

# **Informe Parcial I**

Subtítulo

**Fabian Hoyos**

**Karen López**

**Yuribia Arroyave**

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

# Índice

<b>1. Sección Introductoria</b>	<b>2</b>
<b>2. Contenido</b>	<b>2</b>
2.1. Análisis del Problema . . . . .	2
2.2. Esquema para el desarrollo del algoritmo . . . . .	2
2.3. Algoritmo implementado . . . . .	2
2.4. Problemas que se presentaron . . . . .	4
2.5. Evolución y consideraciones . . . . .	4
<b>3. Inclusión de imágenes</b>	<b>4</b>

## 1. Sección Introductoria

## 2. Contenido

### 2.1. Análisis del Problema

Se necesita crear una aplicación para un puesto en una empresa llamada Informa 2 S.A.S, donde se requiere desarrollar una animación, la empresa presenta una dificultad y es que tiene un limitante de arduinos, los que poseen no cuentan con suficientes puertos digitales, se debe lograr una animación que muestre letras o una figura que el usuario ingrese. Dicha animación debe tener un patrón de leds. (64leds), se debe conectar y controlarlo con un arduino y el integrado 74HC595, la matriz de leds debe de ser de 8x8, se deben mostrar esos patrones y lograr realizar la conexión del sistema operativo con los 64 leds siguiendo la estructura dada. Para lograr esto se debe realizar una función que pida un patrón por la consola serial y mostrarlo. El objetivo principal es crear una función que muestre cuantos patrones se quieren mostrar y pida los patrones.

### 2.2. Esquema para el desarrollo del algoritmo

### 2.3. Algoritmo implementado

```
// 74HC595 = Pines de Arduino const int dato = 2; const int reloj = 3;
const int paso = 4;
//Prototipo de funciones void verificacion(int); //Encender todos los leds
void imagen(int); //Mostrar un patron /*void publik (); //Publicar secuencia de
patrones
    ingresados por usuario void bindec(int); //Convertir binario a decimal*/
// Contador de columnas int j = 0;
// Contador de duración de secuencia int k; int fila[8] = 127, 191, 223, 239,
247, 251, 253,
254;
// columnas de prueba que ayudaron a verificar
funcionamiento de leds y codigo int columnaV[8] = 255, 255, 255, 255, 255,
255,
255, 255; //encendido completo
int columnaF[8] = 60, 66, 165, 129, 165, 153,
66, 60; // Emoticon Feliz
int columnaN[8] = 60, 66, 165, 129, 129, 189,
66, 60; // Emoticon Normal
int columnaT[8] = 60, 66, 165, 129, 153, 165,
66, 60; // Emoticon Triste
// Se inicializa el 74HC595, y los puertos
digitales del arduino void setup() Serial.begin(9600);
pinMode(dato, OUTPUT); // dato pinMode(reloj, OUTPUT); // reloj pin-
Mode(paso, OUTPUT); // paso
```

```

pinMode(13, OUTPUT);
void loop() verificacion(columnaV); //Encendido total con
la función verificación //imagen(columnaV); //Encendido total con
funcion imagen //imagen(columnaN); //emoticon cara normal //imagen(columnaT);
//emoticon cara triste //imagen(columnaF); //emoticon cara feliz //Serial.println(binary,
analogValue);
// Funcion para imprimir un patron, definido por
el vector columna(N,T,F y lo que ingrese el
usuario) //Se usa el nombre del vector de columnas,
como puntero de la entrada cero del mismo
vector void imagen(int tipo[8]) for(k = 0; k<100; k++) for(int i=0; i<8; i++)
digitalWrite(reloj, LOW); //Se baja con un
pulso el reloj shiftOut(dato, paso, MSBFIRST, *(tipo
+j)); //Trae lo que hay en el vector columnas, por
medio del contenido de su puntero shiftOut(dato, paso, MSBFIRST, fi-
la[i]); //La
fila trae tiempos de duración de la secuencia digitalWrite(reloj, HIGH); j++;
delay(0.1); j = 0;
//Se usa el nombre del vector de columnas,
como puntero de la entrada cero del mismo
vector void verificacion(int columnaV[8]) for(int k = 0; k<100; k++) for(int
i=0; i<8; i++) digitalWrite(reloj, LOW); //Se baja con un
pulso el reloj shiftOut(dato, paso, MSBFIRST, *(columnaV
+j)); //Trae lo que hay en el vector columnas, por
medio del contenido de su puntero shiftOut(dato, paso, MSBFIRST, fi-
la[i]); //La
fila trae tiempos de duración de la secuencia digitalWrite(reloj, HIGH); j++;
delay(0.1); j = 0;
//conversor de binario a decimal /*void bindec(int binario) int result, res-
to=0; int digito[8]; //cout «İngrese binario: »; //cin» binario; for(int i=0; i<8; i++)
digito[i] =binario binario /=10; for (int i=7; i >=0; i-) result = (resto *2)+di-
gito[i]; resto = result; //cout«result; return(result); */
*/ //Funcion para mostrar una secuencia de
patrones void publik () unsigned int cant; Serial.print(.Escriba la cantidad
de patrones que
debe ingresar: »); Serial.println(cant); int patrones[8][cant]; for (int i=1;
i<=cant; i++) Serial.print(İngrese el patron», i, ». »); //cargara una fila de la
matriz patrones, con la
información de cada columna */

```

»2.4. Problemas que se presentaron

»2.5. Evolución y consideraciones

### »3. Inclusión de imágenes

»En la Figura (1), se presenta el logo de C++ contenido en la carpeta images.

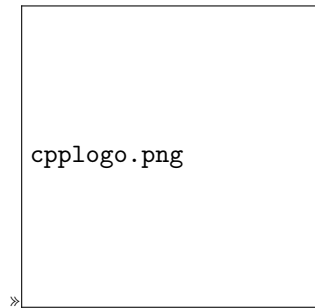


Figura 1: Logo de C++

»Las secciones (1), (2) y (3) dependen del estilo del documento.

### »Referencias