

	<b>UNIVERSIDAD TECNOLÓGICA NACIONAL, FRC</b>	2009
	<b>TECNICAS DIGITALES III</b>	5R

## **Trabajo práctico N° 2: *Introducción al Procesamiento Digital de Señales.***

### **Introducción:**

Para el desarrollo del presente TP el alumno deberá tener conocimientos teóricos sobre los siguientes temas:

- ✓ Funciones elementales en tiempo discreto: impulso unitario, escalón unidad, rampa unidad, exponencial.
- ✓ Teoría del muestreo, aliasing.
- ✓ Muestreo multifrecuencia.
- ✓ Manipulaciones simples de señales en tiempo discreto: desplazamiento temporal, reflexión, escalado temporal o submuestreo, escalado de amplitud, suma y producto.
- ✓ Convolución.
- ✓ Correlación cruzada y autocorrelación.
- ✓ Transformada de Fourier, Transformada Discreta de Fourier (DFT) y Transformada Rápida de Fourier (FFT).
- ✓ Programación básica en lenguaje SCRIPT de Matlab.
- ✓ Programación en lenguaje C u otro lenguaje de alto nivel.

### **Objetivos:**

Finalizado el presente TP el alumno habrá adquirido práctica en el uso de herramientas matemáticas y algoritmos básicos en lenguaje de programación para el procesamiento digital de señales. Además fijará conocimientos teóricos a través de los ejercicios prácticos.

### **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](http://www.dspguide.com) )

Título: **Tratamiento Digital de Señales.**

Autores: Proakis - Manolakis.

Editorial: Pentice Hall.

Título: **Tratamiento de Señales en Tiempo Discreto.**

Autores: Oppenheim – Schafer – Buck.

Editorial: Pentice Hall.

Título: **The Student Edition of MATLAB.**

Autores: Hanselman – Littlefield.

Editorial: Prentice Hall.

	<b>UNIVERSIDAD TECNOLÓGICA NACIONAL, FRC</b>	2009
	<b>TECNICAS DIGITALES III</b>	5R

### Enunciado:

#### 2.1 Señales básicas utilizadas en el procesamiento digital de señales

Las señales en tiempo discreto se representan matemáticamente como una secuencia de números. En la práctica esta secuencia surge del muestreo de una señal analógica.

Las señales básicas utilizadas en el procesamiento digital de señales son: el impulso  $\delta[n]$ , el escalón unidad  $u[n]$ , las señales exponenciales del tipo  $a^n u[n]$ , las senoidales y su generalización a exponenciales complejas.

Objetivo: Utilizar Matlab para generar las señales básicas utilizadas en el Procesamiento Digital de Señales.

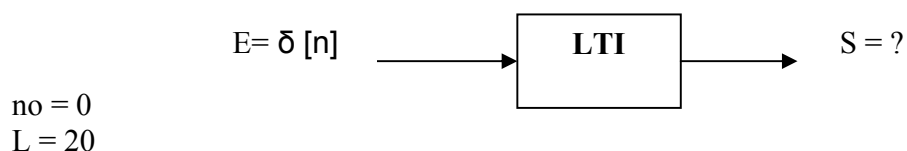
##### 2.1.1 Señal impulso unitario (desplazado)

esta señal se define como la secuencia

$$\delta[n - n_0] = 1 \text{ para } n = n_0, \text{ y}$$

$$\delta[n - n_0] = 0 \text{ para } n \neq n_0$$

Genere en lenguaje script de Matlab y represente gráficamente una señal impulso la cual se usará para excitar un sistema lineal invariante en el tiempo causal de modo que se obtengan L muestras a la salida de dicho sistema.



##### 2.1.2 Tren de impulsos

Un tren de impulsos de periodo P y longitud  $L = M \times P$  está dado por:

$$S[n] = \sum_{i=0}^{M-1} \delta(n - ip) \quad \text{donde } n = 0, 1, 2, \dots, L-1$$

Genere y represente gráficamente mediante Matlab un tren de impulsos de periodo 5 y longitud 100.

##### 2.1.3 Escalón unitario

el escalón unitario tiene como expresión

$$u[n] = 1 \text{ para } n \geq 0, \text{ y}$$

	UNIVERSIDAD TECNOLÓGICA NACIONAL, FRC	2009
	TECNICAS DIGITALES III	5R

$u[n] = 0$  para  $n < 0$ .

