

# PRÁCTICA 2: Procesado digital de señales analógicas

## Preparación del entorno:

1. Copiar el directorio PDS\_practica1 al directorio Alumnos. Establecer el directorio Alumnos>

PDS\_practica1 como directorio de trabajo de Matlab. En este directorio se deben encontrar los ficheros:

parametros11.m	parametros12.m	parametros13.m
practica11.m	practica12.m	practica13.m

- El fichero parametros 1x.m contiene los parámetros de la simulación x.
- El fichero practica1x.m realiza la simulación x y representa los resultados gráficamente.

2. Añadir al path de Matlab el directorio PDS\_practica1\_LIB donde se encuentra una librería con todas las rutinas necesarias para la ejecución de la práctica. File>Set Path>Add Folder y después Close.

## 1.1. Conversión C/D ideal

Considerar el sistema de la Fig. 1 en el que la señal continua  $x_c(t)$  es generada y convertida en la señal discreta  $x[n]$  mediante un convertidor C/D ideal.

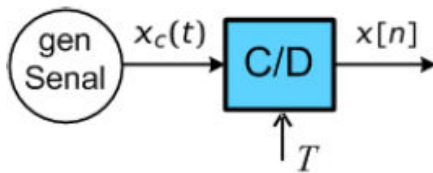


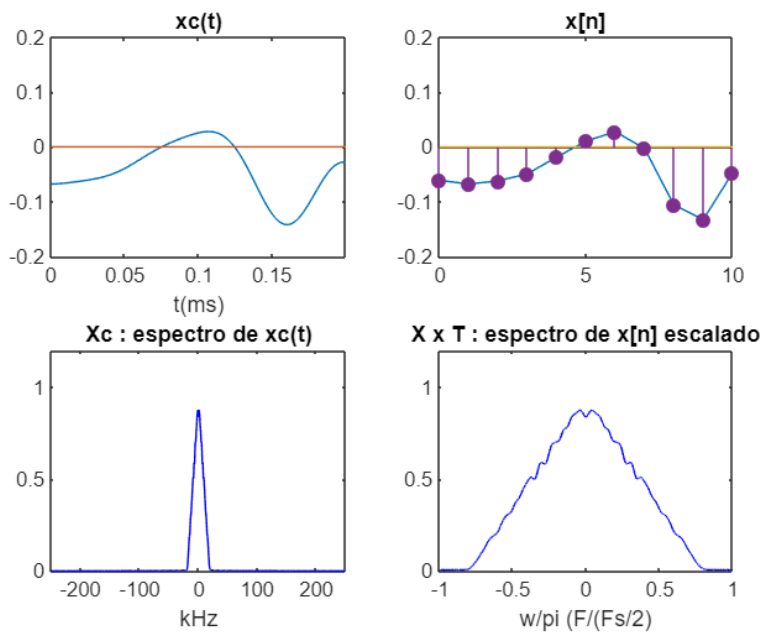
Figura 1. Sistema\_11.

Los parámetros para la simulación del sistema11 se encuentran en el fichero parametros11.m que se puede ver en el listado 1

Realizar la simulación del sistema11 ejecutando el comando practica11. Como resultado debe aparecer una gráfica con un tramo de la señales  $x_c(t)$  y  $x[n]$  así como sus espectros.

**Cuestión 1.** Con los parámetros elegidos ¿se cumple el criterio de Nyquist? ¿cuál es la frecuencia mínima a partir de la cual se puede muestrear la señal  $x_c(t)$  sin aliasing?.

```
practica11
```



