DESAFIO I

INFORMATICA II

ALFONSO DIAZ LINA YOJANA  
RINCON SALDARRIAGA SEBASTIAN

GUERRA ANIBAL JOSE

UNIVERSIDAD DE ANTIOQUIA

2024-2

Para abordar el problema planteado, se requiere un análisis más detallado de las funcionalidades necesarias y una solución estructurada en términos de procesamiento de señales y programación en Arduino. A continuación, se presenta un esquema de la solución, sin utilizar código, pero detallando los pasos clave:

1. Montaje en Tinkercad

El sistema propuesto debe estar montado en la plataforma Tinkercad con los siguientes componentes:

1. Arduino UNO: Es el microcontrolador encargado de gestionar las entradas y salidas.
2. Generador de señales: Este generador emite señales analógicas que varían en frecuencia, amplitud y forma de onda. La señal se conectará a uno de los pines analógicos del Arduino para su lectura.
3. Pulsadores: Se utilizarán dos pulsadores:
   * El primero inicia la adquisición de datos. Cuando el usuario lo presiona, el programa comienza a capturar y almacenar los datos de la señal.
   * El segundo pulsador se utilizará para detener la adquisición de datos y visualizar los resultados en la pantalla LCD.
4. Pantalla LCD: Utilizando una pantalla LCD, se mostrará la frecuencia, amplitud y el tipo de señal detectada.

Conexiones:

* Los pulsadores se conectarán a pines digitales, con resistencias pull-down para evitar interferencias eléctricas.
* La pantalla LCD se conectará utilizando la librería adecuada, como Adafruit\_LiquidCrystal.h, que simplifica el manejo de la pantalla desde el código.
* El generador de señales se conectará a uno de los pines analógicos del Arduino para capturar los datos de la señal en tiempo real.

1. Registro de la señal en un arreglo de apuntadores

El programa debe capturar los datos de la señal en tiempo real y almacenarlos en un arreglo de apuntadores. Este método es eficiente porque permite gestionar de forma dinámica una gran cantidad de datos, almacenando solo los valores relevantes.

La captura de la señal debe realizarse en intervalos regulares de tiempo, lo que genera una serie de valores que representan la señal. Estos valores se almacenan en un arreglo que luego será utilizado para analizar las características de la señal, como la frecuencia, la amplitud y el tipo de onda.

Cada vez que la señal cambia, por ejemplo, en su frecuencia o amplitud, el programa debe almacenar estos cambios para analizarlos posteriormente.

1. Cálculo de la frecuencia y amplitud de la señal

Frecuencia: Se determina calculando cuántas veces la señal cruza el valor cero, es decir, el punto donde la señal cambia de positivo a negativo o viceversa. Cada cruce por cero representa un ciclo de la señal. Para calcular la frecuencia:

1. Se detecta cada cruce por cero en los datos almacenados.
2. Se mide el tiempo entre dos cruces consecutivos.
3. La frecuencia se calcula como la inversa del tiempo entre esos cruces (es decir, el número de ciclos por segundo).

Este método es preciso porque toma en cuenta el comportamiento natural de una onda periódica, asegurando que cualquier tipo de señal, ya sea senoidal, cuadrada o triangular, se mida correctamente.

Amplitud: Se calcula tomando los valores máximos y mínimos de la señal durante el período de adquisición. La amplitud es simplemente la mitad de la diferencia entre el valor máximo y el valor mínimo de la señal. Esto es crucial para determinar la "altura" de la onda.

El proceso consiste en:

1. Encontrar el valor más alto y bajo de la señal capturada.
2. Restar el valor mínimo del valor máximo.
3. Dividir este resultado entre dos para obtener la amplitud.
4. Identificación del tipo de señal

Después de capturar los datos de la señal, el siguiente paso es determinar el tipo de onda. Existen varias formas de onda posibles: senoidal, cuadrada, triangular, etc. El algoritmo debe analizar la señal capturada para identificar su forma. Esto se puede hacer de la siguiente manera:

1. Onda senoidal: El análisis debe buscar patrones suaves en los valores capturados, donde no haya cambios bruscos entre muestras consecutivas.
2. Onda cuadrada: Para identificarla, el algoritmo buscará cambios abruptos en la señal, donde los valores suban o bajen drásticamente de un extremo al otro.
3. Onda triangular: Se identificará si la diferencia entre valores consecutivos es constante o lineal.
4. Señal desconocida: Si la señal no corresponde a ninguna de las formas estándar, se clasificará como desconocida. Esto puede ocurrir si la señal tiene un comportamiento errático o no cumple con los patrones esperados de una onda senoidal, cuadrada o triangular.

Flujo de la solución:

1. El usuario presiona el primer pulsador, lo que inicia la adquisición de datos de la señal.
2. El programa comienza a capturar los valores de la señal de entrada y a almacenarlos en un arreglo de apuntadores.
3. Una vez que el usuario presiona el segundo pulsador, la adquisición se detiene, y el programa procesa los datos capturados.
4. El algoritmo analiza los datos para determinar la frecuencia y amplitud de la señal.
5. Finalmente, se analiza la forma de la onda y se clasifica como senoidal, cuadrada, triangular o desconocida.
6. Los resultados (frecuencia, amplitud y tipo de señal) se muestran en la pantalla LCD.

Con este enfoque estructurado, se garantiza que el sistema pueda identificar y procesar de manera eficiente las características principales de la señal analógica, cumpliendo con los requerimientos del proyecto.

b. Esquema donde describa las tareas que usted definió en el desarrollo de los algoritmos.