Genere y represente gráficamente la secuencia escalón unitario de longitud  $L=20$ .

#### 2.1.4 Señal sinusoidal

Una señal sinusoidal de amplitud  $A$ , frecuencia  $\omega_0$  y fase  $\varphi$  en tiempo continuo esta dada por:

$$X(t) = A \sin(\omega_0 t + \varphi)$$

En tiempo discreto  $t=nT_s$ , donde  $T_s$  es el periodo de muestreo, entonces:

$$X[n] = A \sin(\omega_0 nT_s + \varphi) = A \sin(2\pi f_0/f_s n + \varphi); \quad \omega_0 = 2\pi f_0$$

Donde  $f_0/f_s$  se la denomina frecuencia normalizada y  $f_s$  es la frecuencia de muestreo.

Genere y represente gráficamente mediante la función **stem** una señal cosenoidal en tiempo discreto de  $N$  muestras, cuyos parámetros son:

$$\begin{aligned} A &= 10, \\ f_0 &= 10 \text{ Hz}, \\ f_s &= 1000 \text{ Hz}, \\ \varphi &= 0^\circ \end{aligned}$$

$$N = 100.$$

#### 2.1.5 Señales exponenciales

Una señal exponencial en tiempo discreto es de la forma

$$X[n] = a^n, \quad \text{siendo } a \text{ un número real y } n=0, 1, 2, 3, 4\ldots$$

Si  $0 < a < 1$  o  $-1 < a < 0$ , entonces la exponencial es decreciente

Si  $a > 1$  o  $a < -1$ , entonces la exponencial es creciente

Genere y represente gráficamente las señales exponenciales decrecientes y crecientes de longitud  $L=20$ .

	<b>UNIVERSIDAD TECNOLÓGICA NACIONAL, FRC</b>	2009
	<b>TECNICAS DIGITALES III</b>	5R

### 2.1.6 Señales complejas

Si bien en el mundo real las señales tienen valores reales, sin embargo es muy frecuente el uso de señales con partes real e imaginaria, es decir señales complejas, por ejemplo en sistemas de modulación.

Para el caso en que el valor **a** de la señal exponencial sea un valor complejo, tendremos:

$$\mathbf{a} = \mathbf{r}e^{j\Theta}$$

por lo tanto

$$\mathbf{X}[n] = \mathbf{r}^n e^{j\Theta n} = \mathbf{r}^n (\cos \Theta n + j \sin \Theta n)$$

Gráficamente se podrá ahora representar una parte real:

$$\mathbf{X}_R[n] = \mathbf{r}^n \cos \Theta n$$

y una parte imaginaria:

$$\mathbf{X}_I[n] = \mathbf{r}^n \sin \Theta n$$

Genere una señal exponencial compleja donde  $r = 0.9$  y  $\Theta = \pi/10$ .  
¿ Que sucede si  $r = 1$ ?

## 2.2 Muestreo de señales en el dominio del tiempo - Aliasing

Objetivo: Utilizar Matlab para simular el muestreo de señales y analizar el efecto causado por el submuestreo denominado ALIASING.

Para este ejercicio práctico deberá generar y simular el muestreo de la siguiente señal

$$x(t) = A \sin(2 \pi f_o t + \Phi)$$

La señal muestreada será

$$x[n] = x(t) \text{ para } t = nT_s = n/f_s \quad ; \text{ donde } f_s \text{ es la frecuencia de muestreo}$$

$$x[n] = A \sin(2\pi f_o/f_s n + \Phi)$$

	UNIVERSIDAD TECNOLÓGICA NACIONAL, FRC	2009
	TECNICAS DIGITALES III	5R

### 2.2.1 Represente la señal senoidal tomando

$f_0 = 100 \text{ Hz}$  ; frecuencia de la senoide  
 $T_s = 1 \text{ mS}$  ; periodo de muestreo  
 $A = 1$  ; amplitud

La fase puede ser cualquier valor entre 0 y  $2\pi$ .

Utilice la función **stem** para visualizar la función muestreada  $x[n]$ .

Luego utilice la función **plot** para visualizar la función de modo que las muestras sean unidas (interpoladas) mediante líneas rectas.

