	ECGR 4101/5101 LAB 4 Report	09/28/2023
Nahush D. Tambe		**ntambe@charlotte.edu**
Bharat Kumar Duraisamy Krishnamoorthi		**bduraisa@charlotte.edu**

Lab Objective:

The objective of this lab was to build on top of the previous lab's complexity by using analog to digital conversion (ADC) values in addition to interrupts and timers to first determine what the digital values are based on the analog inputs and then display those digital values. The goal was to create a buffer to store the 3 digits and display the scaled digital values as well as the negative sign at the appropriate place on the Quad-Digit Display. This lab specifically tested the ability to code as efficiently as possible, while keeping the number of executable lines of code within the limit of 50, thus making the program optimized.

Lab Figures/Tables:

Table 1: PORT 1 PIN Mappings to Potentiometer and Quad-Digit Display

PIN # on PORT 1	Purpose
PIN 3	Analog Input
PIN 4	Digit 1
PIN 5	Digit 2
PIN 6	Digit 3
PIN 7	Digit 4

To power ON the specific digits on the Quad Display, PINS 4, 5, 6, and 7 on PORT 1 were connected to the Quad-Digit as shown in Table 1. Ina addition, the Potentiometer was also connected to the MSP430G2553 with VCC, GND, and the analog input to PORT 1 PIN 3.

Table 2: PORT 2 PIN Mappings to Segments on the 7-Segment LED

PIN # on PORT 2	Specific Segment
PIN 0	a
PIN 1	b
PIN 2	c
PIN 3	d
PIN 4	e
PIN 5	f
PIN 6	g
PIN 7	dp

To connect the seven Segments of each of the individual digits on the Quad-Display with MSP430g2553, pins 0-7 on port 2 were used and the connections as shown in Table 2 were made. Pin 0 was connected to segment a, and so on until the last segment, the point (dp) was connected to Pin 7.

Table 3: Hex Digit's Binary and Hexadecimal Values Used

Hex Digit	Binary Combination	Hexadecimal Combination
0	1100 0000	0xC0
1	1111 1001	0xF9
2	1010 0100	0xA4
3	1011 0000	0xB0
4	1001 1001	0x99
5	1001 0010	0x92
6	1000 1001	0x82
7	1111 1000	0xF8
8	1000 0000	0x80
9	1001 0000	0x90
-	1011 1111	0xBF

To make the code efficient and smaller in length, an array with 11 elements was declared that included decimal digits 0 to 9, and the eleventh element was the negative sign. The hexadecimal combinations of the 11 unique elements were stored.
The connections were made for the Anode. i.e., 0 represented OFF state and 1 represented ON state. To display different digits, a predefined combination in terms of Hexadecimal values was used and every time, a different element was turned ON. Table 3 shows a list of the hex digit and its binary as well as hexadecimal representations.

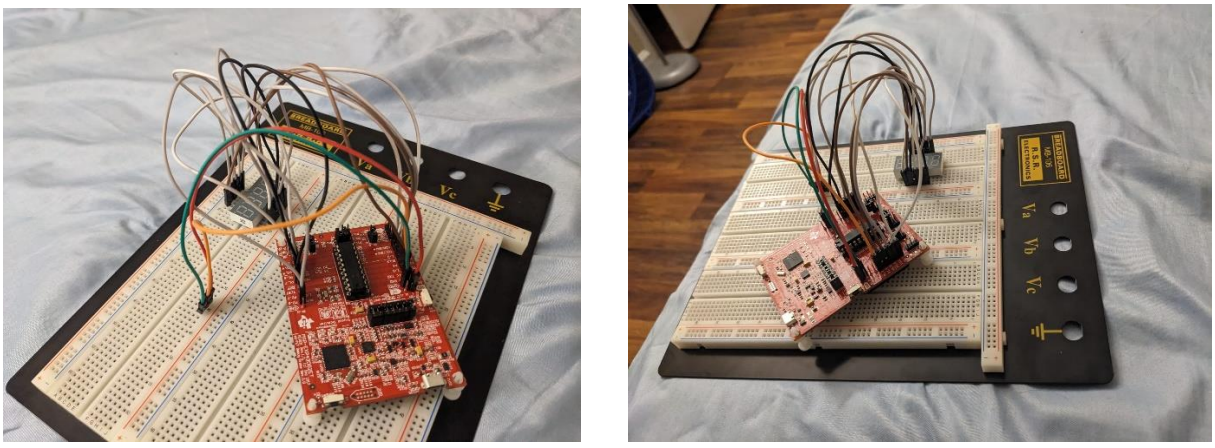

Figure 1: Connections made between the Breadboard and MSP430g2553 for Quad-Digit LED Display to display Digital Values

Commentary and Conclusion:
Lab 3 was easy to follow and code because the data sheet of the Quad Display provided every detail that was needed to connect the Quad-Display with the MSP430g2553. Like the 7-Segment LED, the connections for 1-g segment were made to PORT 2, and 4 power connections for each of the 4 digits were made on PORT 1. Next, some code was written to test the display, and gradually the code was built for the lab. Apart from the Quad Display, rest everything was

kept the same in terms of physical connections, including the potentiometer connection to the board by making connections to Vcc, Gnd and to an Input PIN on PORT 1.

With all the connections made, the potentiometer was tested for values -512 and +511. The digital values ranging from 0-1023 were scaled down to get it the range -512-511. As this lab had to be built up on the previous lab, the code for timers and interrupts was also added to the working code. A few problems that were faced along the way included flickering of digital values and slightly incorrect range that did not go past 508. The flickering was partly solved by taking the average sum of 7 values to make the display more stable. In conclusion, the lab went as expected and the Quad-display properly showed the digital values, but skipped some values in between, and this was since the delay was too long.