Electrical, Computer & Energy Engineering
UNIVERSITY OF COLORADO BOULDER

ESM Lab Grading Report

Course Number: 1
Module: 2: Lab 1 on Thermistors

Lab Report Date 25th September 2018

Student Name(s) **Rushi James Macwan** and **Poorn Mehta**

Each Section of this Lab Report Counts for 20 points. Points will be allocated as follows:
20 points: section fully meets requirements of this rubric
15 points: section mostly meets requirements of this rubric
10 points: section meets roughly half the requirements of this rubric
5 points: section does not meet requirements, but shows a weak attempt
0 points: section blank

(A) **Functional demonstration of your circuit to our TA.** In this exercise, you schedule an appointment with your TA to show that your hardware functions as designed. For the thermistor lab, this will involve the following steps:

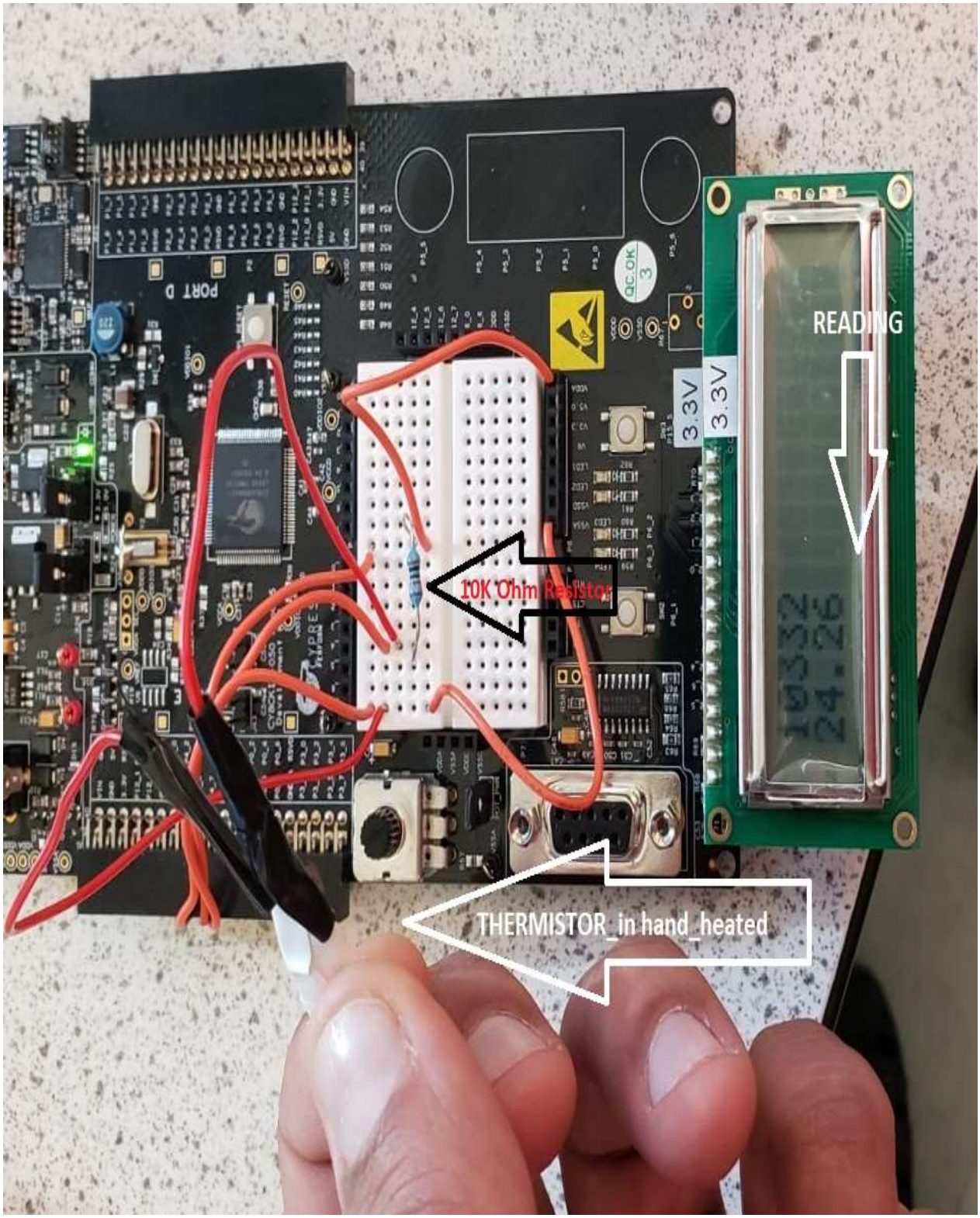
1. Show that all hardware is in place, and that your PSoC software can read the temperature of the thermistors.
2. Alter the temperature of the thermistor manually. This is done by blowing hot air over it for 2-3 seconds, holding it with your fingers for 15 seconds, or touching it with an object much colder than room temperature. Show that the thermistor reading on your LCD changes accordingly.