Pruebe con  $f_s$  próxima a  $2f_0$  y con  $f_s \gg f_0$ , por ejemplo  $f_s = 10 \text{ KHz}$ .

### 2.2.2 Varíe $f_0$ entre 100 Hz y 475 Hz en saltos de 125Hz, para $f_s = 8 \text{ KHz}$ .

Utilice la función **subplot** para representar los cuatro casos en una misma ventana.

¿Que sucede con la frecuencia de la señal senoidal?

### 2.2.3 Varíe $f_0$ entre 7525 y 7900 en saltos de 125 Hz.

¿Que sucede ahora con la frecuencia de la señal senoidal representada?, ¿Porque?

### 2.2.4 Nuevamente varíe $f_0$ entre 32100 Hz y 32475 Hz en saltos de 125 Hz.

¿Que sucede ahora con la frecuencia de la señal senoidal representada?, ¿Porque?

**Describa el efecto que se observa en los gráficos anteriores.**

Nota: para poder analizar los gráficos, estos deberán tener la misma escala.

## 2.3 Muestreo multifrecuencia: diezmado e interpolación.

En muchas aplicaciones prácticas de procesamiento digital de señales es necesario cambiar la frecuencia de muestreo, incrementándola o decrementándola. El proceso de convertir una señal discretizada a una frecuencia de muestreo en otra señal discretizada a otra frecuencia de muestreo se denomina **conversión de frecuencia de muestreo**. Los sistemas que emplean múltiples frecuencias de muestreo en el procesamiento de señales digitales se denominan **sistemas de procesamiento digital de frecuencia múltiple**.

### 2.3.1 Analice las siguientes funciones de Matlab: **decimate**, **interp** y **resample**.

### 2.3.2 Genere una señal senoidal de frecuencia $f_0 = 100 \text{ Hz}$ , muestreada con una $F_s = 10 \text{ KHz}$ .

3.3.2.1 Decrementa el número de muestras de la señal senoidal con un factor de 4.

3.3.2.2 Incrementa el número de muestras de la señal senoidal con un factor de 2.

3.3.2.3 Modifique la frecuencia de muestreo en un factor de  $2/3$ .

En todos los casos utilice la función **stem** para graficar los resultados.

	UNIVERSIDAD TECNOLÓGICA NACIONAL, FRC	2009
	TECNICAS DIGITALES III	5R

## 2.4 Manipulaciones simples de señales en tiempo discreto

Objetivo: comprender los algoritmos de manipulación o modificaciones simples de las señales en las que intervienen la variable independiente o la amplitud.

### 2.4.1 Desplazamiento en el tiempo.

Introducción:

Sea una señal discreta  $x(n)$ , esta puede ser desplazada en el tiempo reemplazando la variable independiente  $n$  por  $n-k$  donde  $k$  es un entero.

Si  $k$  es un entero **positivo** el desplazamiento provoca un **retraso** de la señal en  $k$  unidades de tiempo.

Si  $k$  es un entero **negativo** el desplazamiento temporal resulta en un **adelanto** de la señal en  $|k|$  unidades de tiempo.

Note que en el muestreo en tiempo real es imposible obtener un adelanto, es decir, no podemos contar con muestras futuras de la señal.

2.4.1.1 Dado un sistema al cual se le aplica una señal  $x(n)$  y cuya salida es  $y(n) = x(n-2)$ , dicho sistema, ¿adelanta o retarda?

2.4.1.2 Dadas las siguientes muestras de una señal,

$$x[n] = [0 \ 3 \ 2 \ 1 \ 0 \ 1 \ 2 \ 3 \ 0] \quad ; \text{ la flecha indica el origen de tiempo, es decir, } n = 0$$

Grafique en una misma figura  $x(n)$  e  $y(n) = x(n-k)$  siendo  $k=2$ .

### 2.4.2 Reflexión.

Introducción:

Esta operación consiste en cambiar la variable independiente  $n$  por  $-n$ , el resultado de esto es la reflexión de la señal con respecto al origen de tiempos  $n = 0$ .

2.4.2.1 Dada la siguiente señal,

$$x[n] = [-5 \ -4 \ -3 \ -2 \ -1 \ 0 \ 1 \ 2 \ 3 \ 4 \ 5]$$

Obtenga la reflexión  $y(-n)$  de  $x(n)$ . Grafique ambas señales en una misma figura.

