

**Objetivo: Programar un sistema para la adquisición y registro de datos de 4 sensores de movimiento utilizando una tarjeta SD y Arduino Uno en la plataforma Wokwi.**

Materiales necesarios:

- Cuenta en la plataforma Wokwi
- Arduino Uno virtual en Wokwi
- 4 sensores de movimiento virtuales en Wokwi
- Tarjeta SD virtual en Wokwi
- Módulo de lectura y escritura para tarjeta SD virtual en Wokwi

### **Descripción de la práctica:**

El objetivo de esta práctica es programar un sistema que permita la adquisición y registro de datos de 4 sensores de movimiento en la plataforma Wokwi, y que guarde las lecturas en una tarjeta SD virtual cada X segundos en formato CSV con el ID del sensor, la hora de la lectura, el valor de lectura y otros datos que consideren importantes en un ejercicio de Internet de las cosas.

Para llevar a cabo la práctica, es necesario seguir los siguientes pasos:

1. Ingresar a la plataforma Wokwi y crear un nuevo proyecto.
2. Añadir el Arduino Uno virtual y los 4 sensores de movimiento virtuales al proyecto.
3. Añadir la tarjeta SD virtual y el módulo de lectura y escritura para tarjeta SD virtual al proyecto.
4. Realizar las conexiones virtuales entre los componentes siguiendo el diagrama proporcionado.
5. Programar el Arduino Uno virtual para leer los datos de los 4 sensores de movimiento y guardarlos en la tarjeta SD virtual cada X segundos.
6. Configurar el formato de los datos para que se guarden en formato CSV con el ID del sensor, la hora de la lectura, el valor de lectura y otros datos que consideren importantes en un ejercicio de Internet de las cosas.
7. Realizar pruebas del sistema para comprobar su funcionamiento.
8. Tomar una captura de pantalla de las conexiones realizadas en la plataforma Wokwi y enviarla al profesor como evidencia de la práctica realizada conjuntamente con el código fuente del programa y el fichero resultante de grabación.

### **Criterios de evaluación:**

La práctica será evaluada en función de los siguientes criterios:

- Correcta realización de las conexiones virtuales entre los componentes en la plataforma Wokwi.
- Programación correcta del Arduino Uno virtual para la adquisición y registro de datos de los sensores de movimiento.
- Formato correcto de los datos guardados en la tarjeta SD virtual en formato CSV.
- Pruebas realizadas con éxito del sistema.

- Entrega de una captura de pantalla que muestre las conexiones realizadas en la plataforma Wokwi.

### **Recomendaciones:**

Se recomienda a los estudiantes que, antes de comenzar con la práctica, realicen una investigación sobre los sensores de movimiento y su funcionamiento, así como sobre el formato CSV para el almacenamiento de datos. Además, se sugiere que realicen pruebas en cada etapa del proceso para comprobar que todo funciona correctamente antes de continuar con la siguiente fase. También se recomienda que revisen su captura de pantalla antes de enviarla para asegurarse de que se vean claramente las conexiones realizadas en la plataforma Wokwi.

## **Especificación del formato CSV que se podría utilizar para esta práctica:**

El formato CSV (Comma-Separated Values) es un formato de archivo que se utiliza para almacenar datos tabulares en texto plano. En este formato, cada línea del archivo representa una fila de datos, y los campos de cada fila están separados por comas.

Para esta práctica, se podría utilizar el siguiente formato para los datos a almacenar en la tarjeta SD:

- ID del sensor: Un número que identifica de manera única cada sensor. Por ejemplo, el sensor 1 tendría el ID 1, el sensor 2 tendría el ID 2, y así sucesivamente.
- Hora de la lectura: La hora a la que se realizó la lectura, en formato hh:mm:ss.
- Valor de lectura: El valor leído por el sensor en el momento de la lectura.
- Otros datos: Se podrían incluir otros datos relevantes para el ejercicio de Internet de las cosas, como la fecha, la ubicación del sensor, etc.

Un ejemplo de línea de datos en formato CSV sería:

```
"1,09:30:45,23.5,2023-03-02,\n"
```

En este ejemplo, el sensor tiene ID 1, la lectura se realizó a las 9:30:45, el valor leído fue de 23.5, la fecha es el 2 de marzo de 2023.

Es importante tener en cuenta que, para que los datos se puedan leer correctamente desde un archivo CSV, es necesario que los campos estén separados por comas y que no se utilicen comas u otros caracteres especiales dentro de los campos de datos.

Es importante también que el final de la línea incluya el elemento "\n" o NEW LINE que indica una nueva línea de texto, y que cuando la conversión, generará una nueva entrada de línea en los registros de los datos leídos ( por ejemplo una nueva línea Excel).

Para añadir un salto de línea (`\n`) a un archivo, puedes utilizar la función `println()` en lugar de la función `print()`. La función `println()` es similar a `print()`, pero añade automáticamente un carácter de salto de línea al final de la cadena que se está escribiendo en el archivo.

Ejemplo que muestra cómo utilizar `println()` para escribir una cadena con un salto de línea en un archivo:

```
#include <SD.h>

File dataFile;

void setup() {
  // Inicializar la comunicación serie y la tarjeta SD
  Serial.begin(9600);
  while (!Serial) {
    ; // Esperar a que se establezca la conexión serie
  }
  Serial.println("Inicializando la tarjeta SD...");

  if (!SD.begin(4)) {
    Serial.println("Error al inicializar la tarjeta SD.");
    return;
  }
  Serial.println("Tarjeta SD lista para usar.");

  // Abrir el archivo de datos
  dataFile = SD.open("datos.csv", FILE_WRITE);

  // Escribir una línea de datos en el archivo
  dataFile.print("ID del sensor, Hora de la lectura, Valor de lectura");
  dataFile.println(); // Añadir un salto de línea
```

```
}
```

```
void loop() {
```

```
    // Leer los datos de los sensores y guardarlos en el archivo
```

```
    // Cerrar el archivo para liberar recursos
```

```
    dataFile.close();
```

```
    // Esperar un tiempo antes de volver a guardar los datos
```

```
    delay(5000);
```

```
}
```