

Universidad de San Carlos de Guatemala

Facultad de Ingeniería

Escuela de Ciencias y Sistemas

Arquitectura de Computadores y Ensambladores 1

Ing. Otto Escobar

Aux. José Valdéz

Aux. Marielos Herrera



## **PRÁCTICA #1**

### **Visualización de Texto en Matrices LED 8x8**

Grupo #1

Diego Andrés Obín Rosales	201903865
Erick José André Villatoro Revolorio	201900907
Bryan Eduardo Gonzalo Méndez Quevedo	201801528
Elder Anibal Pum Rojas	201700761

Guatemala, 13 de Agosto del 2021

# MANUAL TÉCNICO

## PRÁCTICA #1

### **Descripción de la Práctica:**

- Que se adquieran conocimientos, que se aplique e interactúe con el microcontrolador Arduino

### **Objetivos de la Práctica:**

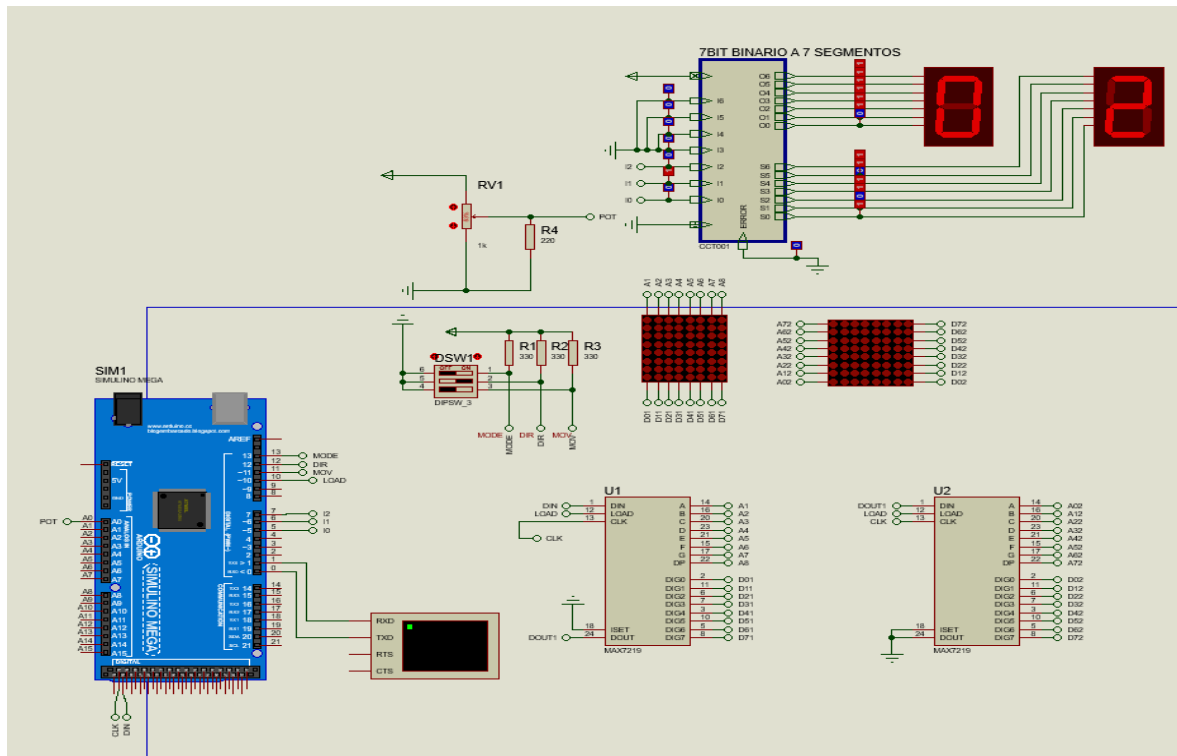
- Comprender el funcionamiento de las entradas y salidas, tanto digitales como análogas del microcontrolador Arduino.
- Comprender la configuración de las matrices LED.
- Conocer las funciones básicas de la salida serial del microcontrolador Arduino.
- Aplicar el lenguaje C para estructuras de control en Arduino, utilizando como editor el Arduino IDE.

