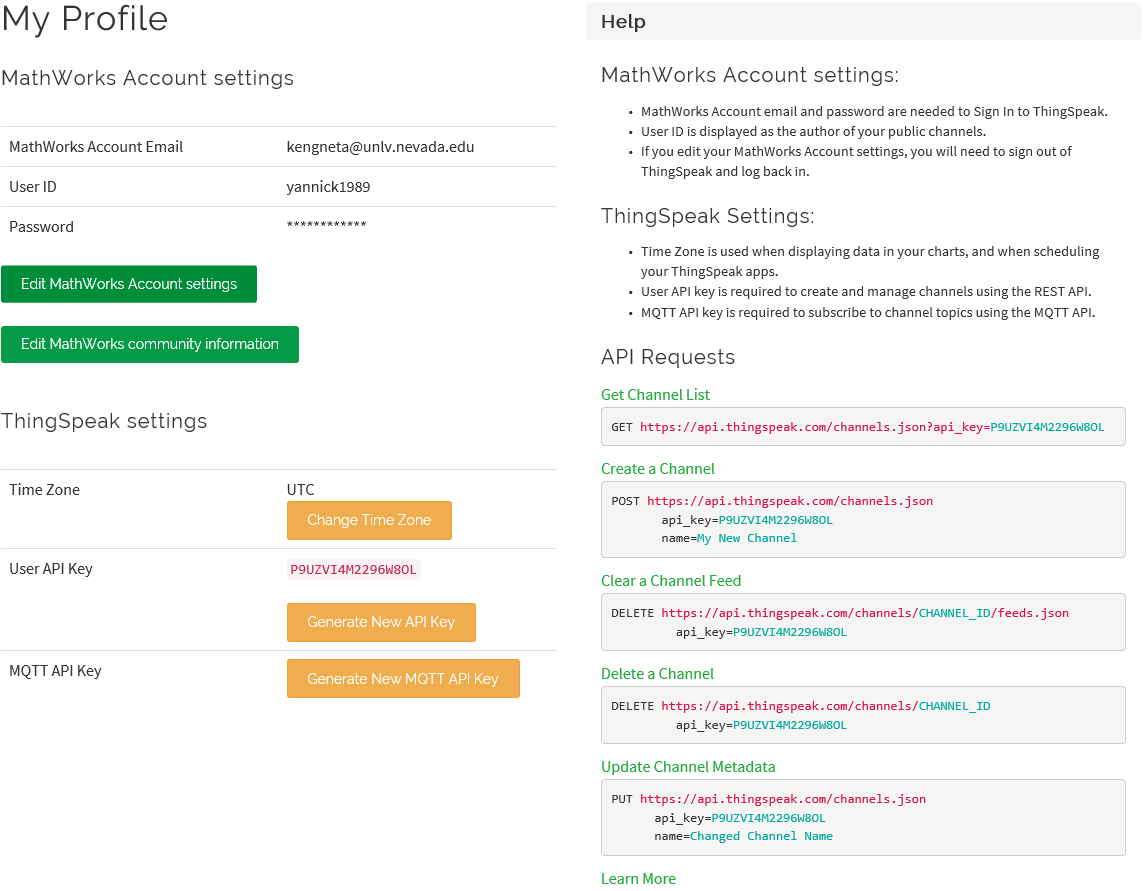


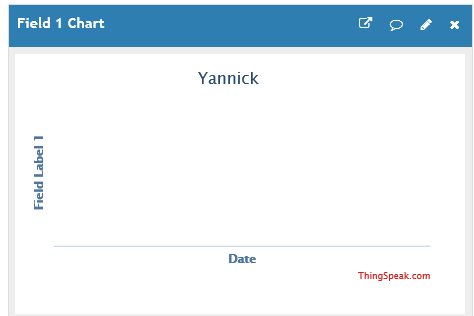
1. **Flowchart**

Timer1

Read temperature

1. **THINGSPEAK ACCOUNT AND CHANNEL KEY**





1. **GITHUB LINK**

[**https://github.com/Vasty1995/CPE301**](https://github.com/Vasty1995/CPE301)

**Student Academic Misconduct Policy**

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

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

Yannick Kengne Tatcha