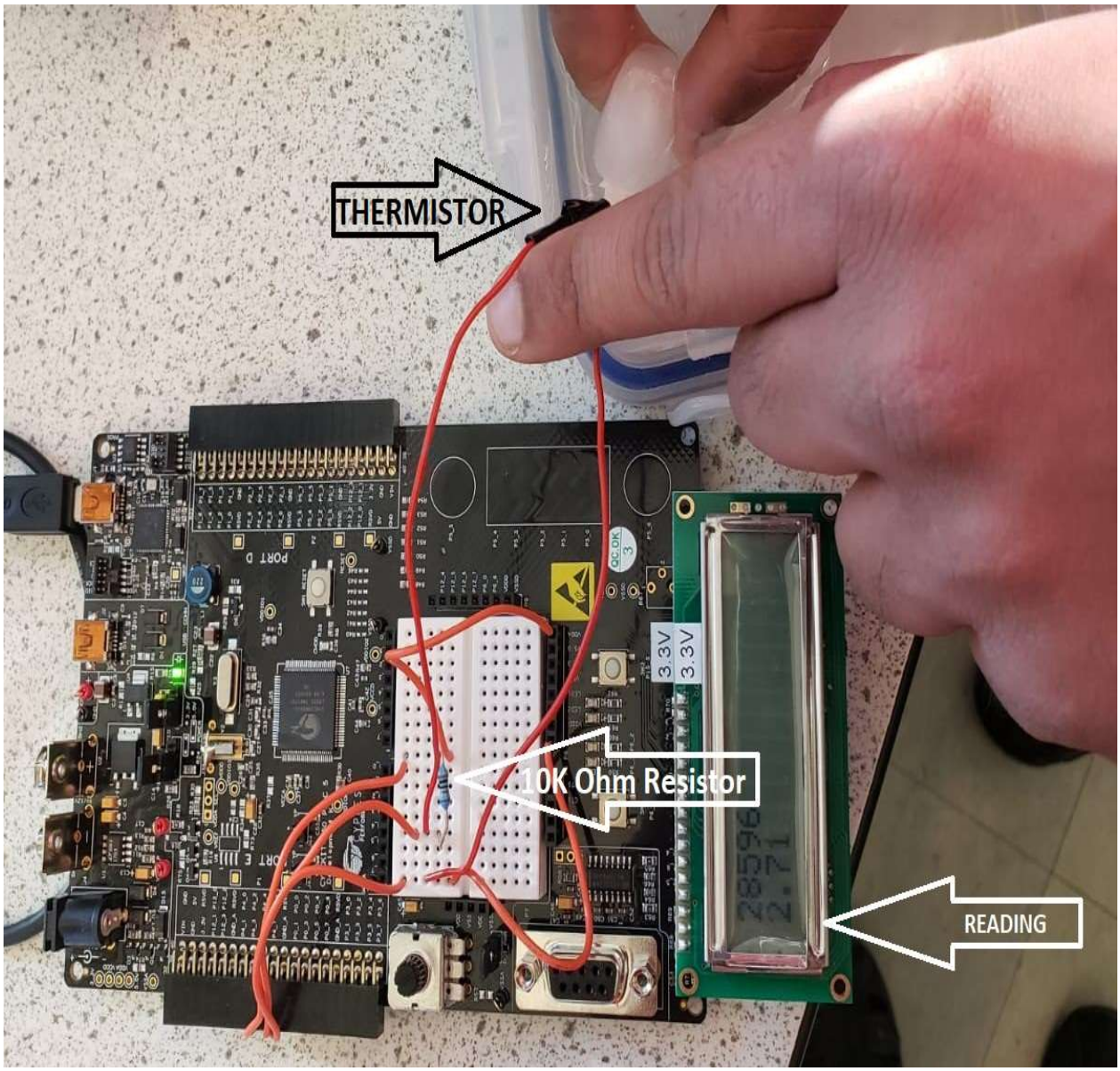
If you are an on-campus student, then show your circuit to one of our TA's during office hours.

