



TECNICAS DIGITALES III

Guía N° 4: *Introducción al Procesamiento Digital de Señales con Labview*

Objetivos:

Introducir al alumno a la programación gráfica con Labview

Bibliografía recomendada:

Título: The Scientist and Engineer's Guide to Digital Signal Processing.

Autor: Steven W. Smith.

Editorial: California Technical Publishing.

(Este libro puede ser obtenido en formato electrónico del sitio web: www.dspguide.com)

Título: Tratamiento Digital de Señales.

Autores: Proakis - Manolakis.

Editorial: Pentice Hall.

Título: LabVIEW Basics I Course Manual (Disponible en autogestión)

Enunciado:

4 DSP con LABVIEW

4.1 Convolución

La convolución es una operación matemática que permite combinar dos señales f y g , obteniendo como salida la integral del producto de la señal f por una versión invertida y desplazada de la función g . En tiempo discreto, esto se expresa matemáticamente como:

$$y[n] = \sum_{h=-\infty}^{h=\infty} f[h] * g[n-h]$$

Si la señal f o g son iguales a cero para $n < 0$ y $n > N$, la ecuación se simplifica a

$$y[n] = \sum_{h=0}^{h=N-1} f[h] * g[n-h]$$

La convolución se utiliza principalmente para filtrar señales. Implementaremos esta ecuación para realizar un filtro simple, conocido como promediador móvil.

- 4.1.1 Crear un nuevo VI. Colocar dos controles, uno llamado *Medición* que representará la magnitud de un parámetro que queremos medir y otro llamado *Nivel de ruido* que se usará para simular el ruido presente en cualquier medida. Ubicarlos dentro de un bucle WHILE, temporizado cada 100ms.



TECNICAS DIGITALES III

- Colocar el nodo ubicado en **Mathematics** → **Prob & Stat** → **Probability** → **Continuous inverse CDF**. Este nodo nos permitirá generar muestras de ruido gaussiano a partir de muestras aleatorias de distribución uniforme. Seleccionar *Normal* como el tipo de CDF a calcular.
- 4.1.2 Conectar la entrada *mean* del nodo al control *Medición*, y la entrada *std* al control *Nivel de Ruido*. Colocar un nodo generador de números aleatorios, ubicado en **Programming** → **Numeric** → **Random Number**. Conectar la salida de este nodo a la entrada *cdf(x)* del nodo de CDF inversa.
- 4.1.3 Hacer click derecho sobre el lateral del bucle y seleccionar Add Shift Register. Del lado izquierdo, expandir el registro de desplazamiento hasta que tenga 10 elementos. Inicializar los 10 elementos a cero. Colocar un nodo de suma múltiple, ubicado en **Programming** → **Numeric** → **Compound Arithmetic** y usarlo para sumar las diez salidas de los registros. A la salida de la suma, dividir por 10. Esta será la salida de nuestro filtro promediador, que equivale a usar una función *f* en la convolución compuesta por 10 muestras de valor 1/10.
- 4.1.4 En el panel frontal, colocar un Waveform Chart. Usar el nodo **Programming** → **Clusters, Class & Variant** → **Bundle** para combinar la salida de nodo de CDF inversa y la salida de nuestro filtro promediador. Conectar al visualizador y correr el VI. Se debe apreciar que la señal filtrada posee un menor nivel de ruido que la original.
- 4.1.5 Repetir el filtro anterior para una cantidad *P* de muestras promediadas, programable desde un control. Sugerencia: Usar un arreglo para generar manualmente un registro de desplazamiento. En cada ciclo del WHILE este arreglo se actualizará con la muestra actual y las *N-1* anteriores, que se leen desde un registro de desplazamiento de todo el vector. Se pueden usar los nodos **Programming** → **Array** → **Initialize Array**, **Programming** → **Array** → **Array Subset**, y **Programming** → **Array** → **Build Array**.

4.2 Correlación

La correlación es una operación similar, pero sin la inversión de la señal:

$$y[n] = \sum_{h=-\infty}^{h=\infty} f[h] * g[n+h]$$

La correlación se usa para la detección de la presencia de una señal en otra. Utilizaremos esta cualidad para detectar la presencia de una señal de frecuencia determinada en la entrada de audio. Para esto correlacionaremos la entrada con un seno y un coseno, a fin de independizarnos de la fase de las senoidal.

$$det = \sqrt{\left(\frac{1}{N} \sum_{h=0}^{N-1} f_1[h] * g[n+h]\right)^2 + \left(\frac{1}{N} \sum_{h=0}^{N-1} f_2[h] * g[n+h]\right)^2}$$

donde

$$f_1 = \sin\left(2\pi f \frac{i}{N}\right)$$



TECNICAS DIGITALES III

$$f_2 = \cos\left(2\pi f \frac{i}{N}\right)$$

y g es la señal obtenida desde el micrófono.

- 4.2.1 Crear un nuevo VI tomando como base el ejercicio 3.3.7 de la Guía 3. Agregar un bucle FOR, dentro de este agregar un nodo de fórmula y generar una salida senoidal y una cosenoidal, ambas de amplitud unitaria, cuya frecuencia, igual para ambas, se pueda configurar desde el bucle WHILE. Tomar el valor de N y de i del bucle FOR. Estas señales serán nuestras señal f[h] para la correlación.
- 4.2.2 Conectar el vector de datos extraído desde el Waveform al bucle FOR. Al hacer esto el valor de N en el bucle queda configurado automáticamente a la cantidad de elementos del vector. Multiplicar la muestra indexada por la salida senoidal y por la salida cosenoidal. Las dos salidas acumularlas en sendos registros de desplazamientos, que deberán estar inicializados a cero. La acumulación hace el trabajo de la sumatoria, y se calcula para cada valor de n, dado por la señal de entrada.
- 4.2.3 Tomar como salida del bucle FOR, las salidas de los registros de desplazamiento. A estas dos señales, dividir las por N, elevarlas al cuadrado, sumarlas y sacar la raíz cuadrada, de acuerdo a la fórmula. Conectar un indicador a la salida.
- 4.2.4 Usando el VI de generación de audio de la guía 3, generar un tono de una frecuencia distinta en un 1% a la que se desea detectar y anotar el valor obtenido por el detector (que deberá estar corriendo al mismo tiempo). A continuación generar un tono con la frecuencia correcta y observar el valor del detector, que deberá ser significativamente mayor. Determinar un umbral de detección y mediante un comparador encender un led cuando se exceda el umbral. Comprobar el funcionamiento correcto del detector.

4.3 Remuestreo – Interpolación Lineal

El remuestreo o resampling consiste en generar mas o menos muestras por segundo, que las muestras de entrada disponibles. Existen diversas técnicas disponibles para realizar esto, pero en este ejercicio nos concentraremos en una, interpolación lineal, que obedece a la siguiente fórmula:

$$y_i = \left(\frac{y_1 - y_0}{x_1 - x_0}\right)(x_i - x_0) + y_0 \quad \text{con } x_i \text{ entre } x_0 \text{ y } x_1$$

- 4.3.1 Crear un nuevo VI. Generar una señal senoidal de N_1 muestras y aplicar la ecuación anterior para remuestrearla a N_2 muestras. La salida de la fórmula para N_1 nos entregará las muestras y_0, y_1, y_{N_1-1} , mientras que de i/N_1 obtendremos los valores x_0, x_1, x_{N_1-1} necesarios para calcular y_i para cada x_i generado desde j/N_2 .
- 4.3.2 Graficar ambas senoidades en un X/Y Graph, al cual hay que pasar un arreglo de clusters, en cada cluster el vector x y el vector y de cada señal.