

**Nombre:** María Alejandra Rodríguez Roldán  
**Curso:** Electrónica Digital 2  
**Catedrático:** Pablo Mazariegos y José Morales

**Carné:** 21620  
**Sección:** 10  
**Fecha:** 6/10/2023

## Laboratorio 8 – Almacenamiento SD

Código utilizado:

```
//*****  
***  
// Universidad del Valle de Guatemala  
// BE3015 - Electrónica Digital 2  
// Laboratorio 8 - María Alejandra Rodríguez  
//*****  
***  
//*****  
***  
// Librerías  
//*****  
***  
#include <stdint.h>  
#include <SD.h>  
#include <SPI.h>  
  
//*****  
***  
// Definición de pines  
//*****  
***  
//Definición de SD CARD  
#define SCK A2  
#define MOSI A5  
#define MISO A4  
#define CS 32  
  
//*****  
***  
// Prototipos de función  
//*****  
***  
void dibujando(void);  
void dibujar(void);  
void guardar(String);
```

```
//*****  
***  
// Variables Globales  
//*****  
***  
String nombre; //Nombre del archivo que abre o crea  
const int ancho = 20; //Para crear la matriz que va a crear como texto  
const int altura = 10; //Para crear la matriz que va a crear como texto  
char imagen[altura][ancho]; //Matriz donde se va a guardar la "imagen"  
creada  
  
//*****  
***  
// Configuración  
//*****  
***  
void setup() {  
  Serial.begin(9600); // Inicializa la comunicación serial a 9600 baudios  
  SPI.setModule(0);  
  // Inicializa la comunicación con la tarjeta SD  
  if (!SD.begin(CS)) {  
    //Indica que algo pasó y no se inicializó correctamente  
    Serial.println("No se pudo inicializar la tarjeta SD.");  
    return;  
  }  
  //Indica que se inicializó correctamente  
  Serial.println("Tarjeta SD inicializada correctamente.");  
}  
  
//*****  
***  
// Loop  
//*****  
***  
void loop() {  
  //Muestra el menú al usuario en el monitor serial  
  Serial.println("Hola! Bienvenido!");  
  Serial.println("Menú de opciones para conocer como trabaja la SD:");  
  Serial.println("Puedes ver archivos preguardados: ");  
  Serial.println("1. Mostrar dibujo 1: Mike Wazowski ");  
  Serial.println("2. Mostrar dibujo 2: LEGO Man ");  
  Serial.println("3. Mostrar dibujo 3: Google ");  
  Serial.println("0 puedes guardar uno nuevo: ");  
  Serial.println("4. Dibuja tu propia imagen ");  
}
```

```
Serial.print("Selecciona una opción (1, 2, 3 o 4): ");

while (Serial.available() == 0) {
    // Espera a que el usuario ingrese una opción
}

int eleccion = Serial.parseInt(); // Lee la opción ingresada por el
usuario
delay(500);

//Opciones dependiendo lo que seleccioné
switch (eleccion) {
    case 1: {
        // Acciones para mostrar el dibujo 1
        Serial.println("Mostrando dibujo 1...");
        Serial.println("Aquí está Mike Wazowski ... ");
        delay(250);
        File dibujo1 = SD.open("dibujo1.txt");
        if (dibujo1) {
            Serial.println("dibujo1.txt");
            while (dibujo1.available()) {
                Serial.write(dibujo1.read());
            }
            dibujo1.close();
            delay(250);
        }
        break;
    }

    case 2: {
        // Acciones para mostrar el dibujo 2
        Serial.println("Mostrando dibujo 2...");
        Serial.println("Aquí está LEGO Man ...");
        delay(250);
        File dibujo2 = SD.open("dibujo2.txt");
        if (dibujo2) {
            Serial.println("dibujo2.txt");
            while (dibujo2.available()) {
                Serial.write(dibujo2.read());
            }
            dibujo2.close();
            delay(250);
        }
        break;
    }
}
```

```
}

case 3: {
    // Acciones para mostrar el dibujo 3
    Serial.println("Mostrando dibujo 3...");
    Serial.println("Aquí está Google ...");
    delay(250);
    File dibujo3 = SD.open("dibujo3.txt");
    if (dibujo3) {
        Serial.println("dibujo3.txt");
        while (dibujo3.available()) {
            Serial.write(dibujo3.read());
        }
        dibujo3.close();
        delay(250);
    }
    break;
}

case 4: {
    dibujar();
    guardar("nuevo_dibujo.txt");
    delay(250);
    break;
}

default: {
    Serial.println("Ohh no, está opción no es válida");
    Serial.println("Por favor, selecciona una opción válida (1, 2, 3 o 4).");
    delay(250);
    File error = SD.open("error.txt");
    if (error) {
        Serial.println("error.txt");
        while (error.available()) {
            Serial.write(error.read());
        }
        error.close();
        delay(250);
    }
    break;
}
}
```

```
//Limpiar el monitor serial
while (Serial.available() > 0) {
    Serial.read();
}
}

//*****
***
// Funciones
//*****
***
//Esta función es la que muestra en pantalla (monitor serial), la matriz
creada por el usuario
void dibujando() {
    //Imprimir la matriz en el monitor serial
    for (int i = 0; i < altura; i++) {
        for (int j = 0; j < ancho; j++) {
            Serial.print(imagen[i][j]);
        }
        Serial.println();
    }
}

//Esta función permite al usuario crear la matriz con el diseño que desea
void dibujar() {
    Serial.println("Dibuja tu imagen (utiliza 'X' para dibujar y ' ' para
dejar espacios entre carácter) ");
    Serial.println("Tienes espacio para dibujar de 20 columnas y 10 filas");

    for (int i = 0; i < altura; i++) {
        for (int j = 0; j < ancho; j++) {
            imagen[i][j] = ' ';
        }
    }

    for (int i = 0; i < altura; i++) {
        for (int j = 0; j < ancho; j++) {
            while (Serial.available() == 0) {
                // Espera a que el usuario ingrese un caracter
            }
            char caracter = Serial.read();
            if (caracter == 'X' || caracter == 'x') {
                imagen[i][j] = 'X';
            }
        }
    }
}
```

```
        Serial.print(imagen[i][j]);

    }
    Serial.println();
}

Serial.println("Imagen dibujada:");
dibujando(); //Llama a la función para que muestre matriz creada
}

//Función para guardar el archivo en la memoria SD
void guardar(String nombre) {
    File archivo = SD.open("prueba.txt", FILE_WRITE);

    if (archivo) {
        for (int i = 0; i < altura; i++) {
            for (int j = 0; j < ancho; j++) {
                archivo.print(imagen[i][j]);
            }
            archivo.println();
        }

        archivo.close();
        Serial.println("Imagen guardada correctamente en " + nombre);
    } else {
        Serial.println("No se pudo abrir el archivo para escritura.");
    }
}
```

Link a vídeo de funcionamiento: <https://youtu.be/BDNiQgFJOzI>

Link a GitHub: <https://github.com/aler21620/LAB8-21620>