2.4.2.2 Dada una señal  $x(n)$ , ¿Qué operaciones debe realizar para obtener  $y(n) = x(-n+2)$ ? Compruebe su deducción en forma gráfica.

	UNIVERSIDAD TECNOLÓGICA NACIONAL, FRC	2009
	TECNICAS DIGITALES III	5R

### 2.4.3 Escalado temporal

Introducción:

Esta operación consiste en reemplazar la variable independiente  $n$  por  $an$ , siendo  $a$  un entero. Esta operación también suele llamarse *submuestreo*.

Dada una señal senoidal  $x(n)$  de 50 Hz muestreada con una frecuencia de muestreo de 1000 muestras/seg., grafique en una misma figura las señales  $x(n)$  y  $y(n) = x(an)$  siendo  $a = 2$ .

### 2.4.4 Suma, multiplicación y escalado de amplitud.

Introducción:

Estas modificaciones son modificaciones de amplitud.

El *escalado de amplitud* consiste en multiplicar una constante  $A$  por cada muestra de la señal.

La *suma* de dos señales  $x_1(n)$  y  $x_2(n)$  es una señal  $y(n)$  cuyo valor en cualquier instante es igual a la suma de los valores en ese instante de las señales  $x_1(n)$  y  $x_2(n)$ .

El *producto* de dos señales  $x_1(n)$  y  $x_2(n)$  es una señal  $y(n)$  cuyo valor en cualquier instante es igual al producto de los valores en ese instante de las señales  $x_1(n)$  y  $x_2(n)$ .

2.4.4.1 Dada la siguiente señal,

$$x[n] = [1 \ 2 \ 3 \ 4 \ 5 \ 4 \ 3 \ 2 \ 1]$$

Obtenga la señal  $y(n) = A x(n)$ , siendo  $A = 2$ . Grafique  $x(n)$  e  $y(n)$  en una misma figura.

2.4.4.2 Dadas las siguientes señales,

$$x_1[n] = [0 \ 1 \ 1 \ 1 \ 1 \ 0]$$

$$x_2[n] = [0 \ 1 \ 2 \ 3 \ 4 \ 5]$$

Obtenga la señal  $y(n) = x_1(n) + x_2(n)$ . Grafique  $x_1(n)$ ,  $x_2(n)$  e  $y(n)$  en una misma figura.

2.4.4.3 Dadas las siguientes señales,

$$x_1[n] = [1 \ 2 \ 3 \ 4 \ 5 \ 6]$$

$$x_2[n] = [6 \ 5 \ 4 \ 3 \ 2 \ 1]$$

Obtenga la señal  $y(n) = x_1(n) * x_2(n)$ . Grafique  $x_1(n)$ ,  $x_2(n)$  e  $y(n)$  en una misma figura.

	UNIVERSIDAD TECNOLÓGICA NACIONAL, FRC	2009
	TECNICAS DIGITALES III	5R

## 2.5 Convolución

Objetivo: comprender el algoritmo de convolución y demostrar su aplicación en el dominio del tiempo para filtrar señales.

Introducción: la **convolución** es una forma matemática de combinar dos señales para obtener una tercera señal. Esta es una de las técnicas simples más importantes del Procesamiento Digital de Señales. Esta importancia radica en que en sistemas lineales relaciona tres señales: la señal de entrada, la señal de salida y la respuesta al impulso del sistema.

La convolución también puede ser utilizada para el filtrado de señales.

Matemáticamente la convolución de señales en tiempo discreto esta dada por:

$$y[n] = \sum_{k=0}^{M-1} x[k]h[n-k] \quad n = 0, 1, 2, \dots$$

en forma simbólica es:  $y(n) = x(n) * h(n)$

El proceso de separación de dos señales convolucionadas se denomina **deconvolución**. Si contamos con la respuesta impulsional del sistema  $h(n)$  y con la salida del sistema  $y(n)$  la deconvolución permite encontrar las muestras de la señal de entrada  $x(n)$ . Por otra parte, teniendo como datos la entrada al sistema  $x(n)$  y su salida  $y(n)$  podemos determinar, mediante la deconvolución la respuesta impulsional del sistema.

