

California State University, Northridge  
ECE 524L - Advanced FPGA Design  
Fall 2022

Lab 4: VGA

October 9, 2022  
Instructor: Saba Janamian

Troy Israel

## Table of Contents

Introduction	3
Procedure	3
Results	3-5
Analysis/Conclusion	6

Introduction:

The purpose of this lab was to learn about the Video Graphics Array (VGA) technology and to implement a prebuilt design. In the prebuilt design we will reconfigure it so that we can alter the screen resolution parameters and the internal clock wizard. We will also be learning how the prebuilt design works and get a better understanding of how a VGA is implemented.

#### Procedure:

Firstly, will be downloading the prebuilt VGA modules from a digilent repository from Github. We will then need to change the master branch and update the submodules in our repo because the files we just downloaded are outdated. Beyond that, Vivado will add the top module, xdc constraint file and configure a clock wizard to meet the basic requirements for the VGA set up. We will adjust the XDC file to make sure that prewritten port mappings will be in the correct JA to JE slots, since the Zybo Z7 does not support a JB port. Adjustments inside the top module for the box logic and as well as the border sizes will also be made.

#### Results:

In testing this lab I first modified the XDC file because the Zybo Z7 only supports pins JA, JC to JE. I used JC and JD specifically because the VGA pmod needs two of them close to each other. I then generated the bitstream right away verifying the default configuration of the board. What I was in Fig 4.1, a small black square bouncing around a predefined region on the screen. The region was surrounded by checkerboard images and a region of various colors above.

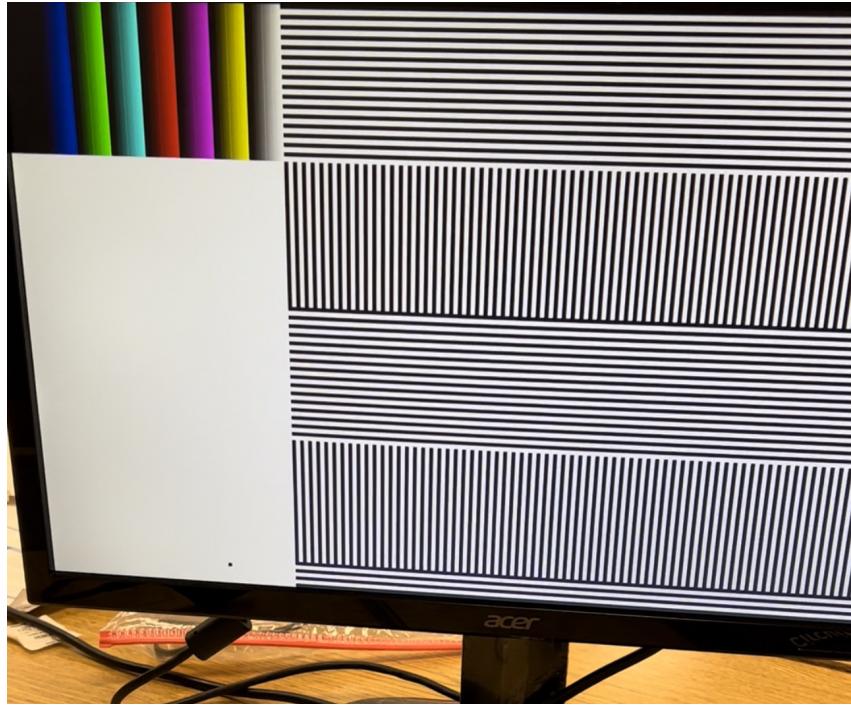


Fig 4.1 VGA initial setup output

There are three areas inside the top module code that ended up modifying. The first thing I did was remove the black and white squares and the rainbow color. To do this I commented out the entire test pattern logic section of the top module. The VGA supports three color outputs, vga\_red, green and blue. And I tested it out where I set the vga\_red to 1 when the pixel\_in\_box signal was 1. I then use the vga\_red signal as a driver for the vga\_blue and green signals. Doing so made it so that the screen was black and white only, with squares bouncing around the predefined region. The square would still be in the default region because I had not modified that parameters yet.

