**Laboratory 2**

**Keyboard Interfacing**

**Post-Lab**

**Team Number 5**

**Authors**

**Christopher Wilcox**

**James Nystrom**

**Christopher Wilcox**

**James Nystrom**

**LAB 2 - Keyboard Interfacing**

In Laboratory 2, we are asked to assemble a program to read a matrix-connected keypad. We are introduced to the operation of such a keypad, and our abilities to properly program the device are put to the test. After reading in the input, we are to display the last four digits pressed to the seven segment array.

We utilized previously written code from Laboratory 1 to drive the display, and constructed new code using Timer 0 and the FIQ interrupt to handle the keypad presses. The FIQ interrupt allows a user to press a key, and have it displayed as the right most digit of the seven segment array. We also added the additional functionality of a repeat key. If a user holds a key for a second, it will repeat on the display. This will continue until the user chooses to release the key.

From this lab, we learned many things. Like laboratory 1, preparation for the lab was paramount. Even with advanced preparation, debugging and other various troubleshooting quickly filled the three-hour lab timeslot. By spending time ahead of time, and generating code, we were able to leave the lab in a timely manner. We also learned from laboratory 1, that register management can be a challenge to debug. We documented register usage heavily in hopes of not spending time tracing hard to find errors. Taking the time to document register usage did save us time, and allowed us time to think of ways to increase the efficiency of our code, as opposed to debugging it. One of the small tweaks we performed was to store the value 0xF in a register for the FIQ mode, as it is used four times in different places, and each time is moved into a temporary register. This improvement, along with others is a few ways in which we utilized our time effectively.

**Post-Lab Deliverable**

The lab manual requests that the following question be answered:

**Q:** Using the trace and what you know about the circuit, estimate the total capacitance on the line. (Using the scope cursors to find key time/voltage points is much better than trying to measure things on the trace print-out!)

Explain how you calculated the capacitance value.

**A:** Below is the calculation of the capacitance value. The trace is attached as Item 1.

Vc = V(1-1/e)

2.1234 = 3.359(1-1/et/RC)

2.1234 = 3.359 ( 1 - e(-760 x 10-9/10 X 103C))

.63212 = 1 - e(-760 x 10-9/10 X 103C)

ln(.36788) = ln( e(-760 x 10-9/10 X 103C))

-1 = -760 x 10-9/10 X 103C

-10 X 103C = -760 x 10-9

C = 7.6 \* 10-11 F

According to the Altera Cyclone manual, the typical values of a user I/O pin is around 4.0 pF. This is close to the number we found because our calculated value includes the capacitance of the oscilloscope wires and the trace in the board. This is the difference from the calculated value as provided by the Altera Cyclone manual.

**We have attached all requested schematics, waveforms, and code. Below you will find a list of contents.**

1. Oscilloscope trace showing the RC charge curve of a keypad column drive line.
2. Quartus Pin-Out Diagram. Note the Open Drain Buffers on the Column drive lines.
3. main.s
4. exceptions.s
5. IRQ\_Handler.s
6. FIQ\_Handler.s