Monty Sourjah

CPE301 – SPRING 2018 ID#1007139171

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 | Y |  |
| 2. | INITIAL CODE OF TASK 1/A | Y |  |
| 3. | INCREMENTAL / DIFFERENTIAL CODE OF TASK 2/B | N/A |  |
| 3. | INCREMENTAL / DIFFERENTIAL CODE OF TASK 3/C | N/A |  |
| 3. | INCREMENTAL / DIFFERENTIAL CODE OF TASK 4/D | N/A |  |
| 3. | INCREMENTAL / DIFFERENTIAL CODE OF TASK 5/E | N/A |  |
| 4. | SCHEMATICS | y |  |
| 5. | SCREENSHOTS OF EACH TASK OUTPUT | Y |  |
| 5. | SCREENSHOT OF EACH DEMO | y |  |
| 6. | VIDEO LINKS OF EACH DEMO | Y |  |
| 7. | GOOGLECODE LINK OF THE DA | Y |  |
|  |  |  |  |
|  |  |  |  |

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

List of Components used:LM34,ATmega 328p X plained mini board, Jumper wires ,USB cable, Analog Thermometer.

Block diagram with pins used in the Atmega328P

PC

ATmega 328p Xplained mini Board

LM34

USB cable

GND

VCC=5V

Analog Signal Vcc=5 GND

PC0

\*\*\*\*\*\*\*\*See attached excel file for the flow chart in GitHub\*\*\*\*\*\*\*

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

#include <avr/io.h>

#include <stdint.h> // needed for uint8\_t

#include <stdio.h>

#include <string.h>

#include <avr/interrupt.h>

#include <util/delay\_basic.h>

#define FOSC 16000000 // Clock Speed

#define BAUD 9600

#define MYUBRR FOSC/16/BAUD - 1

volatile *uint8\_t* ADCvalue; // Global variable, set to volatile if used with ISR

volatile char ReceivedChar;

int USART0SendByte(char u8Data);

char printBuffer[128];

double temp=0.0;

void Wait()

{

*uint8\_t* i;

// Delay will be 61\*(262.1/16) = 995.44375 miliseconds . See description of function \_delay\_loop\_2

for(i=0;i<61;i++)

{

*\_delay\_loop\_2*(0);

}

}

void adc\_init(void)

{

ADMUX = 0; // use ADC0

ADMUX |= (1 << REFS0); // use AVcc as the reference

ADMUX |= (1 << ADLAR); // Left adjust for 8 bit resolution

ADCSRA |= (1 << ADPS2) | (1 << ADPS1) | (1 << ADPS0); // 128 pre-scale for 16Mhz

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

ADCSRB = 0; // 0 for free running mode

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

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

ADCSRA |= (1 << ADSC); // Start the ADC conversion

}

void usart\_init()

{

UBRR0H = (MYUBRR) >> 8;

UBRR0L = MYUBRR;

UCSR0B |= (1 << RXEN0) | (1 << TXEN0); // Enable receiver and transmitter

UCSR0B |= (1 << RXCIE0); // Enable receiver interrupt

UCSR0C |= (1 << UCSZ01) | (1 << UCSZ00); // Set frame: 8data, 1 stop

}

int main(void)

{

adc\_init();

usart\_init();

sei();

while (1)

{

/\*

Since we have only read 8-bit from the left justified ADCH, and not the the full ADC register, the precision

is 256 instead of the full 1024 bit.

Also from the LM34 DataSheet Vout = 10.0 mV per degree Fahrenheit.

We can then obtain the temperature using the expression below:

\*/

temp = ((float)ADCvalue \* 5 / 256) / 0.010;

*sprintf*(printBuffer, "Temperature is %0.2f Fahrenheit\n", temp);

for (int i=0; i<*strlen*(printBuffer);i++)

{

USART0SendByte(printBuffer[i]);

}

Wait(); //Wait approximately 1 second.

}

}

ISR(ADC\_vect)

{

ADCvalue = ADCH; // only need to read the high value for 8 bit

}

ISR (USART\_RX\_vect)

{

ReceivedChar = UDR0; // Read data from the RX buffer

UDR0 = ReceivedChar; // Write the data to the TX buffer

}

int USART0SendByte(char u8Data)

{

//wait while previous byte is completed

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

// Transmit data

UDR0 = u8Data;

