Informe sobre el proceso de desarrollo del desafío 1

Cristian Bravo

Neider Cruz

Universidad de Antioquia

Ingeniería de telecomunicaciones

Informática 2

Análisis del problema y consideraciones para la alternativa de solución propuesta

El problema planteado en el desafío 1 en un principio fue complejo para cada una de las partes involucradas en el desarrollo del mismo, pero después de analizar de manera detenida, se planteó una solución para el mismo que consistía en recolectar, analizar y tratar los datos que el generador de función podía entregar, de manera que, al considerar un función de tipo senoidal se podía llegar a la conclusión que es de ese tipo teniendo en cuenta tres criterios, como lo son, los cruces por cero, las transiciones bruscas, y la pendiente, en este caso se tendría en cuenta si las transiciones son suaves puesto que para una señal senoidal y triangular cumpliría el requisito de tenerlas. Sin embargo, puede que estos no sean suficientes para definir el tipo de onda tratada por lo que se deben encontrar mejores y más criterios para así reducir el número de errores a la hora de analizar la señal.

Definición de tareas en el desarrollo de los algoritmos:

* Montar un circuito en Arduino que nos permita trabajar en el problema planteado, junto con ello hacer de manera correcta todas las conexiones a los pulsadores, el generador de función y a la pantalla LCD
* Crear una función que nos permita leer la frecuencia de la señal
* Crear una función que nos permita leer la amplitud de la señal
* Crear una función que nos permita almacenar todos los datos proporcionados por el generador de función
* Crear una funciona que nos permita procesar los datos, de manera que, podamos analizar patrones, transiciones, etc.
* Crear una función que mediante los datos analizados nos determine el tipo de función

Partes de la posible solución:

* Recolectar datos: en este planteamos como solución el uso de la memoria dinámica a la hora de trabajar con arreglos (en este caso el uso de un arreglo unidimensional), de esta manera podríamos guardar una gran cantidad de datos (limitada por la capacidad de almacenamiento del Arduino) y usarlos después para realizar diferentes operaciones. Se usa un ciclo que guarda los datos en el arreglo hasta que se le de la orden de finalizar; este ciclo empieza y termina cuando se activan los uno de los dos pulsadores (cada pulsador hace algo diferente).
* Analizar los datos: en este punto y después de tener ya los datos, en un arreglo unidimensional, nuestra principal tarea fue el encontrar una estrategia para usar adecuadamente dichos datos. Con los datos podríamos realizar diferentes operaciones como los son:
  + Determinar el tipo de función: en este punto se nos presentó un gran desafío de difícil solución, pero que después de varios intentos se logró tener una aproximación al resultado requerido.
    - Función cuadrada: esta la identificamos viendo sus que sus valores solo eran dos y que se alternaban y duraban lo mismo, dependiendo de la frecuencia.
    - Función triangular: en esta tuvimos que comparar sus pendientes, ya que estas deben ser iguales en magnitud, pero con diferente dirección. También usamos comparaciones para poder determinar el tipo de función.
    - Función senoidal: para verificar usamos comparaciones de sus diferentes puntos, al esta ser una función cuyo desplazamiento es repetitivo tiende a tener valores parecidos en un rango de muestra. Esta sin duda fue la más difícil de verificar, suponiendo el mayor reto a solucionar.
    - Función desconocida: esta es el resultado de que los datos no cumplan con ninguno de los requisitos para ser una de las anteriores funciones. Es decir, no tiene ninguna de las características de las otras funciones y si las tiene no son fácilmente identificables.
  + Obtener la frecuencia: en nuestro análisis usamos la siguiente estrategia: Se cuentan los cruces por cero. Se divide el número de cruces por cero entre 2 para obtener el número de ciclos. Se calcula la frecuencia dividiendo el número de ciclos por el tiempo total de muestreo en segundos. Además, se convierte el resultado a Hertz.
  + Obtener amplitud: esta se calcula por la diferencia entre los valores máximo y mínimo de la función convertida a voltios (multiplicando por 5 y dividiendo por 1023) y dividida entre dos.
* Tratar los datos: una vez obtenidos los resultados, imprimimos en el LCD (usando la librería *Adafruit\_LiquidCrystal.h*)teniendo esta la estructura:

|  |
| --- |
| Tipo: <cualquiera> |
| Frec: <k Hz>- Amp: <j V> |

# Texto Descripción generada automáticamente con confianza baja