CST 345 lab 04

Keypad Controller

By: Cal Barkman

**High level overview:**

This lab will take a keypad’s input and use it to display hex characters on a segmented display. The keypad is a 4x4 scanning keypad that drives an active low line on the 4 bit column output and receives an active low signal on the 4 bit row input. The keypad is driven with a 1 KHz clock signal to help with debounce. The 4 part 7 segmented display has a muxed input which allows for displaying to all 4 parts of the display, one at a time on a rotating basis using the 4 anodes to select which segment in the display and the 8 cathodes to select which of the 8 leds for the active anode will turn on. The 7 segment display runs on a 1 KHz clock to make the display appear to always have each segment lit up to a human eye.

When a key is pressed, its value will be displayed on the first position of the 7 seg display and the rest of the displayed values will be shifted down from right to left like a calculator display. When a push button on the board is pressed, reset the display to all 0’s. The display will initially show all 0’s.

**High level hardware block diagram:**



**Design of hardware/software interfaces:**

7 seg display interface diagram



Keypad controller interface diagram



**Software documentation:**

Flow chart:

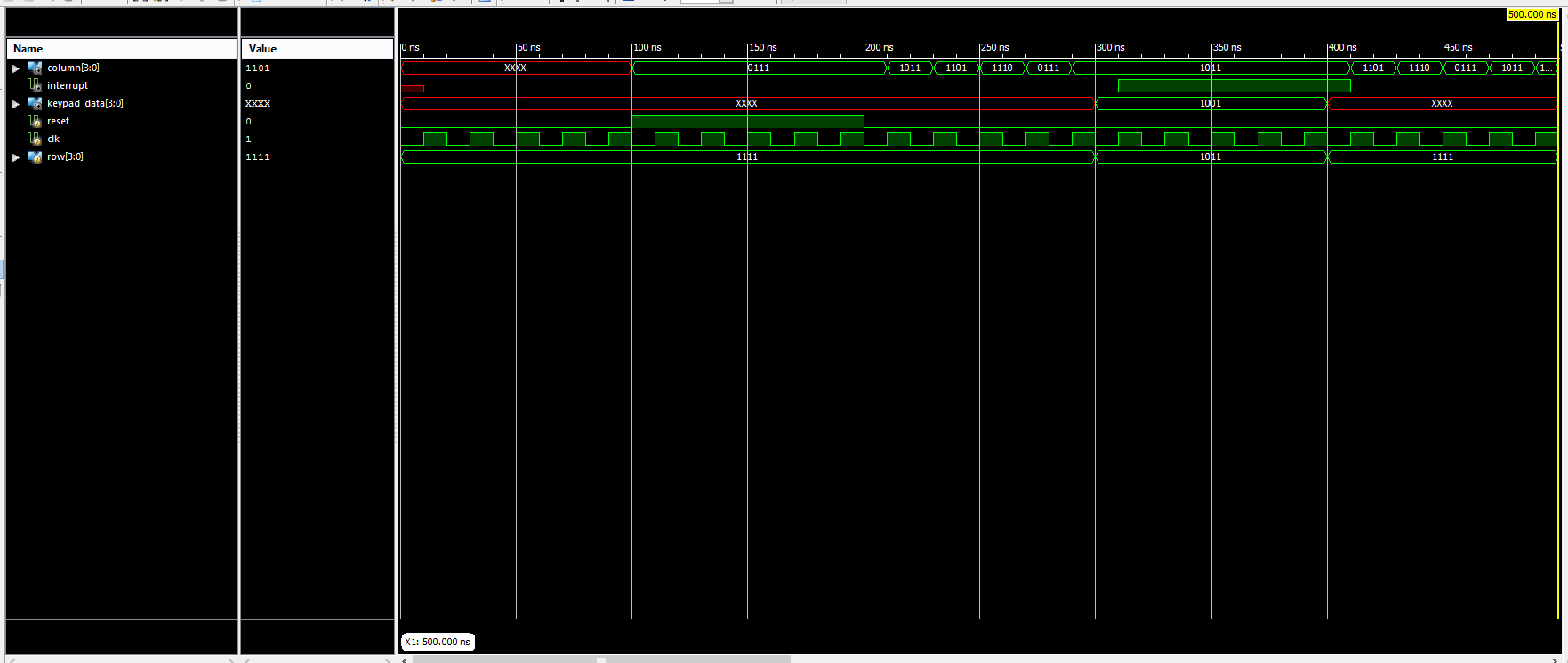


The software for this lab was relatively simple. The only initialization is to enable interrupts. In the main loop, the software reads the button, then checks to see if it is pressed. If it is, the registers for display are cleared. Then the value outputted to the muxed display, and returns to input the button again.

