

# Implementación Arduino: Identificador de Ondas

Santiago Cortés Osorio

s.cortes@udea.edu.co

Sara María Hincapié Hincapié

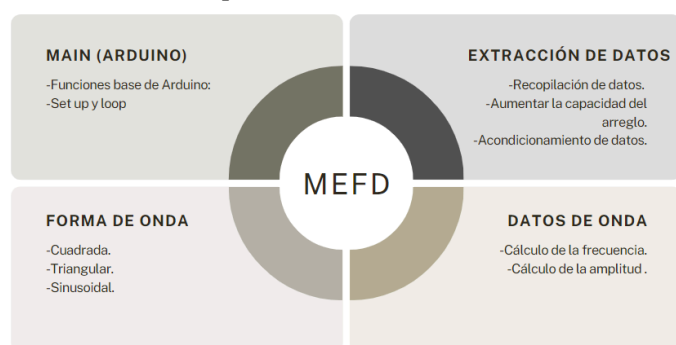
sara.hincapie1@udea.edu.co

**Resumen:** Este proyecto se centra en el análisis de señales utilizando un Arduino y un generador de señales. Se requiere la adquisición y procesamiento de una señal para determinar su amplitud, frecuencia, y forma de onda. Esto se logra mediante la captura de datos, su análisis mediante algoritmos específicos, y la visualización de los resultados en una pantalla LCD.

## 1) Diseño estructural

Primero se desestimó la problematización subyacente, identificando conceptos clave como el tipo de onda, sus parámetros y la extracción de información relevante. Con base en esto, se delimitó el problema y se definieron indicadores para las funciones necesarias.

Adicionalmente, en la delimitación del programa, se estableció un conjunto de funciones específicas para el montaje en Arduino, abarcando la configuración del generador de señales, la adquisición de datos, y la visualización en la pantalla LCD.



## 2) Gráfico de procesos:



## 3)Análisis del problema:

A continuación se presenta más detalladamente la especificación de cada una de las funcionalidades planteadas en el programa.

### 3.1 Main:

Se completó la conexión del circuito utilizando un LCD I2C para minimizar el cableado. Además, se incorporó un osciloscopio para visualizar las ondas, reemplazando la necesidad del monitor serial para la depuración del código. Los pulsadores están configurados para manejar la adquisición de datos, asegurando que el sistema solo recoja información cuando se activa el pulsador correspondiente.

En el void loop, el sistema se activa al detectar la presión del pulsador. Se inicia la adquisición de datos y se calcula la información clave: la amplitud y la frecuencia de la señal, que será mostrada en el LCD. Además, el programa identifica el tipo de onda (cuadrada, senoidal, triangular) basándose en el patrón de datos recolectados.

### 3.2 Extracción de datos:

Con el fin de manejar eficientemente la recolección de datos en una arquitectura de software con grandes limitaciones, se implementó una estrategia basada en la gestión de la memoria dinámica. Por eso se pautó una función que expande el arreglo en el cual se almacenan los datos de las señales; este va en aumento en bloques de 20 elementos cuando es necesario. Al mismo tiempo, se limitó la recolección de datos a uno cada 10 milisegundos, lo que equivale a un dato cada 104 microsegundos. Esto permite realizar un equilibrio entre la precisión de los datos y el uso de la memoria.

Otra funcionalidad que se añadió está relacionada con la forma de onda, específicamente para diferenciar si la señal es sinusoidal, para realizar el ajuste de la función en caso de que no comience en 0. Para esto se usaron dos funciones: una de ellas se encarga de alinear los datos de manera que la señal empiece en el punto correcto; posteriormente, agrega datos adicionales si es necesario para que la longitud de la señal sea adecuada. La otra función es la encargada de realizar el ajuste horizontal de la señal; está calcula cuánto necesita desplazarse la señal en el tiempo para que coincida con el valor esperado de la onda senoidal en ese punto.

### 3.3 Forma de la onda:

Se planea que el programa pueda distinguir entre 3 tipos de onda; en caso de que no sea alguno de estos, se clasifica como onda desconocida.

Primeramente, se analiza cómo identificar si una onda es cuadrada; para esto se evalúan los valores de la señal, y se hace un conteo de la cantidad de niveles distintos. Para que una señal sea clasificada como cuadrada, debe alternar entre dos valores únicos.

Por otro lado, para que el programa identifique si una onda es triangular, se plantea comparar los valores que arroje la onda por medio de las diferencias, puesto que estas deberían seguir una secuencia constante. Esto quiere decir que esta onda tiende a tener rampas lineales ascendentes y descendentes; por lo tanto, las diferencias entre valores consecutivos deben ser consistentes en una secuencia que alterna entre aumentos y disminuciones lineales.

Finalmente, para determinar si una señal corresponde a una senoidal, se capturan los datos de la onda; en caso de que la onda no inicie en 0, se crea artificialmente un pedazo de onda para simular su inicio en 0. Posteriormente, se compara la señal ya ajustada con una señal senoidal; se busca comparar ambas señales y acumular el error. Por lo tanto, se espera que este error sea menor que al compararlo con otro tipo de ondas.

### 3.4 Datos de onda:

Con el fin de obtener información específica de la onda, en este caso, la amplitud y la frecuencia, se realiza el cálculo de ambas en funciones distintas.

La frecuencia se define como cuántas veces la onda se repite en un periodo de tiempo determinado; en este contexto, se cuenta el número de transiciones entre valores positivos y negativos para identificar ciclos completos de la señal. Cada transición de un estado positivo a negativo o viceversa se cuenta como un cambio de estado. Al final del análisis, se divide el total de ciclos detectados por 4 para obtener la frecuencia en términos de ciclos por segundo.

Por otra parte, para la amplitud se identifican los valores máximos y mínimos de la onda; se ajusta el valor de la amplitud para que sea múltiplo de 100. Se realiza una sumatoria entre el valor mínimo y máximo y se divide entre dos para asegurar que se mantenga el rango de la distancia.