

Keyboard to 7-Segment Display.

List of Authors (Ryan Christenson, Andrew Goins)

Electrical and Computer Engineering Department

School of Engineering and Computer Science

Oakland University, Rochester, MI

e-mails: rmchristenson@oakland.edu, asgoins@oakland.edu,

Abstract— The purpose of this project was to better understand the use of the VHDL coding language and use this language to allow the interaction between a basic usb keyboard and a seven segment display used on our nexys 4 board.

I. INTRODUCTION

Our project involved the understanding of VHDL language that would allow us to use a keyboard to successfully type characters on our seven segment display. We decided this would be an interesting and useful project that we may be able to draw experience from for the future of our careers.

Several topics in class were implemented in this project including registers, multiplexors, and decoders. We had to learn how to combine all the components together to make a successful design. Our project could be used to write to any 7-segment display using a keyboard. This could be used to display signs that could be easily changed.

II. METHODOLOGY

A. Initial Approach

When we decided to take on this project the first idea that came to mind was attempting to emulate the design of Lab 4. In this lab we used registers, a mux, and switches for the input of a 7-seg. We decided that if we used a keyboard as the input using code given by Prof. Llamocca, we could reach the desired effect. The top file of the design can be seen in **Figure 1**, and the finite state machines used within the top file are further explained in **Figure 2** and **Figure 3**.

B. Understanding The 7-Segment Display

We reached the point where we could display one character on only one of our eight 7-segment displays at a time. Obviously this wasn't good enough, so we studied the 7-segment display and learned that in order to display all 8 segments at once we needed to alter the multiplexer being used, in other words, display one at a time so quickly that it seems as if all are being displayed at once [1]. The type of multiplexer used for this can be seen in **Figure 4**.

We set the output of the keyboard to go directly to the registers to be stored, the multiplexer then cycles through each register at a rate fast enough that it seems like all 7-segment displays are shown at once. After the data goes

through the multiplexer it is converted to data through a decoder that is readable on the 7-segment display.

III. EXPERIMENTAL SETUP

For this project we used the Nexys 4 board, a keyboard as our input, and the 7-segment display to show our output. We designed our code using ISE Design Suite 14.7. By the end, we expected our project to be able to write exactly what we wanted on the 7-segment display using the keyboard, and allow every character to be shown at once.

IV. RESULTS

After implementing our final code, we found that we could reach our desired effect. We used the switches to set the address of the desired 7-segment display destination and used the keyboard to type whatever we desired. We created a test bench to simulate our results. The simulated timing diagram, as seen in **Figure 5**, shows that any input data will be sent to the multiplexer, decoded onto the specified 7-segment display, and cycled through on a continuous loop.

Unfortunately our project wasn't perfect. One unwanted result is that if we input data to one register and then change registers without turning the enable switch to "read," the 2nd register will be written to as well. This is hard to notice sometimes because that 2nd register will be rewritten anyways as we continued typing.

CONCLUSIONS

From creating this project we found that there can be many uses for the Nexys 4 board and the VHDL coding language. It is a very useful language that can create many things and depending on the amount of work that you put into it, projects can work very well. We wish we could have improved our project so that it worked without glitches.

REFERENCES

- [1] R. E. Haskell, D. M. Hanna, "Digital Design Using Diligent FPGA Boards VHDL/ Active-Hdl Edition LBE Books, 2nd Edition, pp. 93-98, 2012.