### **Componentes Utilizados:**

1. Microcontrolador Arduino (Simulino en este caso)
2. Controlador MAX7219
3. Matrices Led 8x8 Red
4. Resistores de distintos valores
5. Botones
6. Software Proteus v8.10
7. Software Arduino IDE

## Diseño de Simulación en Proteus

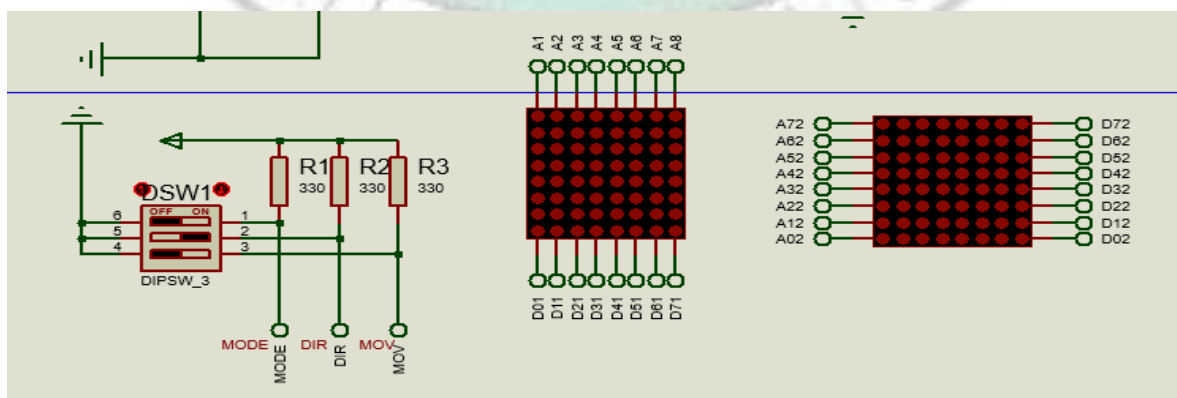
En la siguiente figura se muestra los componentes que se han incluido en el proyecto de Proteus para la simulación de esta práctica:



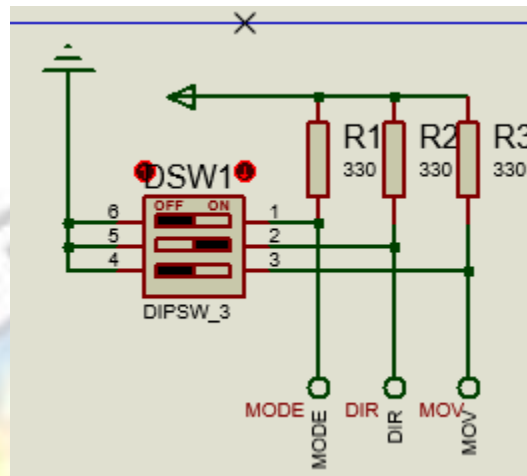
## Código de Arduino

Para la configuración de la práctica se realizó un programa en Arduino IDE, el cual contiene el código que fue cargado al Simulino Mega en Proteus para la ejecución de la práctica. Dicho código se encuentra en la sección de anexos.

## Letrero Matrices 8x8



Las matrices muestran el texto “TP1 – Grupo 1 – Seccion B”, el texto se desplaza de distintas maneras por ambos LEDS. Dependiendo de la configuración de los dip switches, esta cambiará.



La primera entrada controla lo siguiente:

Entrada 1	Función
0	Mostrar mensaje en movimiento.
1	Mostrar mensaje letra por letra sin movimiento. En una sola matriz.

La segunda entrada controla lo siguiente:

Entrada 2	Función
0	Desplazar mensaje de izquierda a derecha
1	Desplazar mensaje de derecha a izquierda

Con el botón y el potenciómetro se puede configurar la rapidez con la que se muestra la frase en el letrero.

