

Implementación

Informe

Jhonny Alejandro Ortiz Osorio
C.C: 1001015092

Departamento de Ingeniería Electrónica y
Telecomunicaciones
Universidad de Antioquia
Medellín
Septiembre de 2021

Índice

1. Clases Implementadas	2
1.1. QImage	2
1.2. Adafruit NeoPixel	2
1.3. Escalamiento	2
1.4. Escritura	2
2. Estructura de las clases	3
2.1. Clase Escalamiento	3
2.2. Clase Escritura	3
3. Módulos de código	4
4. Circuito	4
4.1. Elementos Utilizados	4
4.2. Circuito Completo	5

1. Clases Implementadas

En la solución del problema se utilizaron en total 4 clases:

1.1. QImage

Es una clase establecida de Qt que nos permite manejar imágenes. Tiene métodos que nos permite leer una imagen, escalar el tamaño de una imagen o sacar información de los colores de cada pixel. En este caso solo se utilizó el método de leer imágenes y el método pixelcolor para sacar el color de un pixel en formato RGB.

1.2. Adafruit NeoPixel

Es una clase establecida de Arduino que nos permite controlar tiras de neo-pixel. De esta clase se utilizó el método setPixelColor que nos permite darle un color a un pixel en el formato de color RGB.

1.3. Escalamiento

Es una clase propia que permite procesar una imagen y ajustarla a un tamaño determinado. Esta clase el submuestreo y sobremuestreo de imágenes para poder procesar la imagen apropiadamente.

1.4. Escritura

Es una clase propia que escribe en un archivo de texto una información dada.

2. Estructura de las clases

2.1. Clase Escalamiento

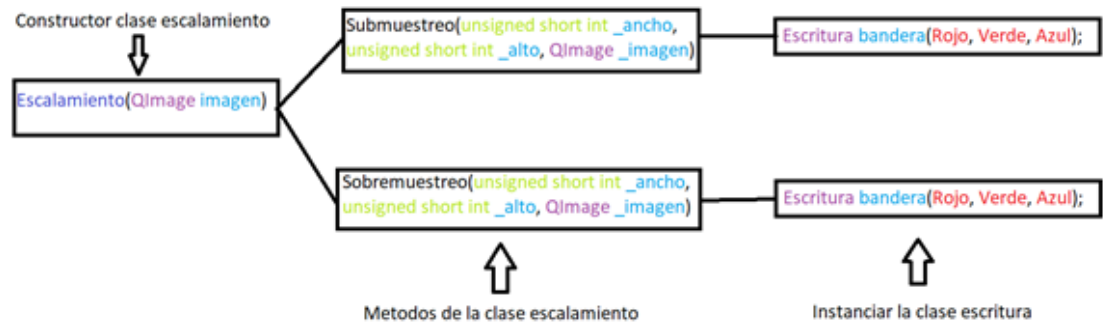


Figura 1: Estructura Clase Escalamiento

2.2. Clase Escritura

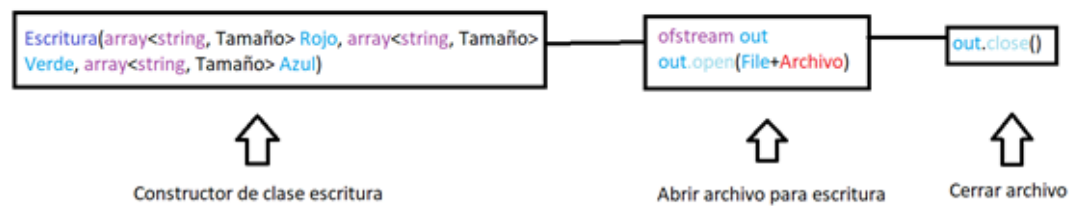


Figura 2: Estructura Clase Escritura

3. Módulos de código

En la primera parte del programa instanciamos la clase QImage para poder sacar toda la información de la imagen que se va a procesar. Después, se instancia la clase Escalamiento, entregando como parámetro al constructor el objeto QImage que se creó anteriormente, en la clase Escalamiento se procesa la imagen con el método de submuestreo o sobremuestreo, esto depende del tamaño de la imagen, por ultimo se instancia la clase Escritura dentro del constructor de la clase Escalamiento para poder escribir en un archivo toda la información de la imagen ya procesada.

4. Circuito

4.1. Elementos Utilizados



Figura 3: Tiras de NeoPixel

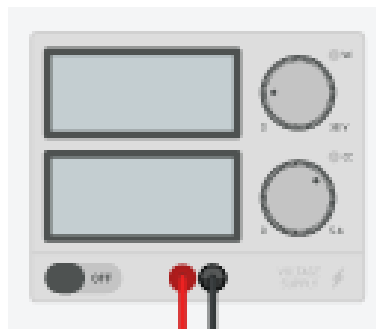


Figura 4: Suministro de Energia

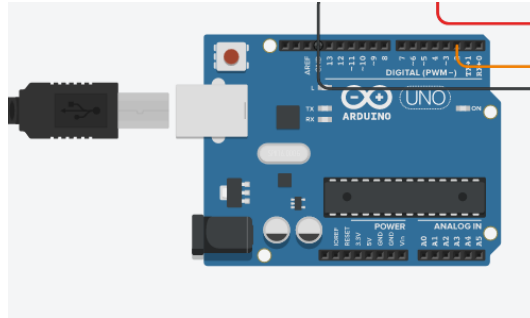


Figura 5: Arduino Uno

4.2. Circuito Completo

Para conectar todo el circuito se necesitaron 12 tiras de neopixeles, un Arduino y un suministro de energía. Se conecta el suministro de energía con un acople de tierras y la parte positiva se conecta al positivo de la primera tira de neopixeles, después de conectan todas las tierras y positivos de las tiras de neopixeles faltantes. El siguiente paso es conectar un pin digital del Arduino a la entrada de la primera tira de neopixeles y la salida de esta tira se conecta a la entrada de la siguiente tira de neopixeles.

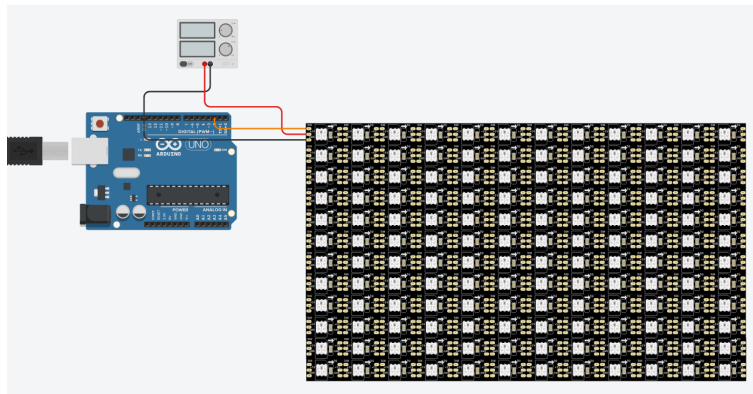


Figura 6: Circuito