If you are a distance learning student, make an online appointment with your TA to demonstrate your work via Zoom meeting or other Web-based meeting tool. You can use the camera on your laptop PC or suitable plug-in webcam (Logitech etc.) to demonstrate a working circuit.

(B) Place **photos here of your hardware setup, including PSoC board, connections to Oscilloscope or nScope, wiring, LCD Display, components**.
Label all components.

Sol.



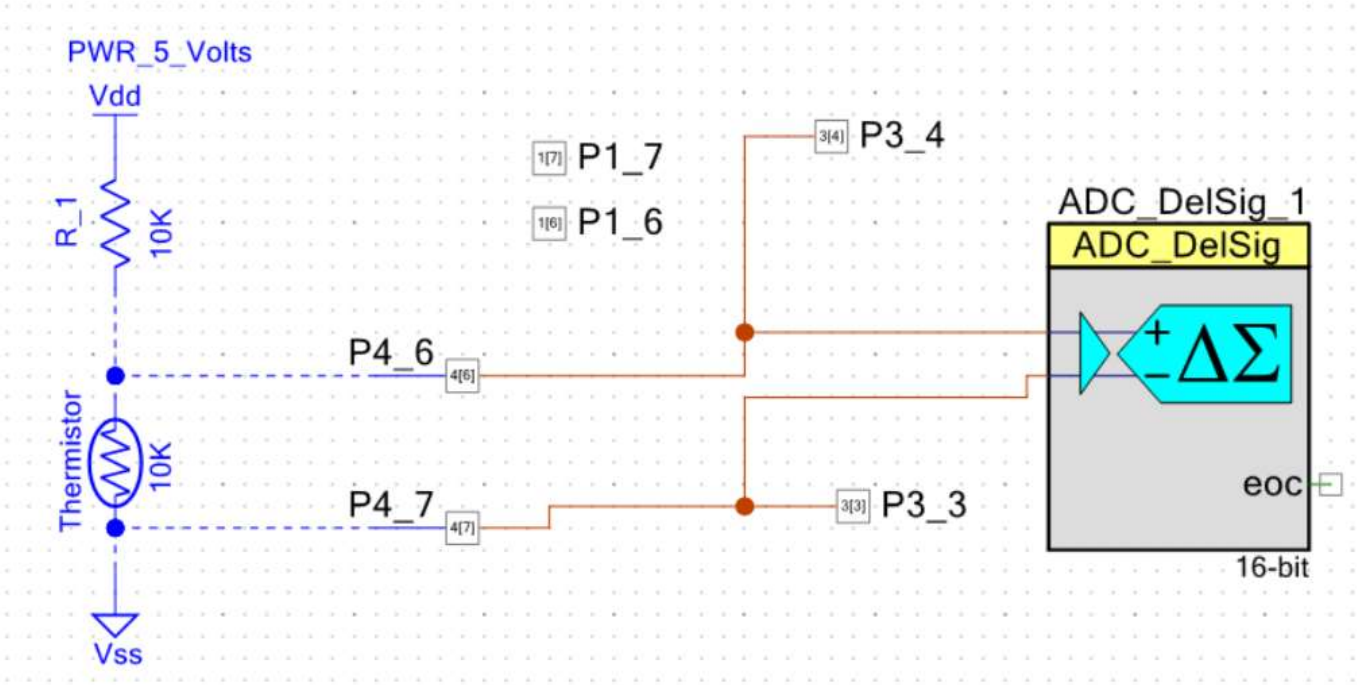


In the first image, the thermistor is subjected to heat in the form of the warmth of the hand touching it.

