Joseph Fiume

6133053

EE 215

Lab 7

10/30/2020

**Objectives**

Write a C-code program to use the LCD screen on the MSP430FR6989 board.

**Explanation of Approach**

To make use of the built in functions for the LCD screen, we have to import the “driverlib” folder. The files in driverlib must be included in the project. The LCD screen 1st displays my first name: JOSEPH for 2 seconds using the myLCD\_showChar() function. Then it displays the numbers 1 through 6 in sequential order using the myLCD\_displayNumber() function. The numbers are displayed in the rightmost location of the screen for one second each. Finally, the LCD displays the battery charging symbols using the myLCD\_showSymbol() function. It cycles the battery charging symbols from the rightmost to the leftmost. After the battery is “full” (all the battery symbols are on), it blanks the screen for 1 second. This cycles infinitely.

**Code**

**#include** <msp430.h>

**#include** <driverlib.h> // Required for the LCD

**#include** "myGpio.h" // Required for the LCD

**#include** "myClocks.h" // Required for the LCD

**#include** "myLcd.h" // Required for the LCD

**const** **unsigned** **int** BATTERY[] = //list of battery symbols

{

LCD\_B6,

LCD\_B5,

LCD\_B4,

LCD\_B3,

LCD\_B2,

LCD\_B1

};

