One Player Pong

Using VHDL

List of Authors (Joshua Kulwicki, Devin Holz, Adriano Pjetrushi)
Electrical and Computer Engineering Department
School of Engineering and Computer Science
Oakland University, Rochester, MI
e-mails: JKulwicki@oakland.edu,DevinHolz@oakland.edu, apjetrushi@oakland.edu

***Abstract*— We wanted to create a single player pong game playable on the Nexys 100T board using VHDL. We will be using the buttons on the board to control the player's paddle. We will then output the game through the VGA Port onto a monitor, and display the score on the 7-segment display.**

I. Introduction

The goal of this project was to implement a single player game of pong. This game of pong would have a single paddle at the bottom of the screen and a ball that would start in the middle of the screen. Around the outside edges of the screen would be walls that the ball would bounce off of. The goal for the player would be to bounce the ball of their paddle to score points. They would try to get as many points they could before the ball passes their paddle. We thought this would be an exciting project while still testing the knowledge learned in class this past semester.

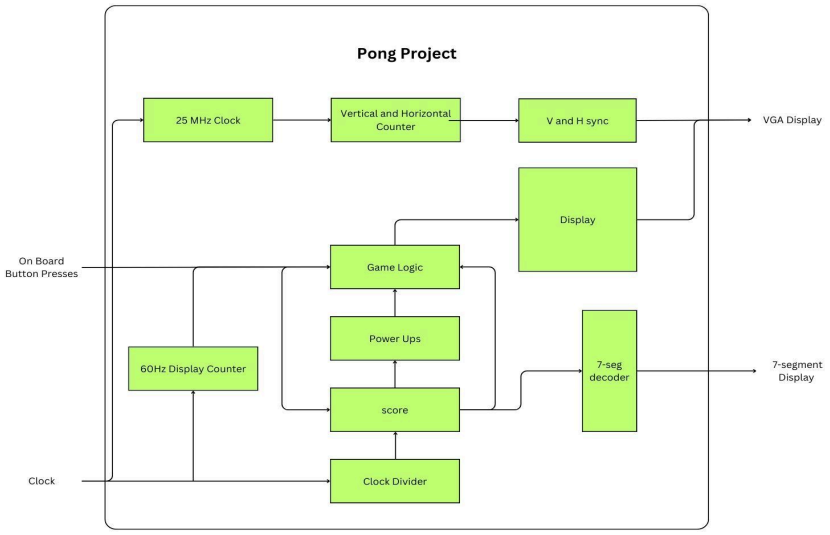
To complete this project we had to use the skills we learned in class about VHDL and learn new skills that were not covered in class. One of these new topics was the use of the VGA port to display the game. We did not have much experience with using the VGA in labs, so this required us to go out and learn more about this function to complete our project. Coming from a background of gaming it was fun making a classic game, though this does not leave much room for innovation. In the rest of this report we will go more in depth about the components that make this project and the improvements we can make in the future.

Methodology

We designed this project to be simple, and easy to use. We will be using nothing but the Nexys 100T board, and a VGA display monitor. The block diagram below shows components for the VGA, including the vertical and horizontal counter, the vertical and horizontal sync, the display, the counter for the score, and the game logic. The inputs from the buttons on the Nexys board, score, and powerups go to the game logic. This game logic component controls how the paddle and ball move, and how these two parts interact with each other. These outputs then go to the display component where the VGA Sync Porch will display the game. Also, a 7-segment decoder will be used to present the score [1].

A. Game Logic

Our first problem is to tackle the logic of the Pong game. There are many different ways to do this, and many different resources, but we chose to make a single player game. The score increases every time that the ball collides with the paddle. The data from the game logic is then sent to the Score component to keep track of the score. The Game Logic component also receives data from the Power Ups component, which controls the ball speeding up. Once the data has been processed, it is sent to the Display component to be displayed through the VGA Port.



B. Power Ups

The Power Ups component receives data from the Score component. As the score increases, the score count reaches a number of different thresholds. Once those thresholds are reached, the ball speeds up which increases the difficulty.

