

Cartel de Leds

Para realizar el cartel de leds se requirió de lo siguiente.

Requerimientos de Software:

- Proteus 8.9
- IDE Arduino

Componentes Utilizados:

- 1 Placa Simulino uno
- 2 Controladores MAX7219
- 2 Matrices 8x8
- 1 Dip Switch de 2 botones
- 1 Botón

Librerías utilizadas:

- MD_Parola.h
- MD_MAX72XX.h

Lógica utilizada:

El cartel esta compuesto por 2 matrices 8x8, las cuales muestran un solo mensaje. Dicho mensaje es: "TP1-GRUPO 4-SECCION B".

Además de esto las matrices mencionadas también darán la opción de mostrar el juego "Snake", como se detallará más adelante.

Antes de entrar a la explicación del código, se muestran las siguientes aclaraciones:

```
#define CP_PIN 10    //LOAD
#define DATA_PIN 11 //DIN
#define CLE_PIN 13  //CLK

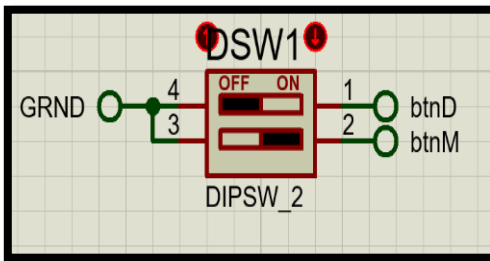
MD_Parola myDisplay = MD_Parola(HARDWARE_TYPE, DATA_PIN, CLE_PIN, CP_PIN, MAX_DEVICES);

const int buttonPin0 = 7; //Modo Texto
const int buttonPin1 = 6; //Dirección
const int buttonPin2 = 5; //Velocidad
```

Para la entrada de información de los botones se establecieron los pines del 5 al 7, del simulador de Arduino.

Continuando con la explicación. Como se mencionó anteriormente en “componentes”, el cartel utiliza 1 Dip Switch de 2 botones, el cual está configurado de la siguiente manera.

Dip Switch



- Botón de Dirección (btnD): Este botón permite cambiar la dirección en la que se desplazará el texto.
Estado 0 = Desplazar mensaje de izquierda a derecha.
Estado 1 = Desplazar mensaje de derecha a izquierda.

- Botón de Modo Texto (btnM): Este botón permite cambiar la forma en la que se mostrará el texto.
Estado 0 = Mostrar mensaje en movimiento.
Estado 1 = Mostrar mensaje letra por letra sin movimiento.

A continuación, el código utilizado para el botón de Dirección:

```
if (digitalRead(buttonPin1)==HIGH) {  
    myDisplay.setTextEffect(PA_SCROLL_LEFT , PA_SCROLL_LEFT);  
}else{  
    myDisplay.setTextEffect(PA_SCROLL_RIGHT, PA_SCROLL_RIGHT);  
}
```

Se utilizó como entrada de información para este botón, el pin 5 de la placa.

Se utilizó la condición “if”, para establecer el efecto de desplazamiento obtenidos de la librería parola. Todo esto según la entrada del pin.

Para el botón del Modo del texto, el código es el siguiente:

```
static uint8_t n = 1;  
if (digitalRead(buttonPin0)==HIGH) {  
    n=modoStatic(n);  
} else{  
    modoAnimado();  
    n=1;  
}
```

Se aplicó la misma lógica (respecto a las condiciones) que el botón de dirección . También se utilizó una variable como contador para la función del modo sin animación (modoStatic).

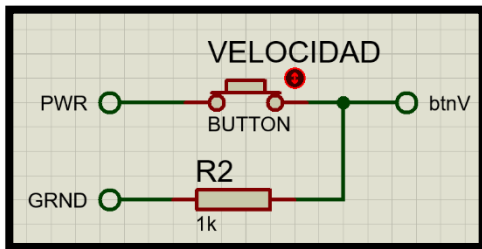
```
void modoAnimado() {
  myDisplay.setTextEffect(PA_SCROLL_LEFT, PA_SCROLL_LEFT);
  myDisplay.setTextBuffer(M[0].msg);
  if (digitalRead(buttonPin1) == HIGH) {
    myDisplay.setTextEffect(PA_SCROLL_LEFT, PA_SCROLL_LEFT);
  } else {
    myDisplay.setTextEffect(PA_SCROLL_RIGHT, PA_SCROLL_RIGHT);
  }
  int slide_scroll_speed = map(velocidad, 1023, 0, 400, 15);
  myDisplay.setSpeed(slide_scroll_speed);
}
```

Para el modo de texto animado, se utilizó diversos efectos de la librería parola.

```
int modoStatic(int cont){
  int n=cont;
  if (digitalRead(buttonPin1) == HIGH) {
    if (n < 20) {
      myDisplay.setTextEffect(PA_NO_EFFECT, PA_NO_EFFECT);
      myDisplay.setTextBuffer(M[n].msg);
      myDisplay.displayReset();
      delay(retardo);
      myDisplay.displayReset();
      n = (n + 1) % MAX_STRINGS;
    } else {
      n = 1;
      myDisplay.displayReset();
    }
  } else {
    if (n < 20) {
      n = 20;
    }
    myDisplay.setTextAlignment(PA_RIGHT, PA_LEFT);
    myDisplay.setTextEffect(PA_NO_EFFECT, PA_NO_EFFECT);
    myDisplay.setTextBuffer(M[n].msg);
    myDisplay.displayReset();
    delay(retardo);
    myDisplay.displayReset();
    n = (n + 1) % MAX_STRINGS;
  }
  return n;
}
```

Para el modo estático, no se utilizaron efectos. Se realizó un recorrido de un arreglo previamente declarado con los caracteres necesarios para mostrar el siguiente texto: “TP1-GRUPO 4-SECCION B”

Botón de Velocidad:



- Botón de Velocidad (btnV): Este botón permite cambiar la velocidad en la que se mostrará el texto. Estado 0 = Bajo. Estado 1 = Alto.

El código para el botón de velocidad utiliza la misma lógica que el resto de los botones.

Aquí se establece la velocidad para el texto animado y el tiempo de retardo para el texto si animación.

```
if (digitalRead(buttonPin2)==HIGH) {  
    velocidad = 80;  
    retardo= 600;  
}else{  
    velocidad = 1000;  
    retardo=1000;  
}
```