**2.5.1** Realice un diagrama de flujo del algoritmo de convolución.

**2.5.2** Calcule y grafique manualmente la convolución entre la señal  $x(n)$  y respuesta al impulso de un sistema  $h(n)$ :

$$x[n] = \{4, 3, 2, 1, 0\}$$

$$h[n] = \{0, 0, 1, 1, 1\}$$

**2.5.3** Codifique el algoritmo realizado en el punto 2.5.1 mediante Lenguaje C o el lenguaje script de Matlab. Verifique los resultados del punto 2.5.2.

**2.5.4** Verifique los resultados del punto 2.5.2 mediante la función **conv** de Matlab.

**2.5.5** Demuestre las siguientes propiedades de la convolución:

**Ley Conmutativa:**  $x(n) * h(n) = h(n) * x(n)$



	UNIVERSIDAD TECNOLÓGICA NACIONAL, FRC	2009
	TECNICAS DIGITALES III	5R

**Ley Asociativa:**  $[x(n) * h_1(n)] * h_2(n) = x(n) * [h_1(n) * h_2(n)]$

**Ley Distributiva:**  $x(n) * [h_1(n) + h_2(n)] = x(n) * h_1(n) + x(n) * h_2(n)$

Siendo:

$$x[n] = [1 \ 2 \ 3 \ 4 \ 3 \ 2 \ 1]$$

$$h[n] = h_1[n] = [1 \ 1 \ 1]$$

$$h_2[n] = [1 \ 2 \ 1]$$

**2.5.6** Si un sistema tiene una respuesta al impulso  $h(n)$  y salida  $y(n)$ , determine mediante la deconvolución, función **deconv** de Matlab, la entrada  $x(n)$ , siendo:

$$h[n] = [1 \ 2 \ 3]$$

$$y[n] = [4 \ 13 \ 28 \ 27 \ 18]$$

Antes de comenzar analice la función **deconv**.

**2.5.7** Genere 10 ciclos de una señal senoidal de frecuencia  $f_0=100$  Hz de amplitud 10, muestreada a una frecuencia  $f_s = 10000$ Hz. Luego genere ruido de amplitud  $\pm 4$ , mediante la función **rand** y súmelo a la señal senoidal.

Ahora convolucione la señal senoidal con ruido con la siguiente respuesta al impulso  **$h(n)$** :

$$h[n] = [0.0040 \ 0.0045 \ 0.0060 \ 0.0083 \ 0.0115 \ 0.0155 \ 0.0202 \ 0.0253 \ 0.0306 \ 0.0361 \ 0.0414 \ 0.0463 \ 0.0507 \ 0.0543 \ 0.0570 \ 0.0586 \ 0.0592 \ 0.0586 \ 0.0570 \ 0.0543 \ 0.0507 \ 0.0463 \ 0.0414 \ 0.0361 \ 0.0306 \ 0.0253 \ 0.0202 \ 0.0155 \ 0.0115 \ 0.0083 \ 0.0060 \ 0.0045 \ 0.0040];$$

Utilice la función **conv** de MATLAB.

Grafique la señal senoidal con ruido y la señal obtenida como resultado de la convolución en una misma figura con las mismas escalas de amplitud en ambos gráficos para poder comparar.

¿Que sucede con el ruido en la señal obtenida?

Obtenga una conclusión con la ayuda de la función **freqz** de Matlab para graficar la respuesta en frecuencia del sistema  $h[n]$ .

¿Cual es la longitud de la señal obtenida?

	<b>UNIVERSIDAD TECNOLÓGICA NACIONAL, FRC</b>	2009
	<b>TECNICAS DIGITALES III</b>	5R

## 2.6 Correlación

Objetivo: comprender el algoritmo de correlación de señales.

Introducción: la correlación es una operación matemática similar a la convolución. La correlación utiliza dos señales de entrada para producir una tercera señal denominada correlación cruzada. Si una señal es correlacionada consigo misma el resultado es una señal denominada autocorrelación.

Una aplicación típica es la detección de una forma de onda conocida en una señal con ruido.

Se define a la correlación cruzada de dos secuencias  $x(n)$  e  $y(n)$  como:

$$r_{xy}[n] = \sum_{k=i}^{N-|L|-1} x(k)y(k-n)$$

Se define a la autocorrelación como:

$$r_{xx}[n] = \sum_{k=i}^{N-|L|-1} x(k)x(k-n)$$

donde

$$\begin{aligned} i &= 1, L = 0 \text{ para } n \geq 0 \\ i &= 0, L = 1 \text{ para } n < 0 \end{aligned}$$

**2.6.1** Realice un diagrama de flujo del algoritmo de correlación, luego codifique dicho algoritmo mediante lenguaje C o el lenguaje script de Matlab.