**Listado 1.** *parametros11.m*

```
%-----
% Parametros 11
%-----

% Parametros del Sistema
%-----
Fp = 20000; % Ancho de banda de la señal xc(t) (Hz)
Fs = 48000; % Frecuencia de muestreo (Hz)
T=1/Fs; % intervalo de muestreo

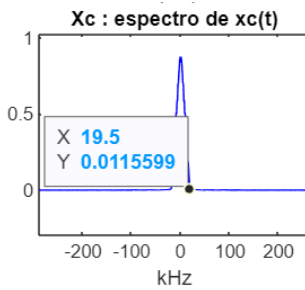
%-----
% Parametros de la prueba
%-----
longT=0.1; % Duración de la simulación (segundos)
formaEspectro='triangular';
%formaEspectro='plano';

%-----
% Parametros de presentación
%-----
% Ventana de presentación señales en el tiempo
tinicio= longT/2; % comienzo ventana (segundos)
tventana=0.0002; % duración ventana (segundos)
```

Sabemos que el Teorema de Nyquist (o Teorema del muestreo) es:

$$\Omega_s > 2\Omega_n$$

Como la frecuencia de muestreo  $\Omega_s$  es 48000 Hz y la frecuencia máxima  $\Omega_n$  medida en la gráfica es 20000Hz por lo que  $48000 > 40000$  y podemos decir que se cumple el Teorema del muestreo y no hay aliasing.



La frecuencia mínima está relacionada con el Teorema del Muestreo (frecuencia a partir de la cual el espectro desaparece). La frecuencia mínima es  $2 \cdot 20\text{kHz} = 40000\text{Hz}$ .

**Cuestión 2.** Teniendo en cuenta los parámetros elegidos, ¿a qué pulsación discreta ( $w_n$ ) corresponde la máxima frecuencia de la señal  $x_c(t)$  desde el punto de vista teórico? Confirmar dicho valor observando las gráficas.

$$\Omega = 2\pi f = 2\pi \cdot 20000 = 40000\pi$$

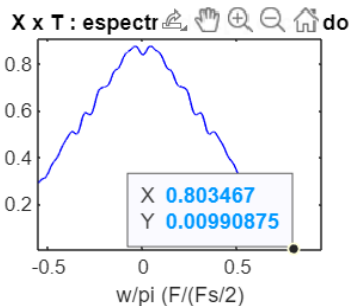
$$w_n = \Omega \cdot T = 40000\pi \cdot \frac{1}{48000} = \frac{5}{6}\pi$$

Normalizando a  $\pi$  (dividiendo  $\frac{w_n}{\pi}$ ) tenemos:

$$\frac{w_n}{\pi} = \frac{5}{6} = 0.8333$$

A continuación medimos el valor en la gráfica:

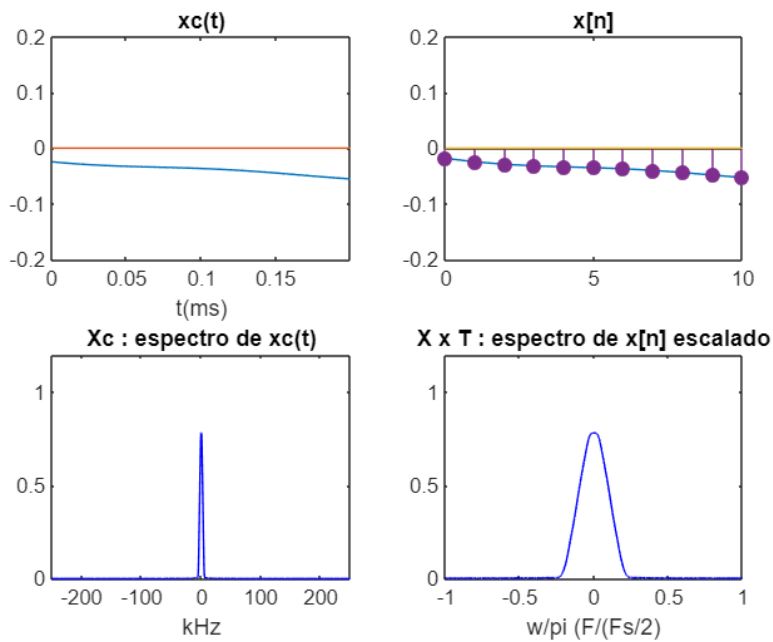
$$\frac{w_n}{\pi} = 0.80346$$



**Cuestión 3.** Repetir la simulación para una señal  $x_c(t)$  con una ancho de banda  $F_p=5\text{kHz}$ . Responder a la cuestión 2 en esta nueva situación.

