Desafío 1

Juan Esteban Solarte López

Juan Felipe Pérez Salazar

Informática II

Universidad de Antioquia

Medellín

2.024

Desafío 1 – Informática 2

Análisis del sistema.

¿Qué se debe diseñar para la ejecución del desafío?

Este desafío nos pide diseñar un programa y/o sistema, en el cual debemos utilizar Arduino, el cual debe adquirir y analizar señales analógicas de distintas formas de onda. El programa/sistema debe medir características relevantes de la señal, como lo es la frecuencia, su amplitud e identificar su forma de onda, las cuales pueden ser senoidal, cuadrada, triangular, entre otras.

Conocimientos previos

* Señales analógicas:

En esta parte debemos conocer las diferentes formas de onda, como ya se mencionaron anteriormente, estas pueden ser senoidal, triangular, cuadrada, entre otras, sabiendo que cada una de ellas tiene su propia frecuencia y amplitud. Esto es impórtate para poder capturar las señales y extraer sus características para analizarlas.

* Arduino:

El Arduino servirá como la plataforma que permitirá capturar y adquirir las señales analógicas. Estas entradas se obtendrán a través de las entradas análogas del Arduino. Estas entradas se refieren a la capacidad del microcontrolador para leer señales que varían de forma continua dentro de un rango, normalmente de 0 a 5V. Un ejemplo de ello pueden ser los potenciómetros, que es una de las entradas que se usaran cuando se esté diseñando el Arduino.

* Visualización:  
  Todos los resultados del análisis de la señal en las cuales se encuentran la frecuencia, la amplitud y la forma de la onda, se deben mostrar en una pantalla LCD la cual estará conectada al Arduino.
* Interacción con el usuario:  
  El usuario/cliente, podrá iniciar la adquisición de la señal con un pulsador, la cual podrá ser pausada para visualizar los datos, y si se desea se podrá reanudar posteriormente para que el programa/ sistema siga con su funcionamiento.

Funcionalidades claves del programa/sistema

* Adquisición de señales:

La adquisición de datos comienza cuando el usuario presione un pulsador, el sistema capturara una señal continua mientras el pulsador no sea accionado para pausar la adquisición.

* Medición de frecuencia y amplitud:  
  Se implementará un algoritmo que mida la frecuencia de la señal en Hertz y la amplitud en voltios, siendo necesario para poder identificar el tipo de onda que se genera.
* Identificador de la forma de onda:

El sistema debe de identificar si la señal corresponde a una de las señales ya mencionadas anteriormente (senoidal, triangular, cuadrada, entre otras). En caso de que el programa no sea capaz de identificarla, se mostrara un mensaje en pantalla diciendo “onda desconocida”, indicando que esa onda no es válida para el programa.

* Visualización del sistema:

Los datos que sean procesados se visualizaran en una pantalla LCD, que contiene el Arduino, el cual será diseñado en Tinkercad, el cual es un programa que nos permite hacer una maqueta realista de como quedaría el Arduino llevado a la vida real y mostrando su correcto funcionamiento.

Condiciones que se implementaran en el desafío

* El sistema será implementado en C++, utilizando punteros, arreglos y memoria dinámica, haciendo uso de lo visto en las clases de teoría del curso.
* El montaje del sistema y su simulación se realizarán en Tinkercad.
* Se hará uso de la librería Adafruit\_LiquidCrystal.h para un correcto funcionamiento de la pantalla LCD en Tinkercad.

Implementación

En la implementación del sistema se tienen en cuenta dos aspectos, los cuales son el hardware y el software:

* Hardware:

Aunque se sabe que todo el programa se entrega de manera digital, hay que tener en cuenta que esto es un sistema que se puede llevar a la realidad e implementarlo en un Arduino físico, por eso es bueno aclarar que los objetos utilizados en Tinkercad, como lo son los cables, la placa base, los potenciómetros, las resistencias, entre otros, serán el hardware que se utilizara para ello. Lo más relevante que se utilizara para desarrollar el sistema son:

* + Arduino UNO: Es el tipo de placa se utilizará y se debe implementar para capturar la señal analógica y procesarla.
  + Generador de Señales (Tinkercad): Debe generar las señales de entrada para la simulación y garantizar el correcto funcionamiento del sistema.
  + Pulsadores: Se implementará uno para iniciar la adquisición y otro para pausar y reanudar el proceso.
  + Pantalla LCD: Esta será la que muestre los resultados obtenidos.
* Software:

Por otra parte, se tiene el software que es el más relevante, ya que será la forma de implementar el sistema, y es donde se pondrán todas las funciones necesarias para que todo funcione correctamente. Para este se empleará lo siguiente:

* Medición de Frecuencia: Se usarán las lecturas del Arduino para contar la cantidad de ciclos por segundo y calcular la frecuencia en Hertz.
* Medición de Amplitud: Se tomarán valores discretos de la señal analógica.
* Identificación de la Forma de Onda: A través de las características de las señales, como la regularidad de los picos o la inclinación de las crestas, se puede determinar si es senoidal, triangular, etc.
* Pausar y Reanudar la Adquisición: La adquisición se pausa cuando el usuario lo solicite oprimiendo el pulsador del Arduino, se muestra la información y luego continúa.
* Memoria Dinámica y Punteros: La implementación debe gestionar eficientemente los datos con memoria dinámica como se ha visto en las clases del curso, para guardar y procesar las lecturas en tiempo real.

Dificultades que se pueden tener en la implementación

Es importante resaltar que en la implementación se pueden tener complicaciones que pueden hacer que el desafío tome más tiempo en la implementación. Unas de esas complicaciones pueden ser:

* Precisión en la Medición: Lograr que el algoritmo mida la frecuencia y la amplitud con precisión podría ser una dificultad y un desafío que se debe solucionar para que el programa funcione de buena manera.
* Identificación de Formas de Onda Complejas: En algunos casos, diferenciar entre señales muy similares podría ser complicado.
* Optimización del Código: Asegurarse de que el sistema sea eficiente en el manejo de grandes volúmenes de datos. También el uso eficiente de memoria, un volcado de memoria es lo que menos se quiere que suceda, esto haría que el programa no sea lo esperado y falle
* Poca información en la entrada: puede dificultar para tomar las muestras de las señales, siendo este un punto importante para diferenciar si es una señal u otra.

Diagrama de flujo general del programa:

Diagrama

Descripción generada automáticamente con confianza media