**2.6.2** Determine en forma manual, mediante gráficos la correlación cruzada de las siguientes señales:

$$\begin{aligned} X(n) &= \{0, 1, -1, 0, 0, 1, 2, 1, 0, 0, 0, 1, 0, -1, 0\} \\ &\quad \uparrow \\ Y(n) &= \{1, 2, 1\} \\ &\quad \uparrow \end{aligned}$$

Luego aplique el algoritmo codificado en el punto 2.6.1 para verificar los resultados.

Los elementos indicados con  $\uparrow$  pertenecen al elemento [0].

	UNIVERSIDAD TECNOLÓGICA NACIONAL, FRC	2009
	TECNICAS DIGITALES III	5R

## 2.7 Muestreo en el dominio de la frecuencia – Transformada de Fourier Discreta

Introducción:

el análisis frecuencial de señales discretas se realiza, normalmente, mediante procesadores de señales digitales. Para realizar un análisis frecuencial de una señal en tiempo discreto es necesario convertir la secuencia  $x[n]$  en el dominio del tiempo, a  $X(w)$  en el dominio de la frecuencia mediante la transformada de Fourier; sin embargo  $X(w)$  es una función continua de la frecuencia, y por lo tanto, no es conveniente para el uso mediante un procesador digital. En lugar de la transformada de Fourier se utiliza la Transformada Discreta de Fourier (DFT) la cual nos devuelve muestras del espectro  $X(w)$ .

**2.7.1** Analice la función *fft* de Matlab utilizada en el análisis frecuencial de señales en tiempo discreto.

**2.7.2** Obtenga, mediante la función *fft* el espectro de frecuencias de las señales senoidal (*sin*), triangular (*tripuls*) y cuadrada (*square*). Tomar como frecuencia de la señal de entrada  $f_0=20\text{Hz}$  y frecuencia de muestreo  $f_s=100\text{Hz}$ . Grafique mediante la función *stem* la parte real (función *real()*) e imaginaria (función *imag()*) del vector resultante. Utilice la función *subplot* para que ambos gráficos se realicen en la misma figura. Grafique a continuación la magnitud (función *abs*) y la fase (función *angle*) de la función en el dominio de la frecuencia. Expresé el eje de las abscisas en frecuencia en Hz, tenga en cuenta que para  $n=N$  tendremos la frecuencia de muestreo  $f_s$ .

**2.7.3** Varíe la frecuencia de la señal de entrada de modo que  $f_0$  sea 20Hz, 40Hz y 80Hz, manteniendo  $f_s=100\text{Hz}$ . Obtenga las transformadas de FOURIER para distintas frecuencias, a partir de que frecuencia comienza a aparecer el solapamiento?

**2.7.4** Desarrolle un instrumento virtual que permita realizar un análisis espectral de la señal inyectada. Mediante el generador desarrollado en el TP1 inyecte una señal senoidal y verifique que sucede al modificar el número de muestras de modo que el número de ciclos de la señal generada sea entero, luego pruebe generando una cantidad de muestras de modo que el número de ciclos de la señal generada no sea un entero. ¿Cómo se denomina el efecto causado?

**2.7.5** Modifique el instrumento virtual creado en el punto 2.7.4 para demostrar que el efecto de fuga o drenaje espectral (spectral leakage) es suavizado agregando distintas ventanas (Hanning, Hamming, Triangular, etc) antes de pasar la señal del dominio del tiempo al dominio de la frecuencia.

**2.7.6** Mediante el instrumento virtual creado en el punto 2.7.4 demuestre el efecto ALIASING en el dominio de la frecuencia. Primero genere una señal senoidal de 10Hz muestreada a 100 muestras por segundo, luego ajuste la frecuencia de la señal a 90 Hz manteniendo la frecuencia de muestreo.

**2.7.7** Realice un instrumento virtual que permita adquirir señales mediante la placa de sonido de la PC y muestre en pantalla la señal adquirida y el espectro en frecuencia de dicha señal.