Creamos un nuevo script que se va a llamar "parametros11\_p3" donde vamos a cambiar  $F_p=5\text{kHz}$  y creamos otro script que llamaremos "practica11\_p3" y lo ejecutamos:

```
practica11_p3
```



Repetimos los cálculos que hemos hecho en la pregunta 2:

$$\Omega = 2\pi f = 2\pi \cdot 5000 = 10000\pi$$

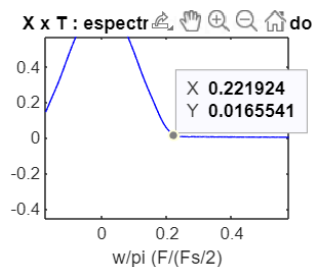
$$w_n = \Omega \cdot T = 10000\pi \cdot \frac{1}{48000} = \frac{5}{24}\pi$$

Normalizando a  $\pi$  (dividiendo  $\frac{w_n}{\pi}$ ) tenemos:

$$\frac{w_n}{\pi} = \frac{5}{24} = 0.208333$$

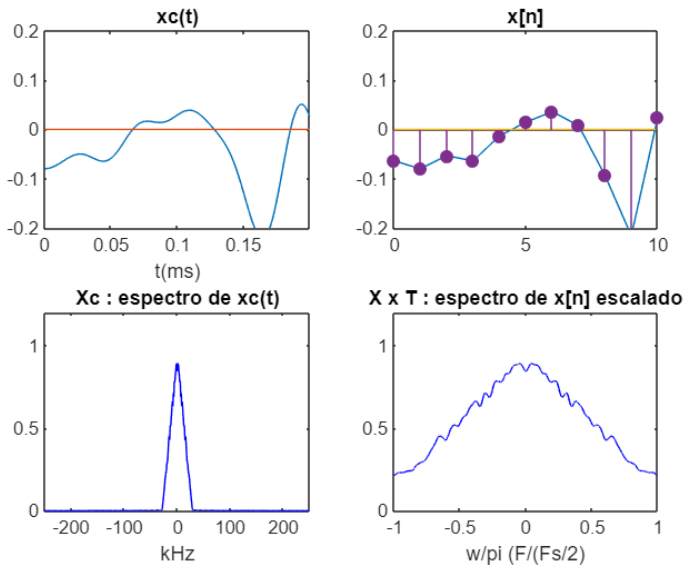
A continuación medimos el valor en la gráfica:

$$\frac{w_n}{\pi} = 0.2219$$



**Cuestión 4.** Repetir la simulación para una señal  $x_c(t)$  con una ancho de banda  $F_p=30$  kHz. Interpretar el resultado que se obtiene.

Hacemos lo mismo que en el apartado anterior:



La frecuencia natural ( $\Omega_n$ ) es mayor a la frecuencia máxima de muestreo ( $\Omega_s$ ), por lo tanto hay aliasing cosa que se puede comprobar en el espectro de la señal discreta ya que ésta no llega a  $x[n]=0$ , es decir, se han superpuesto dos señales (aparece como la suma de ambas señales).

## 1.2. Conversión C/D ideal y conversión D/A real

Considerar el sistema de la Fig. 2 en el que la señal continua  $x_c(t)$  es generada y convertida en la señal discreta  $x[n]$  mediante un convertidor C/D ideal y posteriormente convertida al terreno continuo mediante un convertidor D/A real. Observar que aparecen dos señales continuas nuevas en el esquema:  $x_{imp}(t)$ , que es la salida del convertidor de muestra a impulso y  $x_{ret}(t)$  que es la salida del filtro reconstructor de orden cero.

