Multilingual Banner on 7 Segment Display

Digital Logic - ECE 2700

List of Authors (Khaled Baayoun, Clayton Brockway, Mackenzie Bussell, Huey Le)
Electrical and Computer Engineering Department
School of Engineering and Computer Science
Oakland University, Rochester, MI
e-mails: kbaayoun@oakland.edu, cbrockway@oakland.edu, mackenziebussel@oakland.edu, hueyle@oakland.edu

***Abstract*— The purpose of this project is to construct and implement a fixed message onto an Artix A7 50t FPGA board. The message will be displayed on the 7 Segment Display and be controlled by switches. Each switch will display the message in a different languages and there will also be the option of displaying the message backwards. This will be achieved through the programming and circuit architecture knowledge obtained from previous laboratories in ECE 2700 and class lectures.**

I. Introduction

The goal of this project is to use Vivado to code and create a multilingual greeting on a 7 Segment Display. The banner will have multiple languages that will be controlled by switches and will be displayed on the 7 segment display on the NEXYS A7 board. For this to work, we need to use Decoders, MUXs, Finite State Machines, NOT & AND gates, clock dividers, and a serializer. To control the different languages being displayed, the circuit will have different switches, controlling which language is being displayed. A multilingual greeting can be used to communicate with multiple different languages and welcome them.

This can be useful in airports or at borders where there can be many people who speak different languages. This could also be used to communicate important messages with multiple languages.

II. Methodology

Our project will consist of a finite state machine (FSM), a clock divider, hex-to-7 segment decoder, a 1 ms counter, 3-to-8 mux, 4-to-4 mux, 2-to-32 decoder, and a 3-to-8 decoder. Our 3-to-8 decoder will select the bits for AN to determine which 7-segment display is being used. The hex-to-7 segment decoder will be used to determine which segment of the display is being lit up in order to display the string. The finite state machine will be used to determine which of the seven segments are being used to display the string. The 1 ms counter will turn on and off the eight 7-segment displays every 1 ms to create the illusion that all of the displays are on at once.

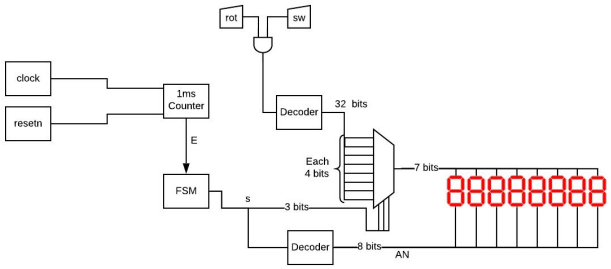

Figure 1: Multilingual Greeter

A. Counter

We ended up only using one counter in our project and that was a 1ms counter. The inputs to the counter were a 100MHz clock and resetn. the clock is generated from the NEXYS board itself and the resetn was controlled by the CPU_RESET button. The resetn is always set to '1'. The output goes in the finite state machine.

B. Finite State Machine

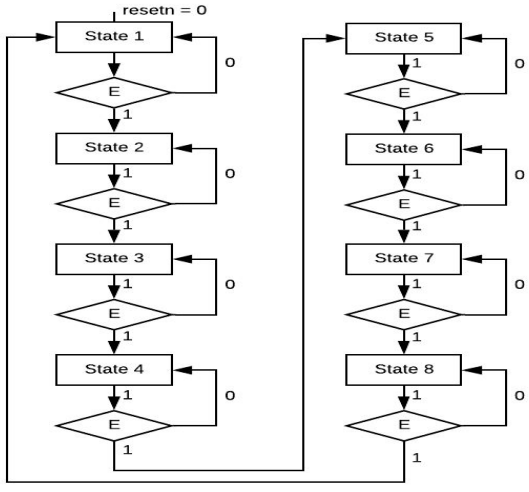

Figure 2: Finite State Machine

The inputs to the finite state machine are the E, 100 MHz Clock, and resetn. The enable for the finite state machine comes from the output of the counter. The finite state machine has a 3 bit output that goes directly to the multiplexor and controls which message is displayed. The finite state machine also send 8 bits to the 7 segment display.

The finite state machine has 8 different states.

C. Decoders

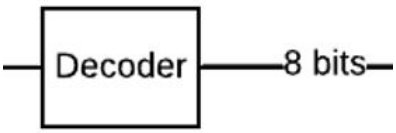

Figure 2: 3-to-8 Decoder

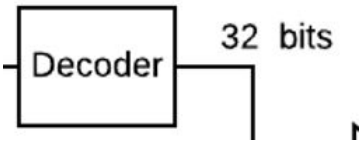

Figure 3: 2-to-32 Decoder

In this project, we have two decoders, one is a 3-8 decoder and a 3-to-32 decoder. The 3-to-8 decoder selects the bits for AN for our 7-segment display. The 2-to-32 decoder uses the switch inputs to decide which language will be displayed onto the 7-segment display.

D. Multiplexor

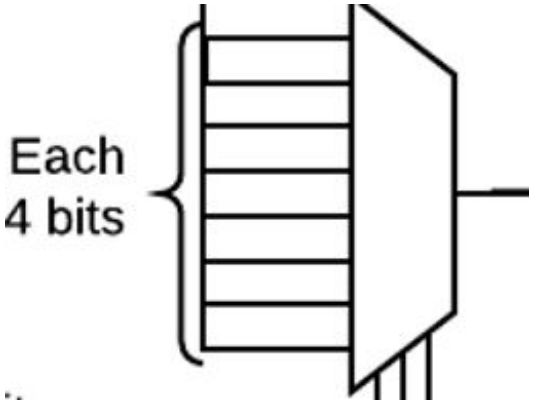

Figure 4: Multiplexor

There are 32 bits sent from the decoder that are the input to multiplexor. The output is what goes to the 7 segment display. The selector is controlled by switches that from the user. When “00” is selected, “HELLO” will be displayed. When “01” is selected, “HOLA” will be displayed. When “10” is selected, “NI HAO” will be displayed. When “11” is selected, “AHLAN” will be displayed. Another switch controls whether the message is displayed forward or backward.

III. Experimental Setup

We used the software Vivado 2018.3 to program our NEXYS A7 FPGA board. We programmed our board using the micro USB cable by plugging it into the port. We expected the board to display a message we wrote in the code and it did exactly that.

IV. Results

The following pictures show each of the four languages programmed into the NEXYS A7 Board. Using different switches result in the following messages.

English

Spanish

Chinese

Arabic

Conclusions

The important things learned during the project was just how powerful FPGA board and the abilities you can do with it. The finite state machine is a powerful that allows you to create anything you want. The main takeaway for us as a group is that it's better to start on paper with trying to map a circuit then just jumping right in to Vivado. Improvements that could be made to this project would be to program more languages and also have the ability to display custom messages. The code used in this project is an adaptation of Dr. Llamocca's "serializer" VHDL code. [1]

Our initial goal was to have the message scroll across the 7 segment display. Unfortunately we couldn't get that to happen as we could only get one letter to scroll. After thinking about it more, we realized that we were going to need registers. We then could have stored each letter in a register and this would have allowed us to shift the letters across the 7 segment display.

References

[1] Llamocca, Daniel. *VHDL Coding for FPGAs,* Unit 7 - Digital System Design. Retrieved from http://www.secs.oakland.edu/~llamocca/VHDLforFPGAs.html