C. Score

The score component is just a simple counter that keeps track of how many times the ball collides with the paddle. The score data is then sent to the Power Ups component, and also a Seven Segment Decoder to display the player's current score on the screen.

D. Vertical and Horizontal Counter

The Vertical and Horizontal Counter keeps track of positional data on the screen. This data is used to draw the boundary of the game, as well as the ball and paddle. This also helps keep track of the on screen elements so the programmer can fairly easily make adjustments to the elements on screen [2].

E. Vsync and Hsync

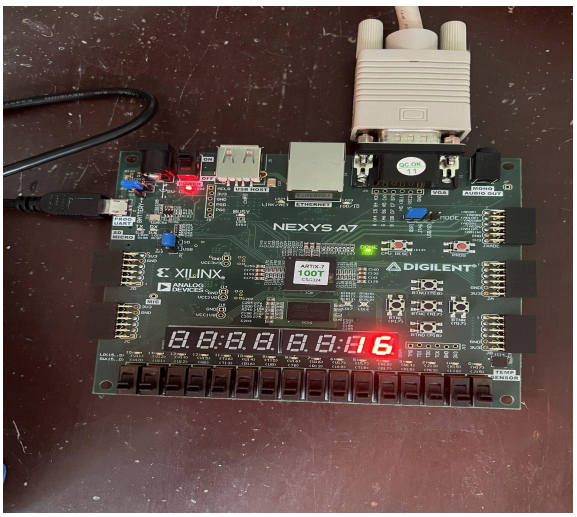
Vsync and Hsync are necessary signals for driving the VGA display. They ensure that the display is drawing at the proper location at the correct time. Unlike the Vertical and Horizontal Counter, Hsync and Vsync is just for driving the VGA port itself, and is not used in the game logic. Vsync and Hsync, however, does use data from the Vertical and Horizontal Counter to generate the Vsync and Hsync signals for the VGA port [3].

E. Display

The Display component handles what is actually being drawn onto the screen. It starts off with writing zeros to the RGB values of the VGA port. This creates a black background. On top of the black background, the Display component draws a white box around the edge of the display. This acts as a border for the ball to bounce off of. As the score increases and the ball gets faster, the white box, ball and paddle change colors to indicate that the game has gotten more difficult. In order to accomplish all of this, the Display component receives data from every component that draws something on the screen, as well as the Horizontal and Vertical Count to keep track of the positional data.

