

Parcial 2 Parte 2

Mateo Cardona Correa

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

Índice

1. Introduccion al Trabajo	2
1.1. Analisis del problema	2
1.2. Esquema Base	2
2. Modulo deCodigo	3
2.1. Estructura del circuito	5
3. Problemas presentados durante el desarrollo	6

1. Introduccion al Trabajo

El trabajo aqui presentado se realiza bajo la consideracion de la presentacion de un problema planteado en la normativa de la cotidianidad, presentando un problema tan comun como puede ser el desarrollo de un programa para transformar nociones, conceptos, programas y algoritmos digitales a un ambiente no digital, como puede ser una pantalla o un grupo de luces, en este caso, leds.

1.1. Analisis del problema

Las clases implementadas fueron:

- Clase base QImage: Clase implementada en el codigo de QT
- Clase base String: Clase implementada en el codigo de QT

1.2. Esquema Base

El esquema que se utilizo fue:

- Imagen descargada
- Submuestreo de la imagen
- Transformacion de la imagen mas pequeña en una serie de caracteres
- Lectura de los caracteres por un Arduino
- Transformacion de los caracteres en señales electricas
- De señales electricas a prender un panel de luces Led

2. Modulo deCodigo

Aqui se mostrara el codigo implementado en el programa QT

```
#include <iostream>
#include <fstream>
#include <string>
#include <QImage>

using namespace std;

int main(){
    string filename = "imagen.jpg";
    QImage im(filename.c_str());
    im = im.scaled(8,8);
    string R = "{}";
    string G = "{}";
    string B = "{}";

    for(int j = 0; j < 8; j++){
        for(int i = 0; i < 8; i++){
            R += to_string(im.pixelColor(i,j).red());
            G += to_string(im.pixelColor(i,j).green());
            B += to_string(im.pixelColor(i,j).blue());
            if(j == 7 && i == 7){
                R += "}";
                G += "}";
                B += "}";
            }else{
                R += ", ";
                G += ", ";
                B += ", ";
            }
        }
    }

    ofstream fout;
    fout.open("copiar.txt");
    fout << "//Pegue el siguiente codigo en tinkercad" << endl << endl;
    fout << "byte_arr[3][64]{" << endl;
    fout << "\t" << R << "," << endl;
    fout << "\t" << G << "," << endl;
    fout << "\t" << B << endl;
    fout << "};" << endl;
    fout.close();
```

```
        return 0;  
    }
```

Como se puede ver en el código superior, las clases String y QImage utilizadas (Pertenecientes a las librerías de igual nombre) se utilizan principalmente para:

- Cargar la imagen
- Escalar la imagen a un tamaño 8x8
- Transformar en carácter cada píxel de la imagen
- Asignación de cada píxel de la imagen al sistema RGB

2.1. Estructura del circuito

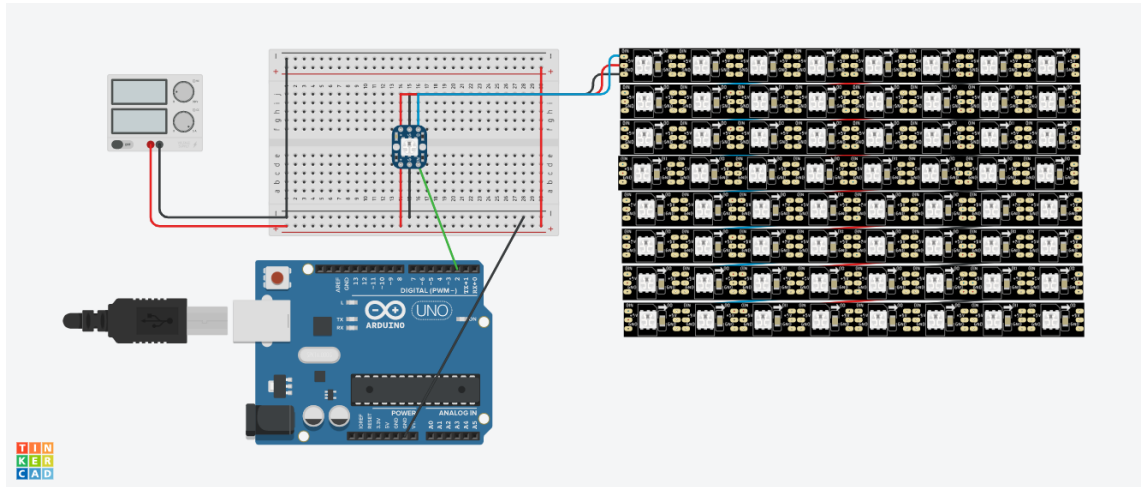


Figura 1: Circuito utilizado para la exposicion

3. Problemas presentados durante el desarrollo

Los principales problemas presentados durante el desarrollo fueron:

- La utilizacion eficiente del modelo RGB
- Guardar la cadena de caracteres en un txt
- La composicion propia de la cadena de caracteres
- El reescalado de la imagen
- La lectura de la cadena de caracteres en un Arduino