School of Engineering

Grand Valley State University

EGR 326 – Lab #6

**Interfacing a Graphic LCD to the MSP432 MCU**

**Objectives**

* To gain experience with interfacing a graphic display device to the MSP432
* To configure and use the Serial Peripheral Interface (SPI) hardware implementation on the MSP432
* To provide visual display feedback to embedded system users

**Introduction**

In this lab exercise, you run the test program that was studied in the prelab assignment as well as your own program that was written to display new information on the LCD. Also, you will format a bit mapped image, display it and will write a program to update information at specific locations on the display.

**Part I: Displaying Characters on the LCD from the MCU**

* Using your project started in prelab, extract portions of the code from the file ST7735TestMain.c into your main.c that perform some screen tests by calling on the graphics libraries in the ST7735.c file. First replace Valvano’s function call Clock\_Init48MHz() with your own version that enables the 48 MHz crystal oscillator as described in lecture. Confirm that the MCLK is set to 48000000 and then configure SMCLK for 12000000. You can also replace Valvano’s delay functions with functions using the SysTick timer, or use his subroutines that count cpu cycles tuned for a 48 MHz clock. **Make sure to include a scope plot of both the MCLK and SMCLK.**
* Add one more test routine that clears the screen, makes the background **BLUE,** and displays “**EMBEDDED SYSTEMS**” in white centered on the screen in a font size of your choosing (you can get creative with this part of the assignment and even add animation if you wish). Come to lab ready to demonstrate that you can run the suite of test routines for the LCD module.

**Part II: Displaying Images on the LCD from the MCU**

* Notice that in the ValvanoWare project ST7735\_MSP432, there are extra files that are not part of the build. CCS allows you to store related files in a project folder so that everything you need to support a project stays together. Read the **BmpConvert16Readme.txt** file for instructions on how to use the **BmpConvert16.exe** program to create an array of color pixel information for the ST7735 from a color bitmap file. Find (or create) a 24 bit color bitmap file and edit it using Microsoft Paint (or some other program). Remember that the maximum size is 160x128 pixels. Scale (or crop) the image in the editing program to fit the screen. You can use the BmpConvert16.exe program to create an image array to include in your program for display on your screen.
* Write a program to display this image at program startup (this is often called a “splash screen”, see how this was done in the example code). Pause three seconds and then clear the screen (make it all white or all black). Follow this by writing 3 lines of text that contain a menu of choices properly aligned in the middle of the screen with characters scaled by a factor of two. The menu can contain any 3 items that fit on the screen, think about what you may use in your design project. Demonstrate this for your instructor.

**Part III: Updating characters at different places on the LCD**

* Now that you have successfully controlled pixels on your LCD to display text and produce a “splash screen”, write one more program that uses a size 4 font, displays a count from 0 to 9 (delay 1 s between counts) in all 4 corners of the screen. Next count from 0 to 9 (delay 1/2 second between counts) in the middle of the screen. Have it pause for 3 seconds and then repeat the sequence. Demonstrate this for your instructor.

**Restate and answer the following questions in your lab notebook:**

* Study the ST7735 graphics library routines in the file ST7735.c and understand how all of the elements of it work.

1. Describe the MSP432 SPI interface with the graphics LCD controller. Provide a brief description of how it is used to display image data from an array in your program to the LCD screen.
2. You used a program to convert from a bit mapped image in a file to an array that was included in your program source code. What conversions are necessary to make the data in the image file into a usable array to be displayed by your program?
3. How does the graphics library display ASCII characters? How does it scale the characters to be of different sizes? How does it place them in a particular spot on the screen?

**At The End Of The Laboratory**

* ***Clean up your workstation.***
* ***Include your .c files containing code you’ve written for the exercise as an appendix to the lab report. Only include code that you’ve written, not the library code that you imported. Then save the .doc file as .pdf for submission.***
* ***Make sure you clearly understand the laboratory deliverables and due date as posted on Blackboard.***