## Estructura del Código

### **Bloque Inicial**

//Inicia la configuración de nuestra práctica. Básicamente posiciona la configuración por default de nuestras matrices mediante los pines del Simulino y además, inicia la comunicación serial mediante el puerto 9600.

```
void setup() {  
    matriz.setIntensity(7);  
    matriz.setPosition(0, 0, 0);  
    matriz.setRotation(0,3);  
    pinMode(MOV, INPUT);  
    pinMode(DIR, INPUT);  
    pinMode(MODE, INPUT);  
    pinMode(I0, OUTPUT);  
    pinMode(I1, OUTPUT);  
    pinMode(I2, OUTPUT);  
    //pinMode(POT, INPUT);  
    Serial.begin(9600);  
}
```

### **Bloque Loop**

//El bloque más importante, el loop básicamente es nuestro main debido a que este se encarga de validar los 4 tipos de entradas mediante los dos dip switch mediante condicionales, además de pasar entre la frase quemada "TP1-GRUPO1-SECCIONB" o comunicación serial. Además inicia nuestro método para leer el potenciómetro

```
void loop() {  
    int mov = digitalRead(MOV);  
    int dir = digitalRead(DIR);  
    int mode = digitalRead(MODE);  
    leerPotenciometro();  
}
```



```

if ( mode == 0){
    mensaje = "P1\40\x2D GRUPO1 - SECCION B";
} else {
    if(Serial.available() > 0){
        Serial.print("Ingresa un mensaje\n");
        mensaje = Serial.readString();
        Serial.print("Su mensaje es: " + mensaje);
    }
}

if(mov == 0){
    for(int i=0; i< mensaje.length(); i++){
        matriz.drawChar(3,1,mensaje[i],HIGH, LOW, 1);
        matriz.write();

        delay(750);
        matriz.fillScreen(LOW);
    }
} else {
    // Mostrar moviéndose
    if (dir==0){
        for (int i =ancho * mensaje.length()+ matriz.width() - 1 - espacio ; i >0 ; i-=
        velocidad){
            matriz.fillScreen(LOW); //Apagamos todos los LED. función dedicada para
            limpiar la pantalla.

            //LETRA POR LETRA

            int letra = i/ ancho;

            int x = (matriz.width() -1) - i %ancho;

            int y = (matriz.height() -8) / 2;

```

```
while (x + ancho - espacio >= 0 && letra >= 0) {
    if(letra < mensaje.length()){
        matriz.drawChar(x,y,mensaje[letra],HIGH, LOW, 1);
    }
    letra--;
    x -= ancho;
}
matriz.write();
delay(tiempo);
}
} else {
    for (int i =0; i < ancho * mensaje.length()+ matriz.width() - 1 - espacio ; i+=
    velocidad){
        matriz.fillScreen(LOW); //Apagamos todos los LED. función dedicada para
        limpiar la pantalla.
        //LETRA POR LETRA
        int letra = i/ ancho;
        int x = (matriz.width() -1) - i %ancho;
        int y = (matriz.height() -8) / 2;
        while (x + ancho - espacio >= 0 && letra >= 0) {
            if(letra < mensaje.length()){
                matriz.drawChar(x,y,mensaje[letra],HIGH, LOW, 1);
            }
            letra--;
            x -= ancho;
        }
        matriz.write();
        delay(tiempo);
    }
}
```

```
}  
}  
}
```

### **Bloque leerPotenciometro**

//Este método nos permite manejar la rapidez de los mensajes desplegados en la matriz mediante un potenciómetro. Básicamente la respuesta de nuestro potenciómetro es mandada mediante un puerto analógico (el A0 específicamente) y es mandado a nuestro MAX7219 y luego pasa a las matrices LED.

```
void leerPotenciometro(){  
    float val_pot = analogRead(A0);  
    if ( val_pot < 255 && val_pot >= 0 ) {  
        velocidad = 1;  
        digitalWrite(I0, HIGH);  
        digitalWrite(I1, LOW);  
        digitalWrite(I2, LOW);  
    } else if ( val_pot < 510 && val_pot >= 255 ) {  
        velocidad = 2;  
        digitalWrite(I0, LOW);  
        digitalWrite(I1, HIGH);  
        digitalWrite(I2, LOW);  
    } else if ( val_pot < 765 && val_pot >= 510 ) {  
        velocidad = 3;  
        digitalWrite(I0, HIGH);  
        digitalWrite(I1, HIGH);  
        digitalWrite(I2, LOW);  
    } else if ( val_pot < 1020 && val_pot >= 765 ) {  
        velocidad = 4;  
        digitalWrite(I0, LOW);
```



```
digitalWrite(I1, LOW);  
digitalWrite(I2, HIGH);  
} else if ( val_pot >= 1020 ) {  
    velocidad = 5;  
    digitalWrite(I0, HIGH);  
    digitalWrite(I1, LOW);  
    digitalWrite(I2, HIGH);  
}  
}
```