To change a lot of the parameters for the VGA square had to be done inside the top module. Specifically where all the constants are defined. Adjusting the box width would change the overall size of the square. Modifying the box min for the x and y coordinates would set the square in a specific starting location. I did find something interesting when it came to setting up the x/y max of the box. You have to change the value of the x max otherwise the box will not rise

up to the top of the screen. Setting the vga\_red to others getting 1 initially can change the background and the square color. Technically, there is a way to set whatever colors you like. All you would need to do is know what specific combinations will result in a specific color.



Fig 4.2 Box Size Adjustment and inverting the box color

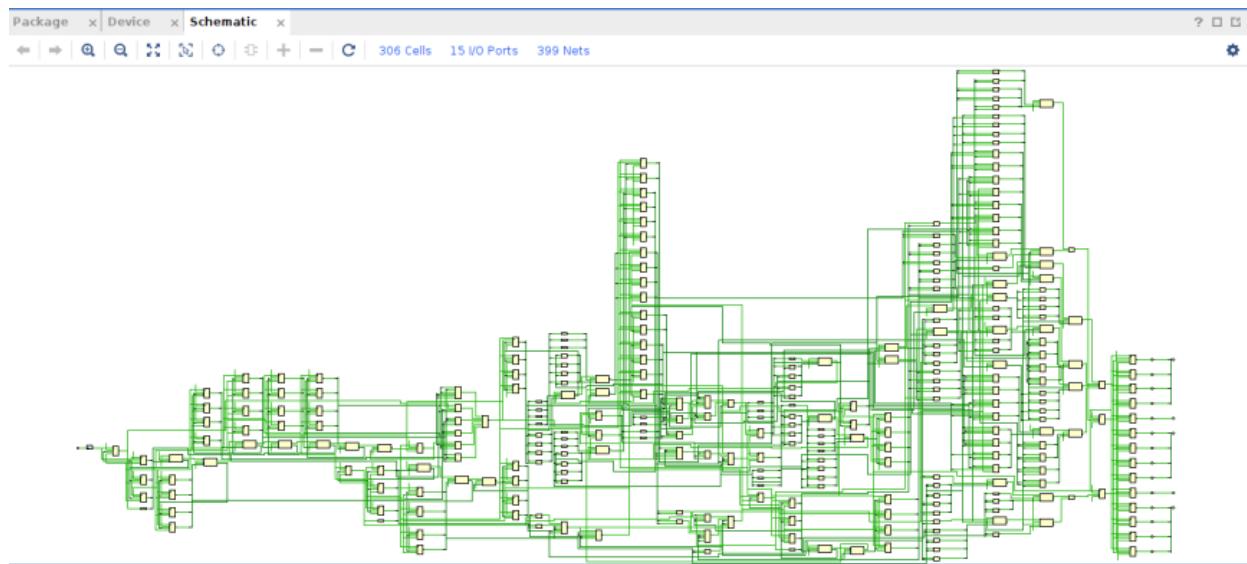


Fig 4.3 Implemented Schematic Design

## Conclusion:

For this lab we learned how to interface with a VGA pmod. We learned how to reconfigure existing HDL code to meet a specific task. For example, in this lab we could adjust the size of the square, make that square a circle or change the colors that would be displayed on the screen. I found it very interesting that the pre written code could support different types of display screens. While most commercial screens are sold at 1080p or better, there are still a lot of computer screens in a more industrial scene where they do not need to use the highest screen resolution possible. This lab was also very interesting from the standpoint of providing better insight into how not just VGA screens work but how a lot of computer screens work in general.

## Appendix:

<https://youtube.com/shorts/Vo3hSDCK2Bo?feature=share>

<https://github.com/csun-ece/fa22-e524-lab4-troykerim>