California State University, Channel Islands (CSUCI)
Department of Computer Science

COMP-462: Embedded Systems
Lab Report
Fall 2019

Lab Number: Lab 2

Lab Topic:

Processing Temperature Sensor Data Using C

Student Name: Marc Julian Jamerlan
Student ID: 001956436
Student Major: B.S. Computer Science

I. Objectives

The objective of this lab is to familiarize students with using C code to manipulate and process data in order to produce a meaningful output. The goal is to create functions that will take data from a series of readings from a simulated temperature sensor, convert that data into relevant information and display it.

II. Introduction

This lab used starter code provided by the University of Texas' EE319K course, which is an implementation of a simulated microcontroller that gathers temperature data periodically. `main.c` contains an array of values called `Readings` that stores temperature data collected by the controller over an arbitrary period of time.

For this lab, the task was to implement three functions in `Lab2.c` that would be used by the controller in `main.c` to process the data in `Readings`. The three functions are as follows:

- `int16_t Find_Mean(Readings[], N):` takes the `Readings` array and the number of elements in the array `N` as parameters. Returns the average of the data in `Readings`.

- `int16_t FtoC(TinF):` Takes a given temperature in Fahrenheit and returns the temperature in Celsius.

- `int isMonotonic(Readings[], N):` Checks if the `Readings` array is an increasing monotonic series. Returns true if all the values in the array are increasing/equal and returns false if there is a decrease in value in the array.
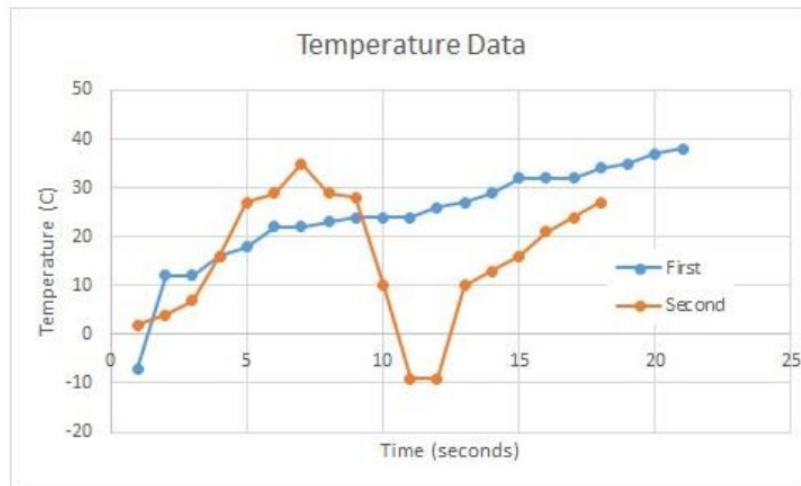

Figure 1: Increasing monotonic series (blue) vs. non-monotonic series (orange)

III. Procedure

1. Implemented `Find_Mean` by using a `for` loop to go through all the elements in the array and add them to a variable `sum`. `sum` was then divided by the number of elements `N` and returned by the function.

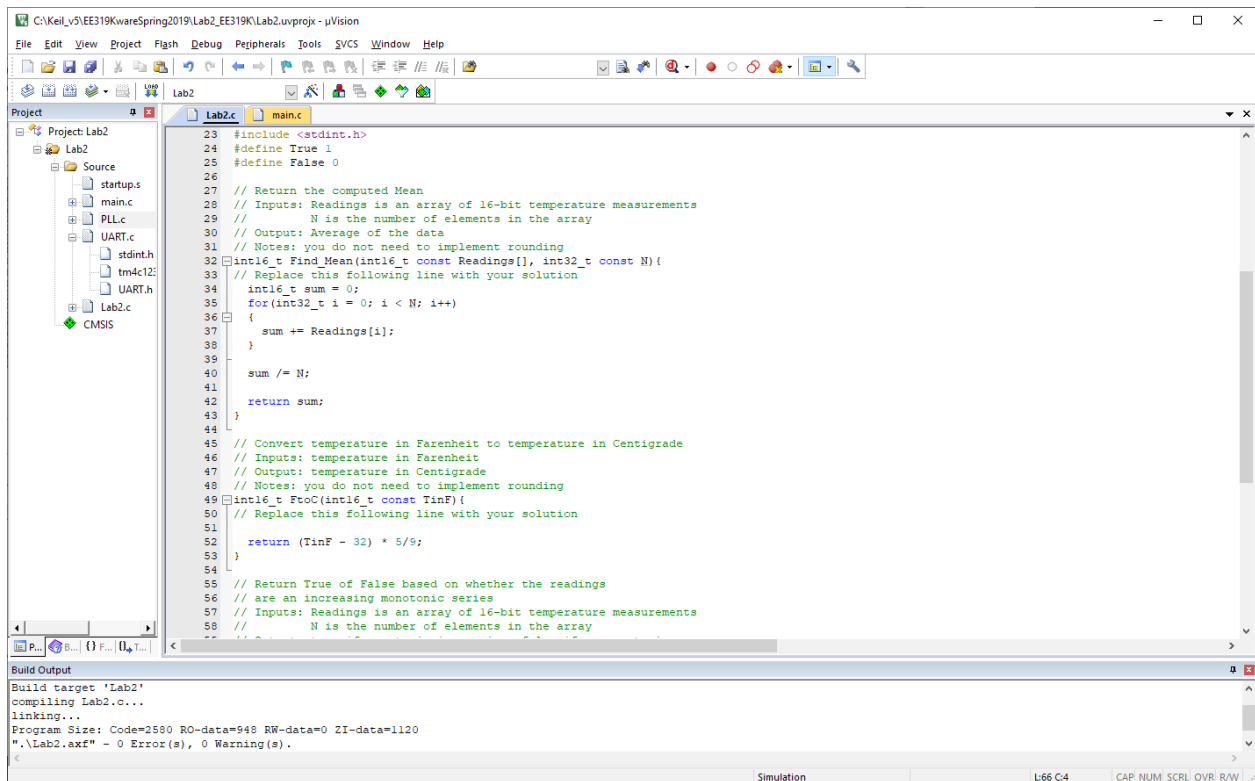2. Implemented `FtoC` by returning the result of the conversion equation from Fahrenheit to Celsius.