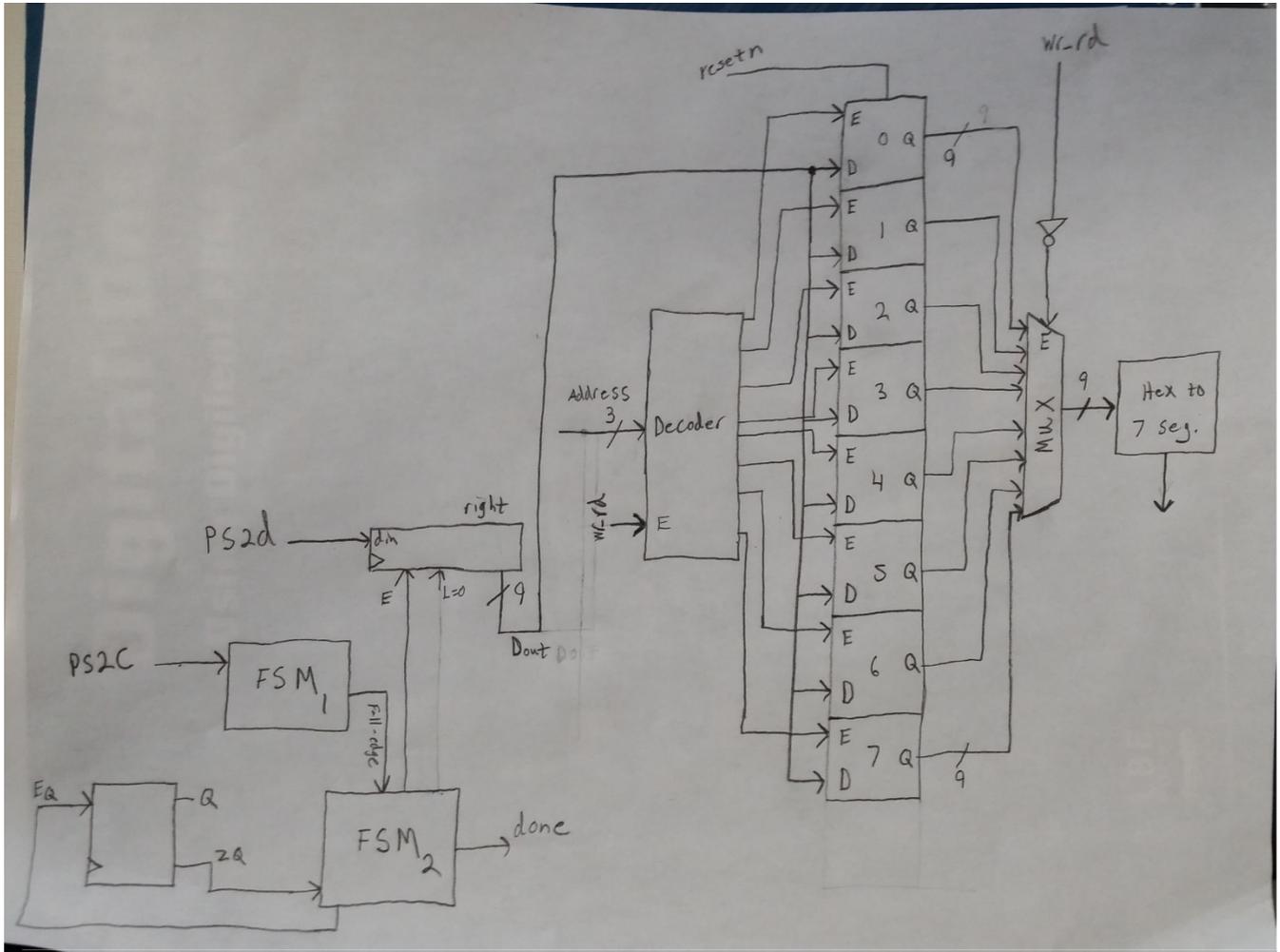


Figure 1: Top-file design of the keyboard to 7-segment display.

FSM₁

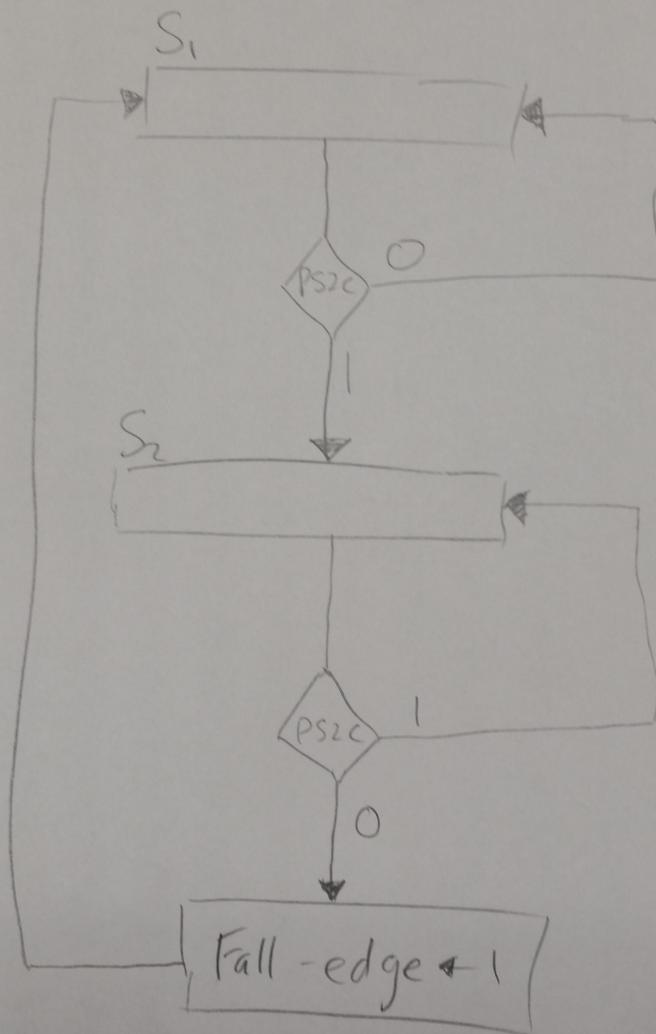


Figure 2: Finite state machine 1 from the top file module.

