# Third Partial Computer Structure 1

**Authors:**
Felipe Jose Benitez Avilez
Laura Sofía Gómez Rosales
Fernando Mateo Valencia Gómez

**Profesor:**
Eduardo Zurek

**Date:** November 11, 2023

## Contents

1 Statement . . . 3
2 Problem Description . . . 4
3 Solution Requirements . . . 4
4 Solution Constraints . . . 4
5 Solution Proposals . . . 5
5.1 **First Solution:** . . . 5
5.2 **Second Solution:** . . . 5
6 Solution Evaluation and Selection . . . 6
6.1 **First Solution:** . . . 6
6.2 **Second Solution:** . . . 6
6.3 Conclusion: . . . 7
7 Eligibility Criteria . . . 7
8 Photographic Evidence of Implemented Solution . . . 7
9 Solution Impact . . . 8
10 Solution 2nd point . . . 9

# 1 Statement

Using an Altera DE2-115 board, implement a VGA protocol in VHDL, set the screen resolution to 640x350 and show an image centered on the screen with a size of 128x128 pixels. The image must be a picture of anyplace in Barranquilla outside the Uninorte campus and must be preprocessed with a Python script as per the example shown in class. The team must develop each one of the following items:

a. Describe the problem that you are going to solve.

b. Describe the design requirements of your solution.

c. Describe the restrictions that you consider when developing the design.

d. Propose two solution design alternatives to the problem.

e. Evaluate the alternatives and select the more convenient considering the criteria, restrictions, and design requirements.

f. What evaluation criteria did you apply to determine that your solution is correct?

g. Implement the selected alternative and include explanatory comments in your design.

h. Considering the context of the problem, describe how your proposed solution meets specified needs with consideration of factors such as: public health, safety, and welfare, as well as global, cultural, social, environmental, and economics. For this evaluation, you can select one of the factors.

## 2 Problem Description

In the context of the Computer Structure 1 course, the design and implementation of a video signal generation system using an Altera DE2-115 board is proposed, with the purpose of displaying an image on a VGA monitor.

The primary objective of this project is to develop a system capable of visualizing an image on a VGA monitor with a specific resolution of 640x350 pixels. The particularity lies in the fact that the image must occupy a space of 128x128 pixels in the center of the screen, thus ensuring its proper positioning and dimensions in relation to the total resolution. Importantly, this image must be a photograph taken in a location in Barranquilla other than the Uninorte campus.

To ensure the quality and appropriateness of the image to the visual requirements, a preprocessing of the image is required through a Python script. This script will be based on an example provided in class and will be responsible for adapting the image in a manner that complies with the criteria of size, resolution, and geographical content demanded.

## 3 Solution Requirements

The system must comply with the following specific requirements:

1. Video Signal Generation: The system must generate video signals that conform to a screen resolution of precisely 640x350 pixels. This ensures accurate display on a standard VGA monitor.
2. Image Positioning and Centering: The 128x128-pixel image must be centrally positioned on the 640x350-pixel screen. The center of the 128x128-pixel image should align with the center of the screen.
3. Selection of a Barranquilla Image: An image representing any location in Barranquilla must be used.
4. Image Preprocessing: You must utilize a provided Python script for the preprocessing of the selected image. This script will extract the RGB color components from the image and convert them into 8-bit binary format.

## 4 Solution Constraints

The constraints to be considered for the development of the solution are as follows:

1. Hardware Platform: The system must be developed using the Altera DE2-115 board as the hardware platform. This constraint ensures compatibility and project feasibility in terms of hardware.
2. Fixed Screen Resolution: The system must generate video signals that adhere to a fixed screen resolution of 640x350 pixels. This constraint determines the display standard

that must be met and is mentioned in the project statement.

3. Predefined Image Size: The image displayed on the screen must have a predefined size of 128x128 pixels, ensuring that the image is centered on the screen and meets the specific design requirement.
4. Geographical Location of the Image: An image geographically related to Barranquilla must be selected, excluding the Uninorte campus. The image chosen for our solution corresponds to a photo of the Barranquilla Metropolitan Stadium.
5. Image Storage Format: The RGB color components in binary format must be stored in three separate .txt files. This ensures that the color components are properly stored for use in generating video signals on the VGA screen.
6. Compliance with Technical Specifications of the VGA Protocol: The system must comply with the technical specifications of the VGA protocol for video signal generation. The appropriate parameters for the monitor model: ViewSonic with 640x480 resolution referenced on the page: VGA Parameters (VHDL) TechForum should be used.

# 5 Solution Proposals

## 5.1 First Solution:

First, the chosen image will be preprocessed in a Python script to obtain the three color vectors in RGB format, one for R, G, and B. These vectors will then be saved as eight-bit binaries.

Subsequently, in the program file within Quartus, using the RGB color vectors, we will copy and read the vectors to form the image within the space defined by the formula (re-v-h - 128)/2, which gives us the origin from where the image should start.

This will result in a color image in the center of the screen.

## 5.2 Second Solution:

First, the Python script will be modified to obtain a grayscale image vector. The values of this vector will be converted to binary and saved in an eight-bit format.

Then, in the program file within Quartus, using the grayscale vector, we will read the vector to form the image within the space defined by the formula (res-v-h - 128)/2, which gives us the origin from where the image should start.

This will ultimately give us a grayscale image in the center of the screen.

# 6 Solution Evaluation and Selection

## 6.1 First Solution:

This solution generates a color image in the center of the screen using the three RGB color vectors obtained from the Python script. This solution has the following advantages and disadvantages:

**Advantages:**

- The color image has higher quality than the grayscale image, as it preserves the original tones and hues of the photograph.
- The solution complies with all the requirements and constraints set for the project, respecting resolution, size, positioning, geographical location, and image preprocessing.