In the second image, the thermistor is subjected to very low temperatures by allowing it to be in close contact with a cube of ice.

The circuitry contains the reading available on the LCD, the wiring connecting the 10 K Ohms resistor and the thermistor to the PSOC board.

(C) **Place complete PSoC schematic here**. This schematic must include internal components from the PSoC board (amplifier, MUX, etc.), as well as external components (thermistor, resistor, display).



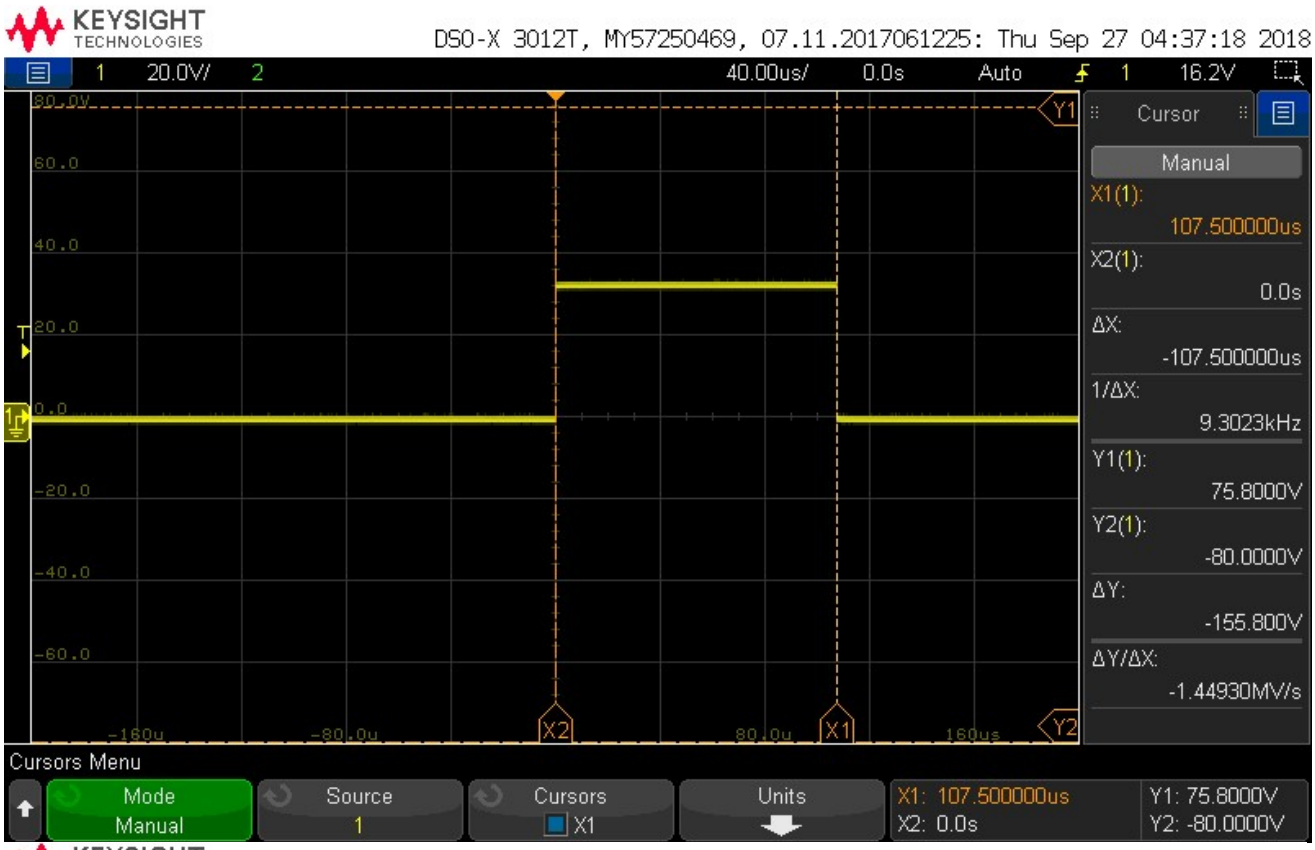
	Name	Port	Pin	Lock
	\LCD:LCDPort[6:0]\	P2[6:0]	2,1,99…95	☑
	P1_6	P1[6]	27	☑
	P1_7	P1[7]	28	☑
	P3_3	P3[3]	47	☑
	P3_4	P3[4]	48	☑
	P4_6	P4[6]	84	☑
	P4_7	P4[7]	85	☑