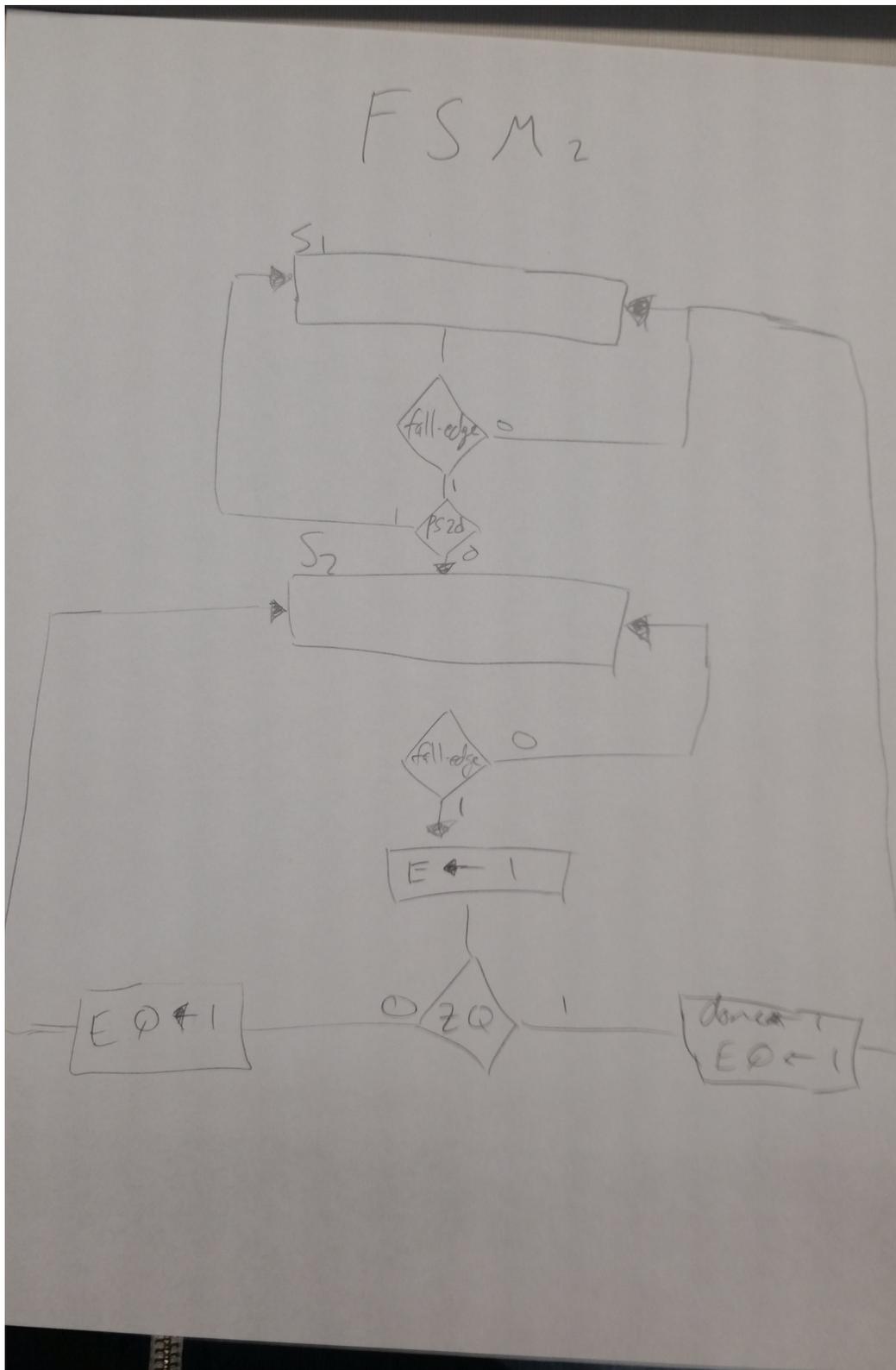


Figure 3: Finite state machine 2 from the top file module.