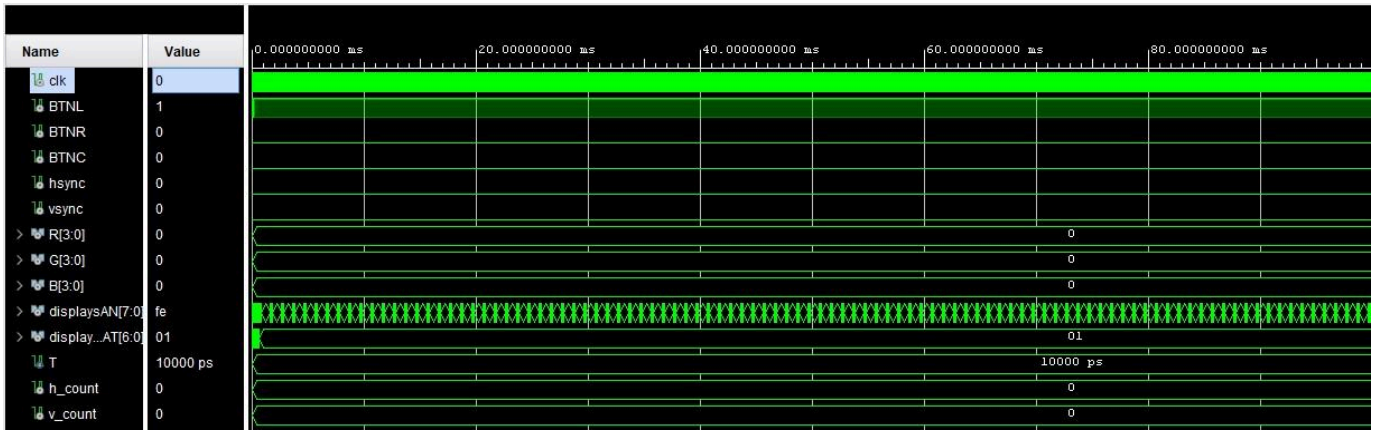
II. Experimental Setup

The setup we are using is the Nexys 100T board connected to a VGA display monitor. We will be programming the board in Vivado using VHDL. This implementation of the pong game allows one player to play. The player will be able to hit the ball from controlling a paddle that moves side to side. To control the paddle the player will use the right and left buttons on the Nexys board. As talked about before, to gain a point, the player will have to hit the ball with the paddle. The game is implemented for one to score as many points as they would like, although the challenge is the ball will continuously get faster making it more difficult for the player to hit the ball with the paddle. This game is continuous meaning the player plays until they want to stop. The player will be able to restart the game using the middle button on the Nexys board.



III. Results

For the testbench, we simply made one of the buttons high for the duration of the whole test bench. However, since this circuit has so many clock dividers, it was hard to get results that accurately reflected what the circuit acted like in a real world scenario. With a circuit of this type, it was easier and quicker to debug and make changes by generating the bitstream and programming the board directly. With that being said, a picture of the testbench timing diagram is provided.



The project works the way it was intended to work based on our updated design. The game is being displayed to the monitor via the VGA port. The ball and paddle component both also move as they were intended to. Though there were still issues with the life counter and the game over function. We originally planned on having a counter that would count the amount of lives the player had left and when that counter reaches zero it would then give the player a game over. This function was not working as intended so it was removed from the final design of the game. We also initially wanted to implement classic pong with two players but it would have been tough with trying to control both paddles and trying to use the VHDL board buttons to control the paddles.

Conclusions

So far we have learned about how a VGA display works, and how complicated simple games can be. There are a plethora of components to combine and correct in order for the VGA display to work correctly. Although there were some challenges, our group completed this project and was able to present it.

Some interesting refinements the group could make to improve the game in the future, or if we had more time, would be to add a second paddle for the normal two player version pong. Along with this second paddle adding a different way to control these paddles would be nice for the players. This different control could be a keyboard or a controller with joysticks. Another improvement we would like to implement is the “lives counter” and “game over” function. This would give the players more pressure when playing the game. Other changes the group could make to vary up the game would be to add some more power ups. Right now, the only powerup is that the ball speeds up after the player successfully paddles the ball a number of times. In the future, an addition of the ability for the ball to slow down for a powerup or having multiple balls that a player would have to keep track of would make the game more entertaining in the future. Finally, a last idea for a future change could be to turn this pong game into a block breaking game. In this game one would bounce the ball back into blocks that break when the ball hits them. Once all the blocks are broken the player wins and a new level is started. These ideas are just some of the ways we could expand on our pong game.

References

[1] Russell. (2023, September 19). *The go board - play pong on a VGA monitor*. Nandland. https://nandland.com/project-10-pong/

[2] Russell. (2023b, September 28). *The go board - VGA introduction (test patterns)*. Nandland. https://nandland.com/project-9-vga-introduction-driving-test-patterns-to-vga-monitor/

[3] Llamocca, Daniel. *VHDL Coding for Fpgas*, www.secs.oakland.edu/~llamocca/VHDLforFPGAs.html. Accessed 18 Apr. 2024.

[4] Hanna, Darrin, and Richard Haskell. “Introduction to Digital Design Using Digilent FPGA Boards.” *Digilent.Com*, Published by LBE Books, LLC, digilent.com/reference/_media/textbooks:intro_digital_design-digilent-vhdl_online.pdf. Accessed 18 Apr. 2024.