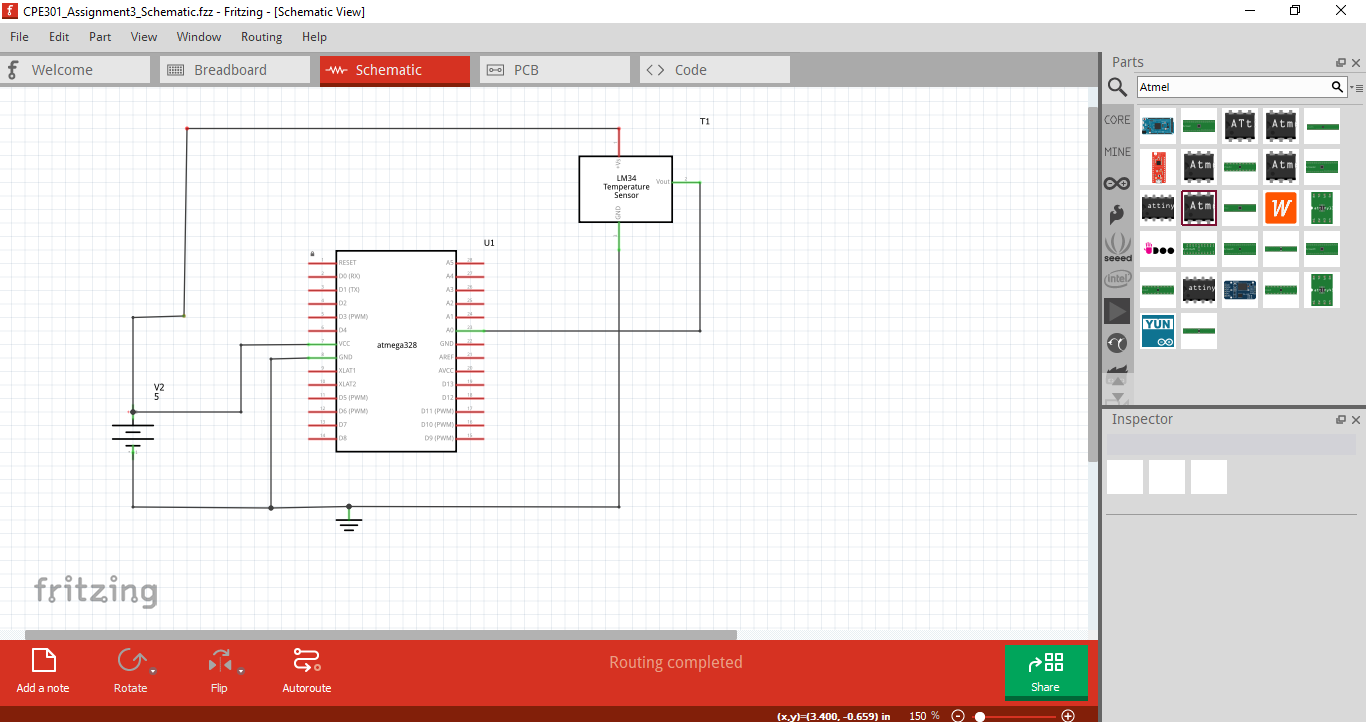
return 0;

}

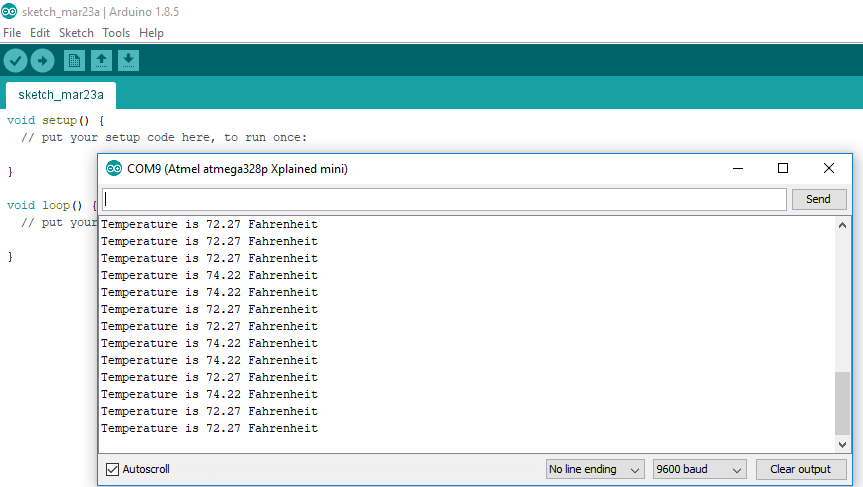
1. **MODIFIED CODE OF TASK 2/A from TASK 1/A**

N/A

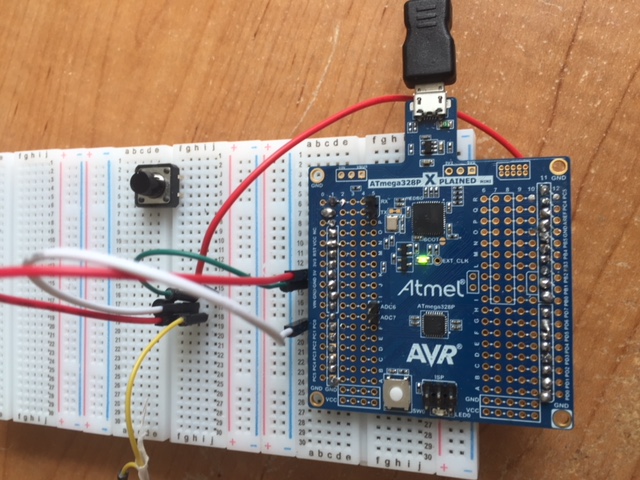
1. **N/A**
2. **N/A**
3. **N/A**
4. **SCHEMATICS**



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



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



1. **VIDEO LINKS OF EACH DEMO**

<http://www.youtube.com/watch?v=aGnm9L9GxR4>

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”.

NAME OF THE STUDENT