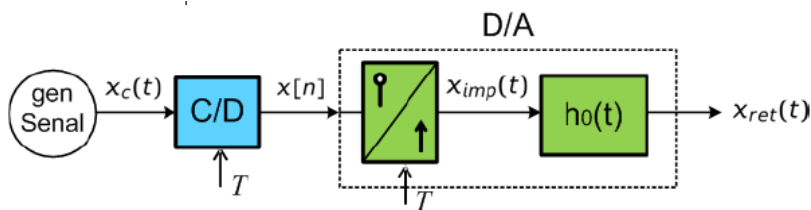


Figura 2. Sistema12.

Los parámetros para la simulación del sistema12 se encuentran en el fichero parametros12.m que se puede ver en el listado 2. Observar que en este caso se ha optado por una señal con espectro plano.

## Listado 2. *parametros12.m*

```
%-----
% Parametros 12
%-----

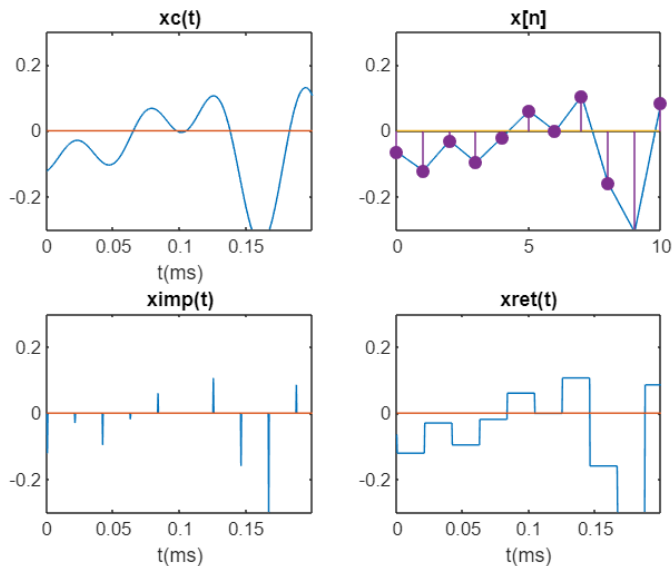
%-----
% Parametros del Sistema
%-----
Fp = 20000; % Ancho de banda de la señales (Hz)
Fs = 48000; % Frecuencia de muestreo (Hz)
T=1/Fs; % intervalo de muestreo

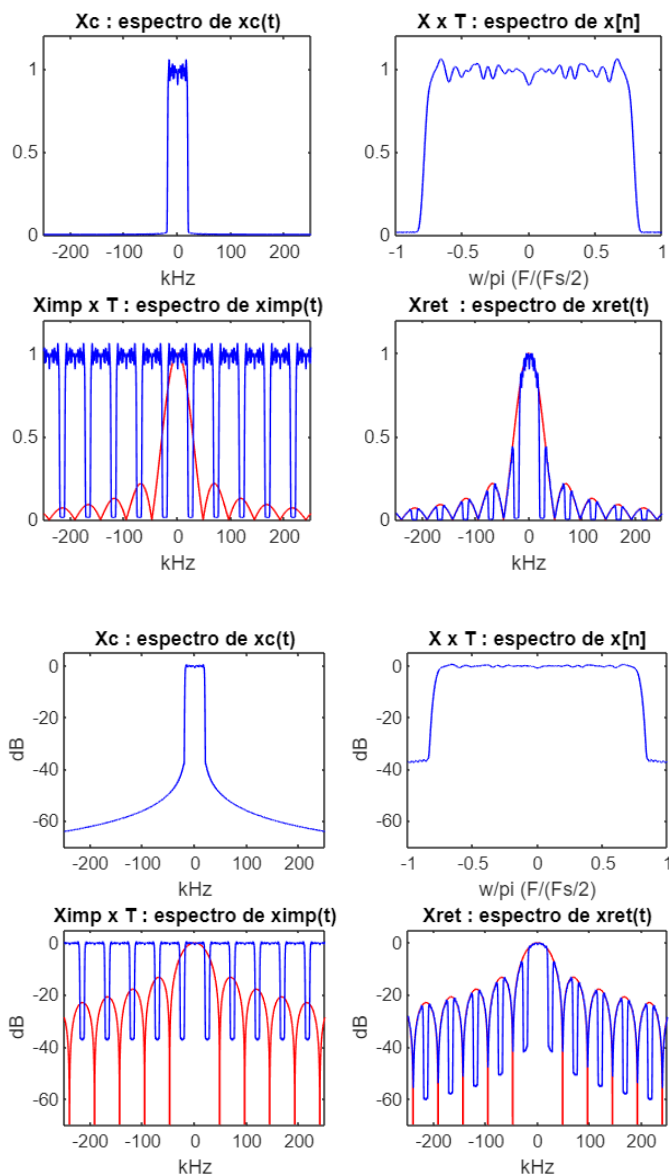
%-----
% Parametros de la prueba
%-----
longT=0.1; % Duración de la simulación (segundos)
%formaEspectro='triangular';
formaEspectro='plano';

%-----
% Parametros de presentación
%-----
% Ventana de presentación señales en el tiempo
inicio= longT/2; % comienzo ventana (segundos)
tventana=0.0002; % duración ventana (segundos)
```