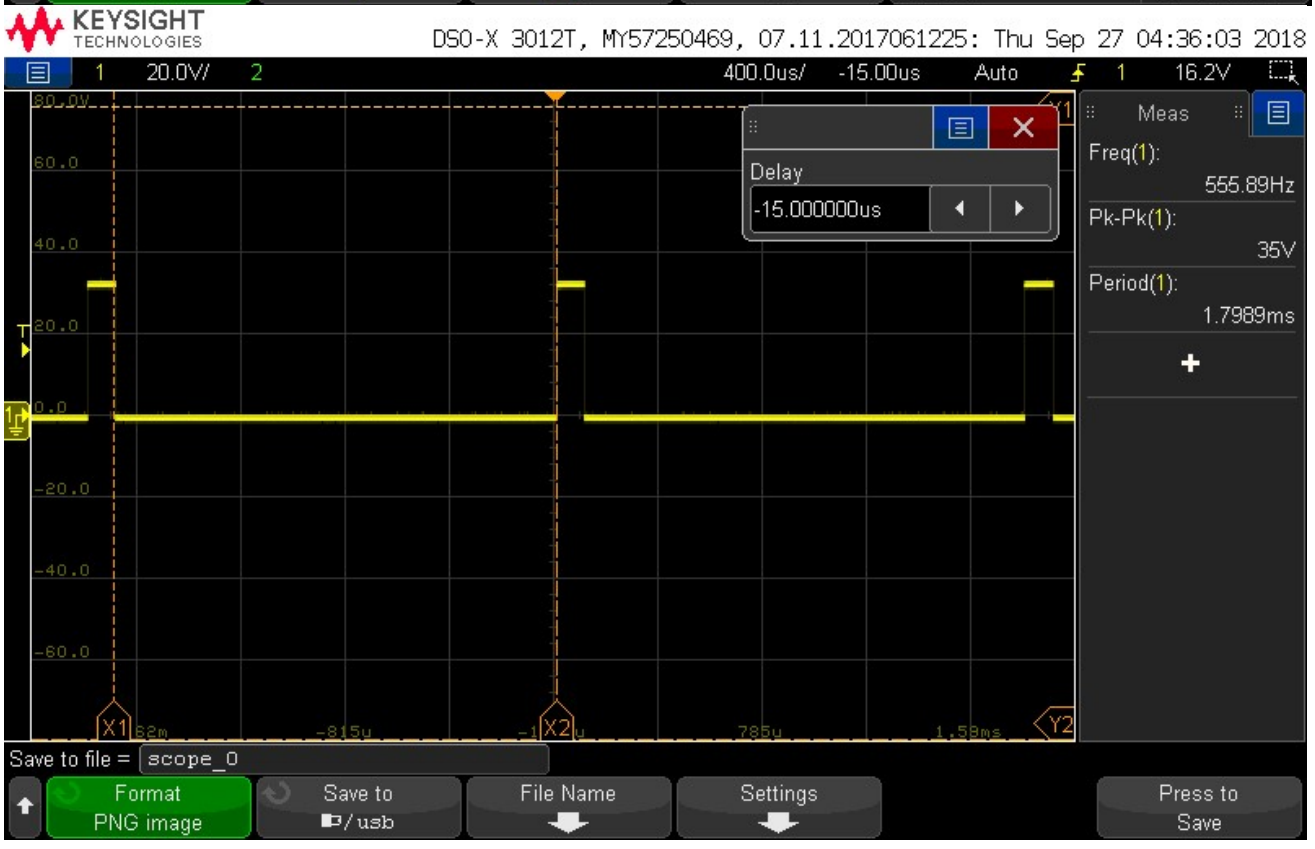
(D) **Place your complete PSoC software here**. This schematic must include calls to all internal functions, appropriate comments, and functional code that you included. We will not grade you on the exact syntax and structure, as there are numerous ways to structure the code and still provide the temperature measurement function. Instead, we will grade you on the completeness of the code relative to using the appropriate PSoC functions to gather the necessary data.

```
/* ========================================
 * Thermistor Project
 *
 * ========================================
*/
#include "project.h"
#include "math.h"
#include "float.h"

void Main_Init(void)
{
    ADC_DelSig_1_Start();
    LCD_Start();
    LCD_Position(0, 0);
    LCD_PrintString(" Thermistor");
    P1_6_Write(0);
    P1_7_Write(0);
}

uint16_t R1_Raw, R1_mV, Rt_Raw, Rt_mV, temperature, count;
float Current, Rt_val, Rt_val2, R1_mV_f, Rt_mV_f, xyz;


int main(void)
{
    CyGlobalIntEnable; /* Enable global interrupts. */

    Main_Init();
    while(1)
    {

        count += 1;
        P1_6_Write(1);
        ADC_DelSig_1_StartConvert();
        ADC_DelSig_1_IsEndConversion(ADC_DelSig_1_WAIT_FOR_RESULT);
        P1_6_Write(0);
        Rt_Raw = ADC_DelSig_1_GetResult16();
        Rt_mV = ADC_DelSig_1_CountsTo_mVolts(Rt_Raw);
        Rt_mV_f = Rt_mV;
        xyz = Rt_mV_f / (3300 - Rt_mV_f);
        xyz *= 10000;
        temperature = Thermistor_1_GetTemperature(xyz);
        CyDelay(1);
        if(count > 250)
```

```
        {
            P1_7_Write(1);
            LCD_ClearDisplay();
            LCD_Position(1, 0);
            LCD_PrintU32Number(temperature/100);
            if((temperature/100) < 10)
            {
                LCD_Position(1, 1);
                LCD_PrintString(".");
                LCD_Position(1, 2);
            }
            else
            {
                LCD_Position(1, 2);
                LCD_PrintString(".");
                LCD_Position(1, 3);
            }
            if((temperature%100) < 10)
            {
                LCD_PrintString("0");
            }
            LCD_PrintU32Number(temperature%100);
            count = 0;
            P1_7_Write(0);
        }
    }
}

/* [] END OF FILE */
```

