	ECGR 4101/5101 LAB 5 Report	10/05/2023
Nahush D. Tambe		**ntambe@charlotte.edu**
Bharat Kumar Duraisamy Krishnamoorthi		**bduraisa@charlotte.edu**

Lab Objective:

The objective of this lab was to enhance the complexity of the previous lab by incorporating UART communication between two MSP430g2553 boards. Board 1 was programmed to transmit a digital value between 0 to 13, obtained from a simple Button Press. Subsequently, Board 2 was expected to perform mathematical conversion to convert the received value to a number (2 ^ value received) before displaying the number on the quad-digit display. The goal was to achieve an efficient and concise code, minimizing the number of the number of executable lines, thus ensuring optimized application easier to read and follow through.

Lab Figures/Tables:

Table 1: Board 1 PIN Mappings

PIN # on PORT 1	Purpose
PIN 2	UART Connection with Board 2
PIN 4	Analog Input

PIN 2 was connected to Board 2 for UART Communication between the two boards. Additionally, PIN 4 was connected to the button switch to collect the user input. Table 1 provides an overview of the same.

Table 2: Board 2, PORT 1 PIN Mappings

PIN # on PORT 1	Purpose
PIN 1	UART Connection with Board 1
PIN 4	Power to Digit 1
PIN 5	Power to Digit 2
PIN 6	Power to Digit 3
PIN 7	Power to Digit 4

On the 2nd Board, PINS 4 through 7 were connected to the Quad-Digit display powering the 4 digits of the display. PIN 1 of this board was connected to the 1st board for UART communication. This board was the Receiver in this communication setup. Table 2 provides an overview of the same.

Table 3: Board 2, PORT 2 PIN Mappings to Segments on the Quad 7-Segment LED Digits

PIN # on PORT 2	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

PINS 0 through 7 on PORT 2 of Board two were connected to each of the 7 segments on the digits of the Quad-Display and the 7th pin was connected to the decimal point. Table 3 provides an overview of the same.

Table 4: 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

By using an array of 10 elements that stored the hexadecimal combinations of the 10 unique digits (0-9), the code was efficiently optimized and reduced. The connections were made assuming an Anode. i.e., 0 represented OFF and 1 represented ON. To display the digits, a predefined combination from one of the 10 values was used and turned ON. Table 4 shows a list of the hex digits, its binary, and hexadecimal representation.

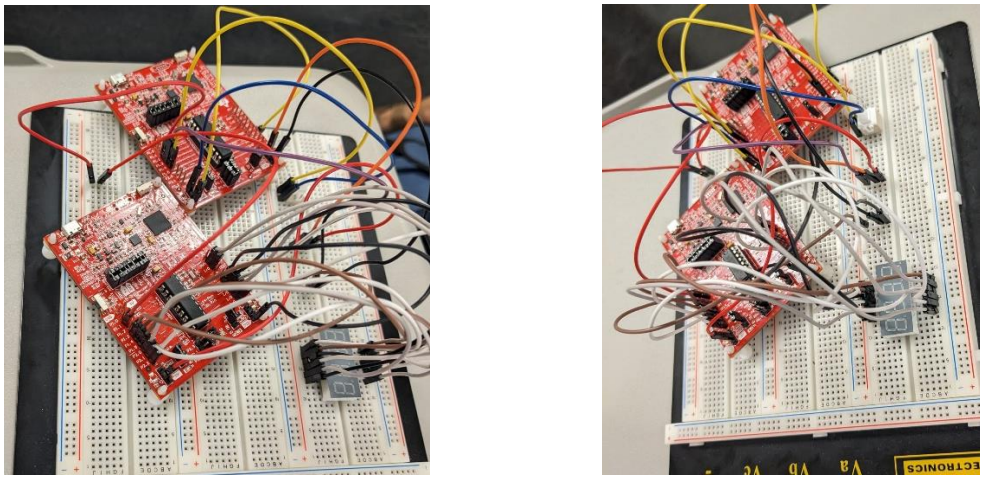

Figure 1: Connections made between the two MSP430 Boards and the Breadboard

Commentary and Conclusion:

In lab 5, two MSP430g2553 boards were interconnected and UART was used for communication. The lab requirements were addressed incrementally. Initially, the code from the previous lab was used as a starting point, which was then edited, and refined as necessary. Next step began with an initial test of the UART communication through transmission and display of a digital value. Next, the button functionality was integrated and the second board with the quad-digit display connections already made was added. Finally, everything was connected together and then tested for any issues with the hardware as well as the software.

During the lab, a few challenges were encountered, yet the overall progress was notably successful. One of the problems was the flickering of a couple of digits on the quad display. Digit 2, 3, and 4 kept flickering, while digit 1 stayed bright and comparatively constant. To fix the issue, multiple methods were tested like changing the delay of the timer, or making some structural changes, but the issue was not solved. In the end, it was figured that the pow() method was causing a considerable amount of delay, and hence was taken out and the code was implement without the method.

One of the problems that was left unresolved was the button Press values were being tracked correctly, but with the help of a resistor. Due to time constraints and use of code and setup from lab 1, the resistor proved to be more of a necessity. When it was taken, the code ran in a loop displaying various numbers or single digits. Another difficulty was the use of multiple different physical connections between the two boards and the breadboard. Some unnecessary connections were made, which made the circuit debugging more difficult. Figure 1 above provides an overview of the physical connections that were made between the various components.