Juan David Castaño Hernández

Diego Alexander Parada Ortega

Curso: Informática II

Ingeniería Electrónica

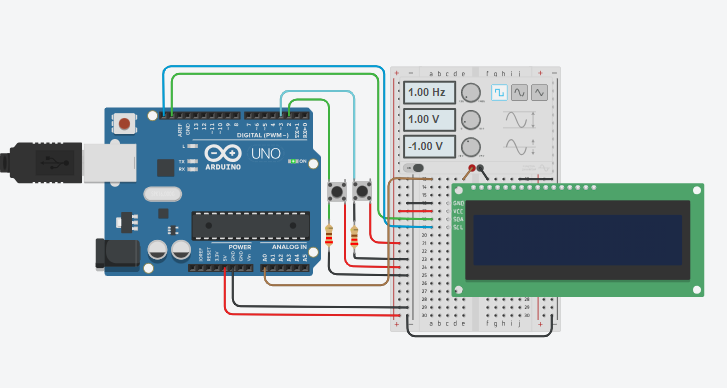
Universidad de Antioquia

Detección de Tipos de Ondas en Generador de Funciones

(Desafío 1)

**Introducción.**

Este documento describe la forma en que se decidió identificar diferentes tipos de ondas (cuadradas, senoidales y triangulares) a partir de los valores leídos por un generador de funciones utilizando Arduino. Se presenta la lógica empleada para diferenciar las ondas en base a sus características fundamentales, así como una descripción de lo que hará el programa para realizar dicha detección.



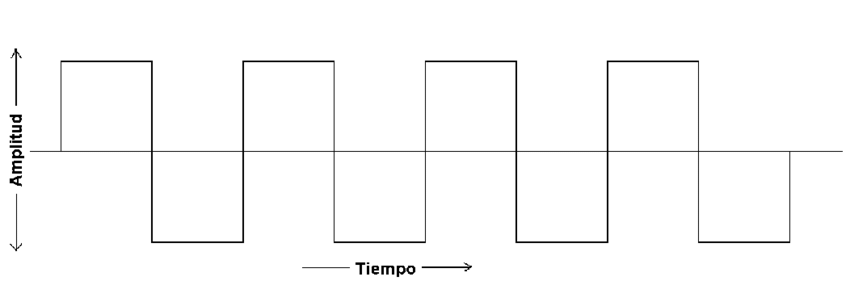
**Análisis del problema.**

Para el desarrollo del problema primero se pasó por un proceso de análisis en donde el principal objetivo era encontrar una manera de almacenar los datos para así poder implementar ciertas operaciones para determinar lo que nos pide el problema y es diferenciar los diferentes tipos de ondas.

Para diferenciar entre las distintas formas de ondas (cuadrada, senoidal y triangular), se analizan las propiedades de los valores capturados por el generador de funciones. Las características principales que se utilizan son las siguientes:

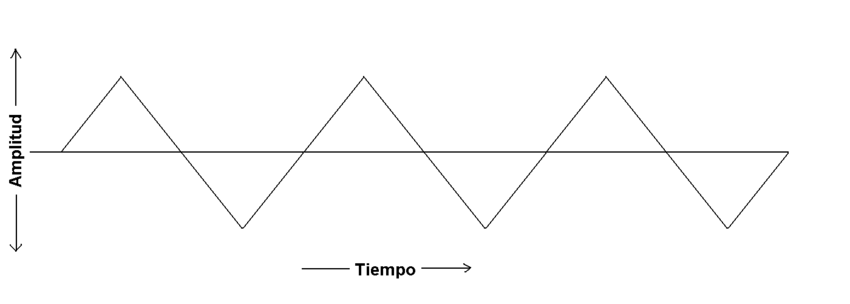
**Onda Cuadrada.**

Las ondas cuadradas se caracterizan por tener únicamente dos niveles de voltaje (uno positivo y uno negativo), alternando entre ambos sin valores intermedios. Para identificarlas, el programa detecta si los valores de la señal permanecen estables en dos niveles diferentes por periodos prolongados antes de cambiar bruscamente.