**void** **main**(**void**)

{

WDTCTL = WDTPW | WDTHOLD; // Watchdog timer stop

// initialize LCD

**initGPIO**(); // Initializes inputs and outputs for LCD

**initClocks**(); // Initialize clocks for LCD

**myLCD\_init**(); // Prepares LCD to receive commands

TA0CTL |= TASSEL\_1|MC\_1|ID\_0; // ACLK, count up to CCR, divide clock by 1

TA0CCR0 = 0x8000; // counter limit 0x8000 = 32768 = 1 second

**unsigned** **int** j = 1; //create a flag for displaying the name

**unsigned** **int** i = 1; //create an index for the numbers counting

**int** bat\_index = 0; //create an index for the battery symbols array

**while**(1) //loop forever

{

**if** (TA0CTL & TAIFG) //If flag is set from counting up

{

**if** (j == 1)

{

**myLCD\_showChar**( 'J', 1 ); // Display "J" in space 1

**myLCD\_showChar**( 'O', 2 ); // Display "O" in space 2

**myLCD\_showChar**( 'S', 3 ); // Display "S" in space 3

**myLCD\_showChar**( 'E', 4 ); // Display "E" in space 4

**myLCD\_showChar**( 'P', 5 ); // Display "P" in space 5

**myLCD\_showChar**( 'H', 6 ); // Display "H" in space 6

j = 0; //turn the name display flag off

}

**else** **if** (i <= 6) //count from 1 to 6

{

**myLCD\_displayNumber**(i); //display the number on the LCD screen

i++; //increment the number

}

**else**

{

**myLCD\_showChar**( ' ', 6 ); //blank the LCD screen

**if** (bat\_index <= 5) //allow battery indexes in range of battery symbol list

{

**myLCD\_showSymbol**(LCD\_UPDATE, BATTERY[bat\_index], 0); //display the battery symbol at given index

bat\_index++; //increment the battery index

}

**else**

{

**for** (;bat\_index >= 0; bat\_index--) //count down from the highest battery index 5-0

{

**myLCD\_showSymbol**(LCD\_CLEAR, BATTERY[bat\_index], 0); //blank the battery symbol at the given index

}

i = 1; //set the numbers index back to its staring position

j = 1; //set the name display flag

}

}

TA0CTL &= ~BIT0; //Reset Flag

}

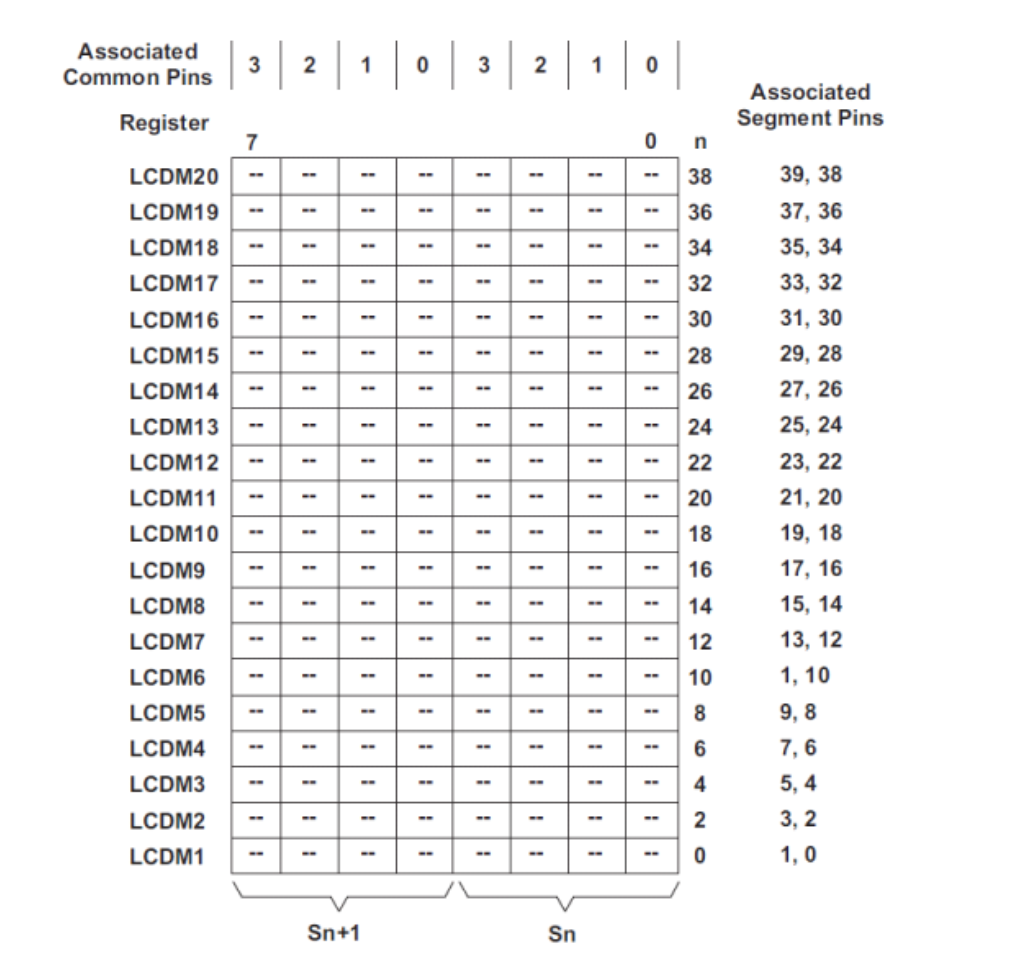
}

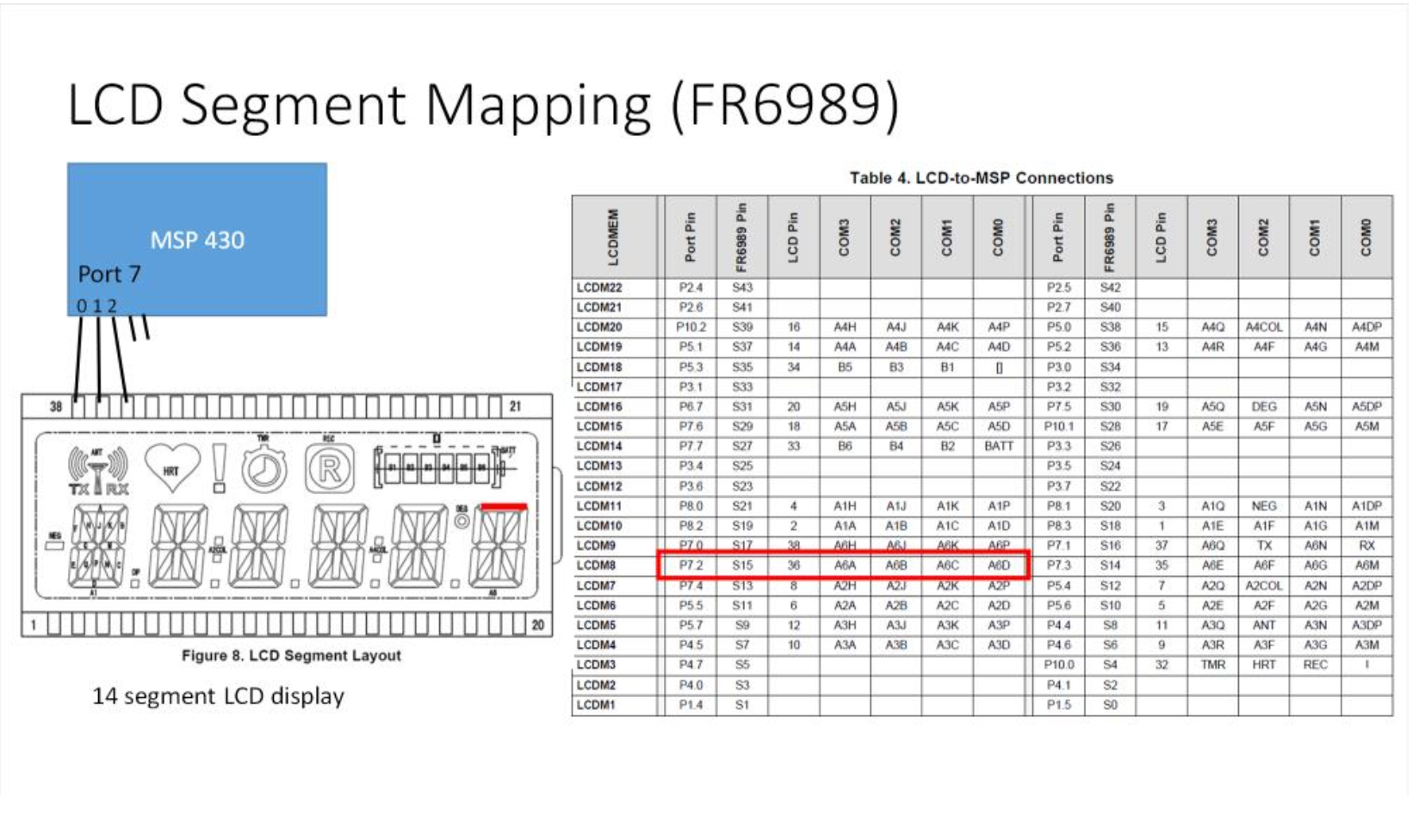
}

**Results**

The LCD screen displayed my name for 2 seconds. The LCD then displayed numbers from 1 to 6. The LCD screen then displayed the battery symbols cycling forever. The LCD loops forever.

**Explanation of Results**

The bits set to turn on LCD\_B1 are Bit 5 in LCDM18. The memory register to turn on LCD\_B1 is Pin 5.3 and COM1.



**Conclusion**

I learned how to use the built in functions to display letters, numbers and symbols on the LCD for the MSP430. I also learned how to include additional folders for a project in CCS.