When an interrupt is triggered, the Interrupt Service Routine (ISR) will input the keypad value, load the registers for muxed display to the adjoining display registers making the values appear to shift left across the display, and load in the keypad value into the rightmost display slot.

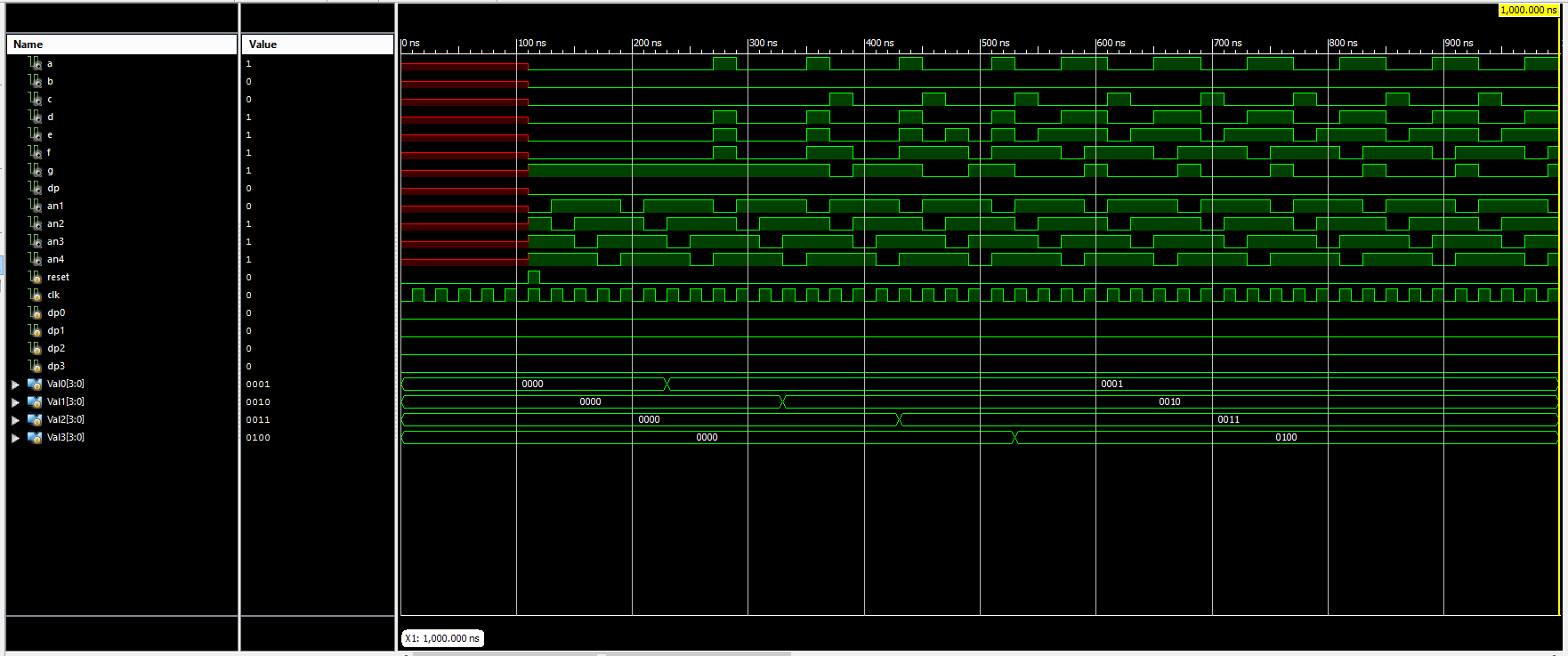
**Results of hardware and software simulation:**

**Hardware simulations:**

Keypad simulation:

Higher res version included in zip as KeypadSimulation.PNG

Muxed 7 seg display simulation:



Higher res version included in zip as KeypadSimulation.PNG

Software test process:

1. Press key 1
2. Confirm that the 1 symbol is at the right most positon on the 7 segmented display
3. Press key 0
4. Confirm that the 0 symbol is at the right most position on the 7 segmented display and the 1 is in the 2nd right most positon
5. Repeat steps 3 and 4 with symbols 2-F in hex.

**Conclusions:**

The only real engineering decision I had to make was how to shift data on the muxed display. At first I considered using 2 Picoblaze registers and using shift commands, but this would have taken several instructions. Instead, I opted to use 4 registers and simply load them in series, taking only 4 operations to “shift” the display.

Using the keypad with the interrupt structure makes sense and works well, but would require an interrupt controller to use any other interrupt devices. Having a single interrupt and the difference in operation speed of the Picoblaze in comparison to the keypad and muxed display controllers ensures that there is no problems with timing. If the picoblaze was to operate at a similar rate as the keypad and muxed display controller there could be potential timing issues.