To: Dr. Snider

From: Nash Reilly

Regarding: Lab #4—Adding an LCD to your SoPC

Date: September 27, 2011

**Introduction**

The purpose of this laboratory session was to use Qsys to add LCD control lines to the Nios II virtual processor, instantiate it within the top-level VHDL code, and then write a program in Eclipse that would use the LCD to tell the user some simple information about switch activity on the board.’

**Lab Summary**

After resolving a small issue with the Qsys file’s default directory, it was a simple matter to route the new connections between the Nios II core and the LCD control lines on the DE2 board. The task of writing to the LCD using the Nios II interface, however, was slightly more involved. How were we supposed to treat the LCD from a software perspective? The answer lay in the board\_diag.h library included with the Nios II development kit. The board\_diag library contains a number of operations designed to test the functionality of a DE2 board environment. Among these is a test LCD function. This contains the following code segment that we modeled our WriteLCD function on. The WriteLCD() function written for lab 4 is included below.

**static** **void** WriteLCD(**char**\* msg)

{

FILE \*lcd;

**static** **char** ch;

**static** **char** entry[4];

lcd = fopen("/dev/lcd\_display", "w");

/\* Write some simple text to the LCD. \*/

**if** (lcd != NULL )

{

fprintf(lcd, "\n%s", msg);

}

printf("The LCD should be displaying \"%s\".\n", msg);

fclose( lcd );

**return**;

}

**Figure 1: WriteLCD() function**

The figure above shows the method used to write to the LCD display on the DE2 board. This treats the LCD screen as a file pointer; the file in question is the device file located in the “/dev” directory of the Nios II firmware. Using the fprintf function, we can write to this file the same way we would write to a text file. Using this, we can write text to the LCD by passing WriteLCD() the pointer to the first letter in a string, and then check to see that the intended text appears on the LCD.

**Conclusion**

By treating devices in the Nios II System as file pointers, we can use C to automate what would be tedious tasks at the hardware level. In summary—a file pointer and a function that takes a string pointer as an argument were the solutions to our problem.