Realizar la simulación del sistema12 ejecutando el comando practica12. Como resultado deben aparecer tres gráficas. En la primera se representa un tramo temporal de las 4 señales involucradas. En la segunda se muestran los espectros de las 4 señales en escala lineal y en la tercera los espectros en escala logarítmica. En la representación gráfica del espectro de  $x_{imp}(t)$  y de  $x_{ret}(t)$  se ha superpuesto en color rojo la respuesta en frecuencia del filtro de retención de orden cero.

practica12





**Cuestión 5.** Observando el espectro de  $x_{ret}(t)$ , ¿a qué frecuencia comienza la (primera) banda imagen? ¿en qué frecuencia está centrada la primera imagen?. ¿coinciden estos valores con los que indica la teoría?.

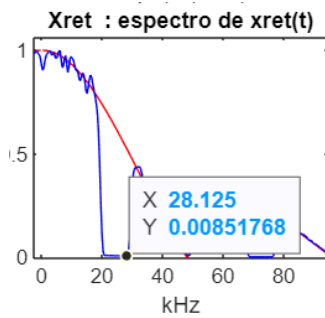
- Frecuencia de la primera banda imagen según teoría:

$$\frac{2\pi}{T} - \Omega_p = 48000 - 20000 = 28000\text{Hz}$$

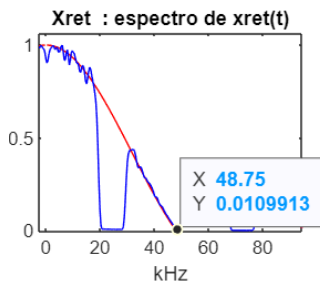
- Frecuencia central banda imagen según teoría:

La frecuencia central de la banda imagen será la frecuencia de muestreo, dado que esta frecuencia es la frecuencia a la que se va repitiendo la señal: 48000Hz

- Frecuencia inicio banda imagen según gráfica: 28125Hz



- Frecuencia central banda imagen según gráfica: 48750Hz



**Cuestión 6.** Según cálculos teóricos, ¿qué atenuación aplica el filtro de retención de orden cero sobre el inicio de la banda imagen?. Confirmar que el resultado coincide con el deducido a partir de la gráfica del espectro de xret(t).