E) **Place screenshots from your oscilloscope or nScope showing critical loop times or signal outputs**. For the thermistor lab, you can show loops that write to the LED, perform ADC conversion, and read the variable that records the temperature reading of the thermistor.


KEYSIGHT TECHNOLOGIES
DSO-X 3012T, MY57250469, 07.11.2017061225: Thu Sep 27 04:37:18 2018
1 20.0V/ 2
40.00us/ 0.0s Auto
1 16.2V
Cursor
Manual
X1(1):
107.500000us
X2(1):
0.0s
ΔX:
-107.500000us
1/ΔX:
9.3023kHz
Y1(1):
75.8000V
Y2(1):
-80.0000V
ΔY:
-155.800V
ΔY/ΔX:
-1.44930MV/s
Cursors Menu
Mode
Manual
Source
1
Cursors
X1
Units
X1: 107.500000us
X2: 0.0s
Y1: 75.8000V
Y2: -80.0000V


KEYSIGHT TECHNOLOGIES
DSO-X 3012T, MY57250469, 07.11.2017061225: Thu Sep 27 04:36:03 2018
1 20.0V/ 2
400.0us/ -15.00us Auto
1 16.2V
Meas
Delay
-15.000000us
Freq(1):
555.89Hz
Pk-Pk(1):
35V
Period(1):
1.7989ms
Save to file = scope_0
Format
PNG image
Save to
/usb
File Name
Settings
Press to
Save