3. Implemented `isMonotonic` by using a `for` loop to go through all the elements in the array. A variable `prev` keeps track of the previous element in the array and compares it with the current element in loop. If `prev` is greater than the current element, the loop ends and `isMonotonic` returns false. Otherwise, `isMonotonic` returns true.

IV. Problems

No problems were encountered implementing any of the three functions. However, occasionally Keil µVision would crash upon exiting the debug session of this particular program, due to unknown causes.

V. Results

Screenshot of code:

Screenshot of `isMonotonic` and UART Monitor results:
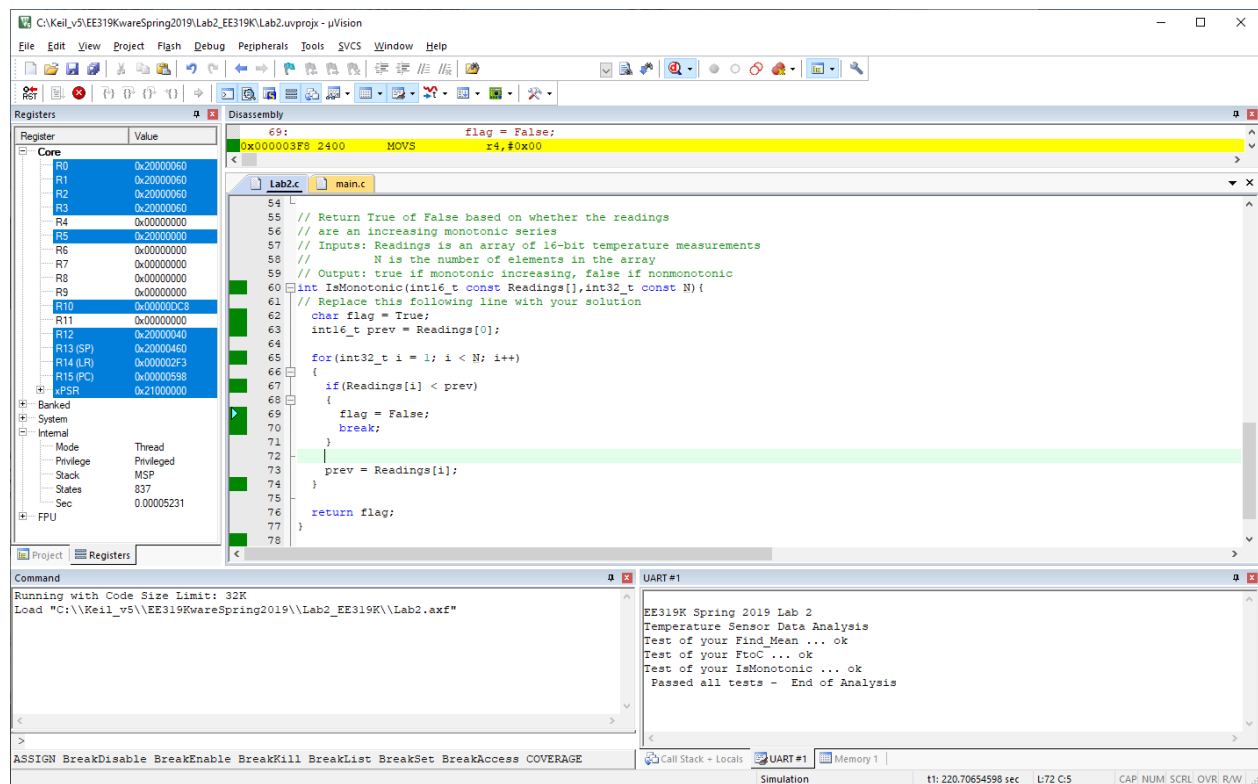
A copy of `Lab2.c` and the implementations of the three functions within is provided as a separate file alongside the lab report.