**Desafio 1**

Samuel Gonzalez Hernandez

**Análisis y Diseño de la Solución**

**1. Análisis del Problema**

La empresa Informa2 tiene la necesidad de implementar un sistema que permita adquirir y analizar una señal analógica, de manera que se puedan identificar sus características principales como la frecuencia, amplitud y forma de onda. Para esto, se utilizará una plataforma Arduino UNO en conjunto con un generador de señales y una pantalla LCD, que permitirá visualizar los resultados obtenidos en tiempo real.

El proyecto consiste en utilizar un pulsador para iniciar y detener la adquisición de datos, mientras que el procesamiento de la señal permitirá calcular su frecuencia y amplitud, además de identificar la forma de onda (ya sea senoidal, cuadrada, triangular, etc.). La solución se debe implementar en C++, haciendo uso de punteros y arreglos para la gestión eficiente de la memoria.

**2. Consideraciones para la Solución**

Al momento de plantear la solución, es importante tener en cuenta varios aspectos:

* Simulación del sistema en Tinkercad: Utilizaremos Tinkercad como herramienta de simulación para construir virtualmente el circuito que incluye el Arduino UNO, un generador de señales y una pantalla LCD. Esto nos permitirá verificar el funcionamiento del sistema antes de la implementación física.

El pulsador será utilizado como medio para controlar la adquisición de datos, permitiendo iniciar y detener el proceso cuando sea necesario.

* Medición de la frecuencia y amplitud: La frecuencia de la señal se calculará midiendo el tiempo entre dos picos consecutivos. Para esto, se detectarán los cruces de la señal por un cierto umbral y se calculará el tiempo transcurrido entre ellos.

La amplitud se determinará midiendo los valores máximos y mínimos de la señal, calculando la diferencia entre ellos (amplitud pico a pico) y convirtiendo este valor a voltios.

* Identificación de la forma de onda: Los algoritmos deben ser capaces de comparar las características de la señal adquirida con patrones típicos de formas de onda (senoidal, cuadrada, triangular) para identificarlas correctamente. En caso de que no se corresponda con ninguna de las formas predefinidas, el sistema marcará la señal como desconocida.
* Uso de memoria dinámica: Para manejar de forma eficiente la cantidad de muestras de la señal, utilizaremos punteros y arreglos dinámicos, lo que nos permitirá optimizar el uso de la memoria disponible.
* Librerías y componentes adicionales: Se usará la librería Adafruit\_LiquidCrystal.h para gestionar de manera eficiente la pantalla LCD que mostrará las características de la señal procesada.

**3. Esquema de Tareas para el Desarrollo de los Algoritmos**

El proceso de desarrollo estará dividido en varias tareas, organizadas de la siguiente forma:

* Montaje en Tinkercad: Configurar el Arduino UNO con los componentes necesarios: generador de señales, pulsador y pantalla LCD. Esto permitirá simular el funcionamiento del sistema en un entorno controlado.
* Adquisición de la señal: Implementar el código en C++ para leer los datos de la señal analógica que llega a través del pin del Arduino.

Usar el pulsador para iniciar y detener la adquisición de datos de manera controlada.

* Medición de Frecuencia y Amplitud: Desarrollar algoritmos que midan la frecuencia de la señal analógica basándose en el tiempo entre picos consecutivos.

Calcular la amplitud de la señal determinando sus valores máximos y mínimos durante un ciclo completo.

* Identificación de la Forma de Onda: Crear algoritmos que analicen las muestras de la señal y las comparen con patrones predefinidos para identificar la forma de onda (senoidal, cuadrada, triangular, etc.).
* Visualización en la Pantalla LCD: Mostrar en tiempo real la frecuencia, amplitud y la forma de onda detectada en la pantalla LCD del Arduino.
* Pruebas y Validación: Probar cada parte del sistema de manera independiente para asegurarnos de que tanto la adquisición de datos como el procesamiento y visualización funcionen correctamente.

**4. Desafíos y Problemas Potenciales**

Durante el desarrollo es posible que surjan algunos retos. Por ejemplo, es importante calibrar correctamente el umbral de detección de picos para calcular la frecuencia de manera precisa. También hay que considerar el manejo eficiente de la memoria cuando se almacenan muchas muestras de la señal, para evitar sobrecargar el sistema.

Algunos puntos a tener en cuenta para mejorar el desarrollo son:

Realizar pruebas de cada parte del sistema por separado para detectar y solucionar errores antes de integrarlo todo.

Optimizar la adquisición de datos para que el sistema sea capaz de capturar señales de distintas frecuencias sin perder información.

Asegurar que el código esté bien estructurado y comentado, lo que facilitará el mantenimiento y posibles mejoras.

**5. Evolución de la Solución**

La solución se desarrollará de manera iterativa, comenzando por la simulación del sistema en Tinkercad para asegurarnos de que los componentes están bien conectados y funcionan como se espera. A medida que se avanza, se desarrollarán y probarán los algoritmos de procesamiento de la señal (frecuencia, amplitud y forma de onda) de manera modular.

Una vez que todo esté integrado, se realizarán ajustes en los algoritmos para mejorar la precisión en las mediciones y asegurar que el sistema funcione de manera óptima en diferentes condiciones.

**Análisis y Diseño de la Solución**

Al principio, la idea era crear una solución modular para dividir el problema en partes más pequeñas y manejables. Sin embargo, debido a las limitaciones de **Tinkercad**, no es posible implementar un diseño modular en esta plataforma. Por eso, tuvimos que modificar la solución para que funcione de forma **lineal y secuencial**.

Ahora, en lugar de dividir las funciones en módulos separados, todo el código estará en un solo bloque. A pesar de esto, vamos a seguir usando buenas prácticas de programación para que el código sea claro y fácil de seguir, incluyendo comentarios y una buena organización de las variables.

El enfoque del proyecto sigue estos pasos:

1. **Configuración del hardware**: Lo primero es configurar los pines del Arduino junto con los componentes que vamos a usar, como el pulsador, el potenciómetro, la pantalla LCD y el generador de señales.
2. **Lectura y procesamiento de datos**: Vamos a leer la entrada analógica que viene del generador de señales y convertir esos datos en valores de frecuencia y amplitud para luego mostrarlos.
3. **Interacción con el usuario**: El pulsador permitirá al usuario activar o desactivar las mediciones. Dependiendo del estado del pulsador, el sistema leerá las señales en tiempo real y las mostrará en la pantalla LCD.
4. **Mostrar datos en la pantalla**: La pantalla LCD mostrará los valores de amplitud y frecuencia en tiempo real.

**Razones para cambiar el diseño**

El cambio de una solución modular a una secuencial se debe a que **Tinkercad no permite implementar un diseño modular**. Esta plataforma es muy útil para simular circuitos, pero no ofrece la opción de organizar el código en módulos o en archivos separados como lo haríamos en otros entornos de desarrollo más avanzados.

Aunque tener un código modular facilitaría mucho el mantenimiento y la evolución del proyecto, hemos optado por una solución más sencilla que se ajusta a las limitaciones de Tinkercad.

A pesar de la falta de modularidad, nos aseguramos de que el código esté **bien organizado y comentado**, para que sea fácil de entender y modificar en el futuro.