

Informática 2

Parcial 1

Johan Hernández Vargas

Kevin Jimenez Rincon

Manuela Gutierrez Rodriguez

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

Índice

1. Análisis del problema y consideraciones para la alternativa de solución propuesta.	2
2. Esquema donde describa las tareas que usted definió en el desarrollo del algoritmo.	2
3. Algoritmo implementado.	2
4. Problemas de desarrollo que presentó.	6
5. Evolución del algoritmo y consideraciones a tener en cuenta en la implementación.	6

1. Análisis del problema y consideraciones para la alternativa de solución propuesta.

El problema principal es identificar en el código la matriz de leds 8x8, además de esto, definir 3 funciones, una para verificar el funcionamiento de los 64 leds, otra para la impresión de un símbolo en esta matriz por el monitor serial y una más para imprimir un patrón ingresado desde el mismo monitor serial. Además de esto la elaboración de un manual completo y detallado en el cual, usuarios y programadores, puedan entender el funcionamiento de este circuito.

2. Esquema donde describa las tareas que usted definió en el desarrollo del algoritmo.

3. Algoritmo implementado.

El algoritmo que tenemos el día 20 de Abril para ejecutar el problema, consta de tres funciones, una para verificar el funcionamiento de los 64 leds, la otra es para recibir una matriz en donde se mostrará el patrón ingresado y una función para mostrar patrones. El código aún cuenta con errores los cuales lo hacen ir a bucles infinitos como el el switch o en la función 3 (patrones).

```
/*
const int SER = 2;
const int RCLK = 3;
const int SRCLK = 4;

int matrizLed [8][8]={
    {1,1,1,1,1,1,1,1},
    {1,1,1,1,1,1,1,1},
    {1,1,1,0,0,0,0,0},
    {1,1,1,1,1,1,1,0},
    {0,0,0,0,0,1,1,1},
    {1,1,0,0,0,1,1,1},
    {1,1,1,1,1,1,1,1},
    {0,1,1,1,1,1,1,0}
};

void relojregistro(int parametro){
    digitalWrite(parametro, 0);
    digitalWrite(parametro, 1);
    digitalWrite(parametro, 0);
}
```

```

void pordefecto(){
    int k;
    for(int m=7; m>=0; m--){
        for(int n=7; n>=0; n--){
            k=matrizLed[m][n];
            digitalWrite(SER, k);
            relojregistro(SRCLK);
            relojregistro(RCLK);
        }
    }
}

void verificacion(){
    for(int k=0; k<64; k++){
        digitalWrite(SER, 1);
        relojregistro(SRCLK);
        relojregistro(RCLK);
    }
}

void RecibirMatriz(){
    long int fila;
    int b;
    Serial.println("Para ingresar su patron, debe tener en cuenta que:");
    Serial.println("- El numero 1 se reflejara como el led encendido ");
    Serial.println("- El numero 0 se reflejara como el led apagado ");
    Serial.println("- Debe ingresar fila por fila ");

    for(int i=0; i<8; i++){
        Serial.print("Ingrese la fila ");
        Serial.print(i);
        Serial.println(":");
        while(Serial.available() == 0){}
        fila = Serial.read();
        for (int j=7; j>=0 ;j--){
            b=fila %10;
            matrizLed [i][j]=b;
            fila=fila /10;
        }
    }
}

void setup(){
    Serial.begin(9600);

```

```

pinMode(2, OUTPUT);
pinMode(3, OUTPUT);
pinMode(4, OUTPUT);

digitalWrite(2, 0);
digitalWrite(3, 0);
digitalWrite(4, 0);

/*
digitalWrite(SER, 1);
relojregistro(SRCLK);
relojregistro(RCLK);

    digitalWrite(SER, 0);
relojregistro(SRCLK);
relojregistro(RCLK);

    digitalWrite(SER, 1);
relojregistro(SRCLK);
relojregistro(RCLK);

    digitalWrite(SER, 0);
relojregistro(SRCLK);
relojregistro(RCLK);

    digitalWrite(SER, 1);
relojregistro(SRCLK);
relojregistro(RCLK);

    digitalWrite(SER, 0);
relojregistro(SRCLK);
relojregistro(RCLK);

    digitalWrite(SER, 1);
relojregistro(SRCLK);
relojregistro(RCLK);

    digitalWrite(SER, 0);
relojregistro(SRCLK);
relojregistro(RCLK);

    digitalWrite(SER, 1);
relojregistro(SRCLK);
relojregistro(RCLK);

```

```

        digitalWrite(SER, 1);
        relojregistro(SRCLK);
        relojregistro(RCLK);
        */
    }

    void loop(){
        int cpatron, tpatron;

        int opcion=1;

        while (opcion!=0){
            Serial.println("Bienvenido al menu de inicio");
            Serial.println("Presione el numero de lo que desea realizar");
            Serial.println("1. Verificar el funcionamiento de los 64 LEDs");
            Serial.println("2. Mostrar un patron");
            Serial.println("3. Secuencia de patrones");
            Serial.println("0. Salir");

            while(Serial.available() == 0){
                opcion = Serial.read();
            }

            switch (opcion){

                case 1:

                    verificacion();
                    //delay(3000);
                    break;

                case 2:

                    RecibirMatriz();
                    break;

                case 3:
                    Serial.println("Ingrese la cantidad de patrones");
                    while(Serial.available() == 0){
                        cpatron = Serial.read();
                    }

```

```

        Serial.println("Ingrese el tiempo entre patrones en segundos");
        while(Serial.available() == 0){
            tpatron = Serial.read();
        }

        for (int j=0; j<cpatron; j++){
            RecibirMatriz();
        }
        break;

    case 0:
        Serial.println("Ha cerrado el menu");
        break;

    default:

        Serial.println("Opcion no valida");
        break;

    }
}
}
*\

```

4. Problemas de desarrollo que presentó.

Uno de los principales problemas fue el uso de los pines digitales, teniendo en cuenta las restricciones planteadas. Además, contamos con problemas de distribución en el TinkerCad de cableado, leds y otros componentes.

Además de esto, hemos presentado problemas en la programación de las funciones, ya que contábamos con poco conocimiento previo de Arduino y ha sido muy complejo traducir el código de C++ a lenguaje Arduino.

5. Evolución del algoritmo y consideraciones a tener en cuenta en la implementación.

Estamos pendientes de la creación del manual para que el usuario sepa cómo trabajar correctamente este programa y para que otros colegas programadores puedan entender de manera acertiva el código implementado y poder hacer

correcciones sobre este. Al pasar de los días, encontramos documentación sofisticada para llevar a cabo las funciones requeridas en el parcial.