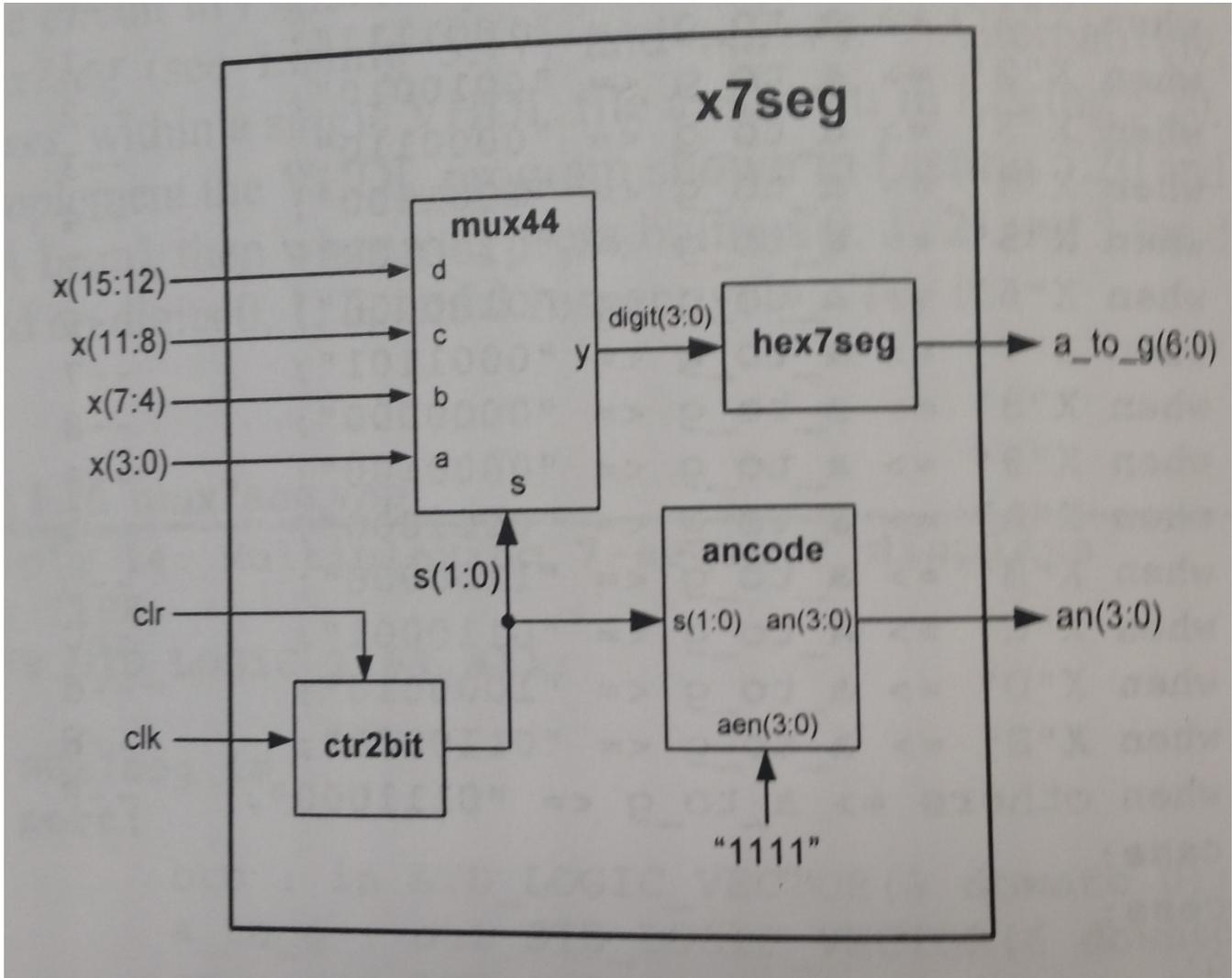


Figure 4: Type of multiplexer used to continuously display all 8 segments.

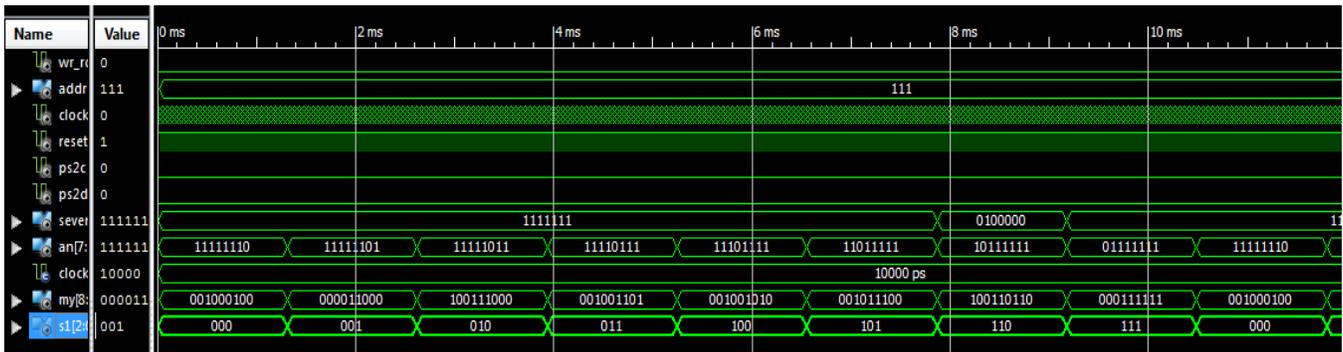


Figure 5: Simulated timing diagram from the test bench module.