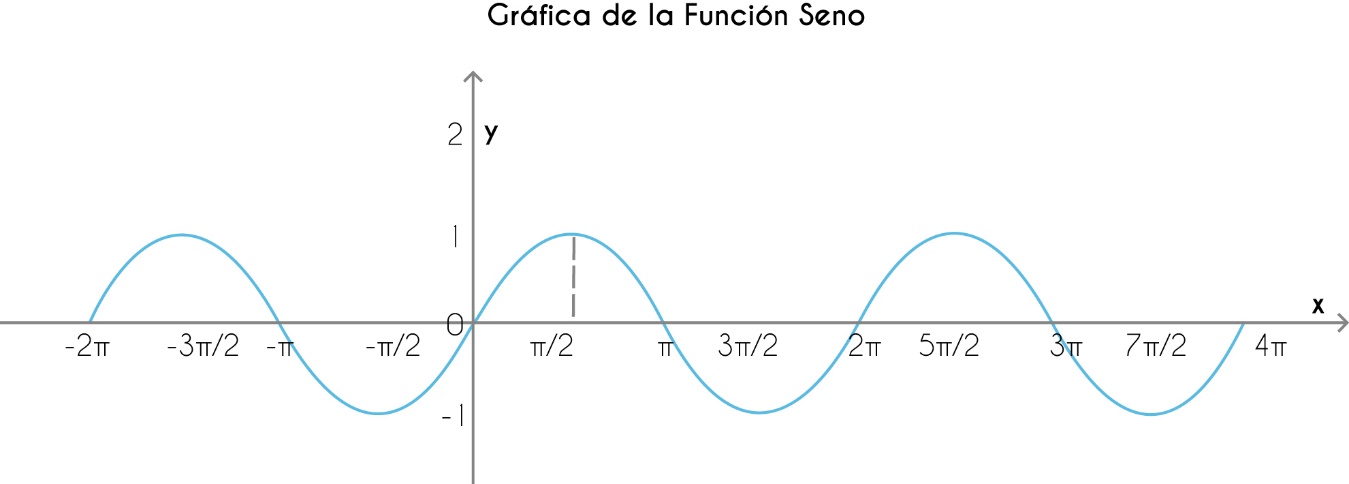
**Onda Triangular.**

Las ondas triangulares presentan una pendiente constante entre los puntos mínimos y máximos, lo que significa que los valores cambian a una tasa constante entre estos extremos. El programa calcula las diferencias entre los valores consecutivos de la señal y verifica si está pendiente se mantiene uniforme para identificar una onda triangular.



**Onda Senoidal.**

Las ondas senoidales tienen una forma más suave, con valores que cambian gradualmente sin los saltos bruscos de las ondas cuadradas ni la pendiente constante de las ondas triangulares. Si una señal no cumple los criterios de cuadrada o triangular, se clasifica como senoidal.



**Inicio del desarrollo del desafío**

El desarrollo del desafío se dividió por partes la primera fue el montaje de nuestro propio circuito para el desarrollo del programa

Diagrama, Esquemático

Descripción generada automáticamente

decidimos utilizar una lcd I2C hacer mas eficiente el montaje ya que requiere de menos conexiones.

**Descripción del Funcionamiento del Programa.**

Este código de Arduino tiene como objetivo capturar y analizar una señal analógica proveniente de un generador de funciones y mostrar en una pantalla LCD el tipo de onda (cuadrada, senoidal o triangular), la frecuencia y la amplitud de la señal.

**Configuración Inicial**

El código comienza definiendo variables clave, como los pines de entrada, botones y el pin analógico que recibe la señal. También se inicializa una pantalla LCD para mostrar los resultados, y se crea un arreglo dinámico grafica que almacena las muestras de la señal leída. Se establece el tamaño inicial del arreglo en 10, aunque puede aumentar si es necesario.

**Captura de Señal**

Cuando se presiona el botón de inicio, el sistema comienza a leer la señal analógica en el pin A0. Los valores leídos se almacenan en el arreglo dinámico, y se monitorean los valores máximos y mínimos de la señal para calcular la amplitud posteriormente. Durante la lectura, el código también detecta los cambios de estado de la señal (alto o bajo) para contar los ciclos, los cuales se utilizan para calcular la frecuencia.

Si el arreglo dinámico alcanza su límite, el código intenta duplicar su tamaño usando la función aumentoArreglo(), hasta que se alcanza un máximo de 300 muestras. Si no hay suficiente memoria disponible, se detiene la captura de datos.

**Procesamiento de la Señal**

Cuando se presiona el botón de fin, el código deja de leer la señal y calcula la frecuencia en Hz y la amplitud en voltios. Ambos valores se muestran en la pantalla LCD. Después, el código llama a la función tipoOnda(), que analiza los datos almacenados en el arreglo y determina si la señal es cuadrada, senoidal o triangular. El tipo de onda también se muestra en la pantalla LCD.

* **Onda Cuadrada**: Se verifica si la señal alterna entre dos valores diferentes de manera regular.
* **Onda Senoidal**: Se compara la señal capturada con una forma de onda senoidal esperada y se revisa si la diferencia entre ambas es pequeña.
* **Onda Triangular**: Se analiza el comportamiento creciente y decreciente de la señal.

**Gestión de Memoria**

El código emplea memoria dinámica para ajustar el tamaño del arreglo que almacena las muestras de la señal. Si se llena el arreglo, se duplica su tamaño hasta un máximo de 300 elementos. Después de que se han procesado los datos y mostrado los resultados, se libera la memoria utilizada para el arreglo y se reinicializa para futuras lecturas.

Este enfoque permite manejar eficientemente la captura y análisis de señales de diferentes frecuencias y formas, mientras optimiza el uso de la memoria disponible en el microcontrolador.

**Dificultades del desafío.**

En lo que llevamos del progreso del desafío se nos han presentado muchos inconvenientes el mas grande ha sido el análisis de cómo se podría implementar un programa que interprete los datos arrojados por el Arduino y así manejarlos y poder concluir el tipo de señal y demás.

A medida que hemos progresado hemos tenido muchos problemas con respecto al uso de la memoria dinámica ya que teníamos muchas inconsistencias y nos generaba un colapso del código.

Una dificulta que se convirtió en una de las más grandes fue la diferencia entre la onda senoidal y la triangular, nos llevó mucho tiempo de análisis, ensayo y errores.