**Disadvantages:**

- The solution has higher complexity than the second solution, as it involves handling three color vectors instead of one, which may increase the development and implementation complexity and time.

## 6.2 Second Solution:

This solution will generate a grayscale image in the center of the screen using a single vector obtained from the Python script. This solution has the following advantages and disadvantages:

**Advantages:**

- The solution has lower complexity than the first solution, as it involves handling a single color vector instead of three, reducing the development and implementation complexity and time.
- The solution complies with most of the project's requirements and constraints, respecting resolution, size, positioning, geographical location, and image preprocessing.

**Disadvantages:**

- The grayscale image has lower quality than the color image, as it loses the original tones and hues of the photograph.
- The solution does not meet the requirement of using the RGB color components of the image, as it only uses a grayscale vector.

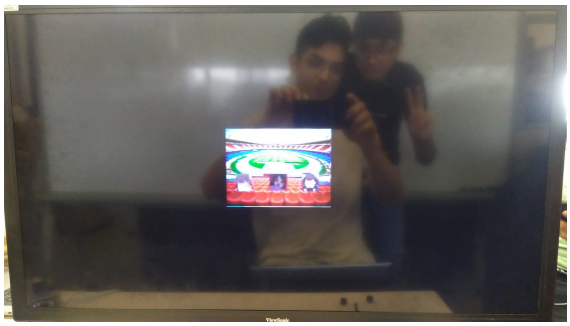
### 6.3 Conclusion:

After analyzing the available alternatives, we have decided to select the first proposed solution for implementation. This choice is based on the priority of maintaining the visual quality of the original image as a fundamental criterion. Although the second solution may seem simpler in terms of resources, as it requires only a single vector compared to the three needed in the first solution, the color version offers superior image quality. Conversion to grayscale, as proposed in the second alternative, would result in the loss of essential tones and hues needed to preserve the fidelity of the original image.

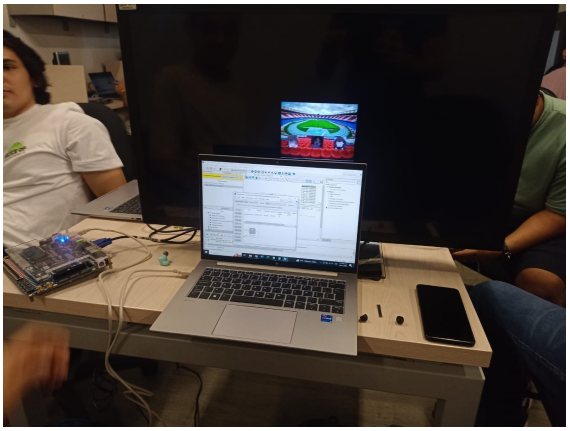
## 7 Eligibility Criteria

The primary criterion we applied to determine that the first solution is the most suitable lies in the need to preserve the quality of the original image. In this regard, the color solution retains the color channels (red, green, and blue), ensuring the highest possible image quality. Our primary focus is to maintain the highest visual fidelity of the image, which would not be effectively achieved in grayscale conversion. Therefore, the choice is based on the preservation of image quality as the primary criterion.

## 8 Photographic Evidence of Implemented Solution



(a) The following image displays the final outcome of the implemented solution, presented on the VGA monitor screen



(b) The image depicts the entire computer system, including the Altera board and connected monitor, all operating and functioning

# 9 Solution Impact

Image visualization plays a fundamental role in the current global context. With the proliferation of smart devices such as mobile phones and tablets, the ability to display information effectively and attractively has become essential. This capability not only impacts society and culture by enabling real-time visual communication, such as video calls and live streams, but also has significant implications in the fields of healthcare, education, and research.

Modern medicine increasingly relies on medical imaging for disease diagnosis and treatment. The quality of medical image visualization is critical to ensure that healthcare professionals can make accurate diagnoses and informed decisions. Therefore, improving image visualization processes has a direct impact on public health and people's well-being, and can help patients recover more quickly.

The implementation of solutions based on Field-Programmable Gate Arrays (FPGAs) offers additional advantages on a global scale. Reprogramming FPGA technology allows for the adaptation and continuous improvement of image visualization processes. This leads to greater efficiency and, in the long run, reduces operational costs and energy consumption. Additionally, FPGA-based solutions can also help reduce the environmental impact of image visualization by enabling more efficient resource utilization and the reuse of inherently reconfigurable FPGA technology.

In summary, the proposed solution meets the global image visualization needs by enhancing the quality of visualization in both everyday applications and the medical field. Furthermore, it contributes to sustainability by enabling long-term adaptability and efficiency through FPGA technology.

## 10 Solution 2nd point

In the execution of this project, the first step involved recording audio for each vowel. The main challenge was to minimize the duration of these audios to ensure the resulting vectors fit within the memory constraints of the NIOS II. This was achieved through meticulous audio editing, resulting in each vowel sound lasting 1 second. Consequently, when processed through the Python code, the size of each final vector was 4252 elements.

The program's logic revolves around identifying which key is pressed, storing this information in a register for subsequent comparison with each vowel. If the pressed key corresponds to a vowel, the program references the appropriate vector and starts playing the sound. This process continues until the final position of the vector is reached. Then, the key register value is reset to 0x00, and the system awaits the next key press.

It's important to highlight the use of exceptions in the program, particularly when using FPGA software, where it is crucial to indicate that these exceptions should be allowed for the system to function correctly.