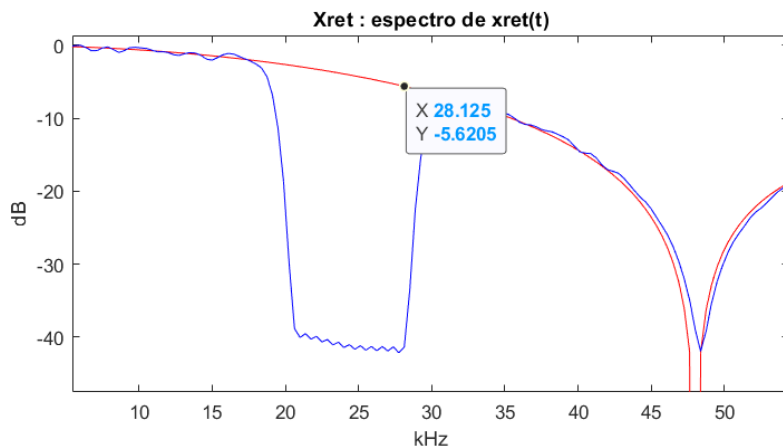
```

Fs=48000;
Fp=20000;
omega=(Fs-Fp)*2*pi;
Ts=1/Fs;
H0=abs(sin(omega*Ts/2)/(omega*Ts/2));
Att=-20*log10(H0) %dB

```

Att = 5.5625

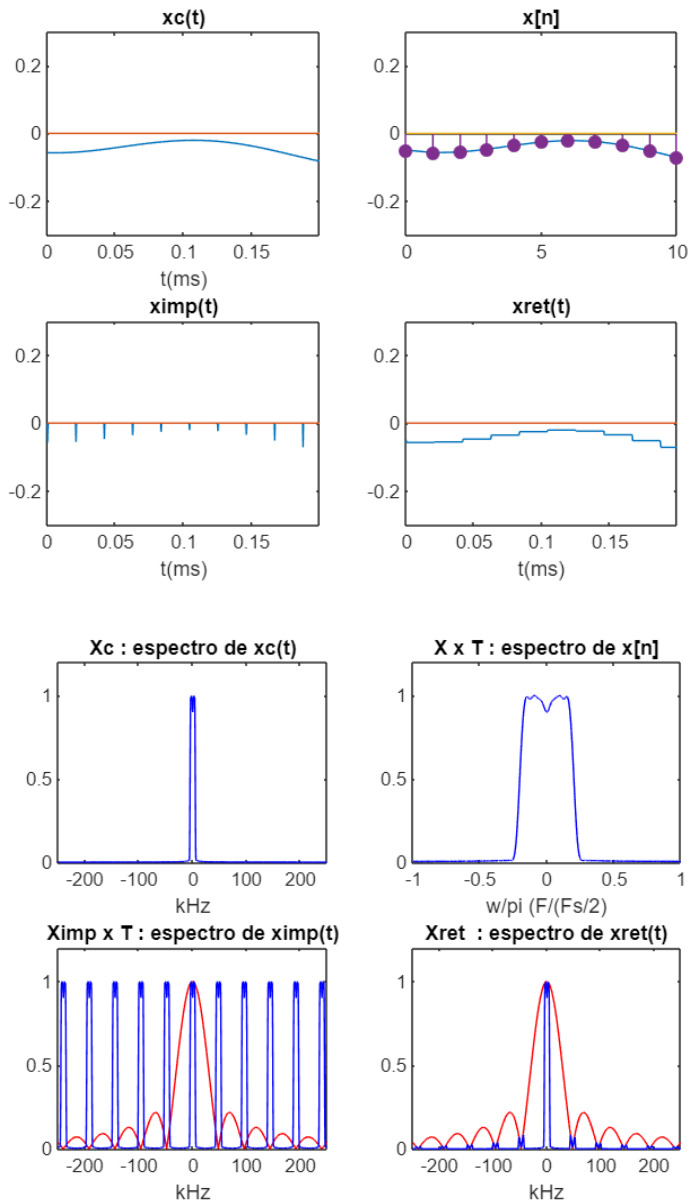
Según la gráfica, el valor de la atenuación es: 5.62 dB

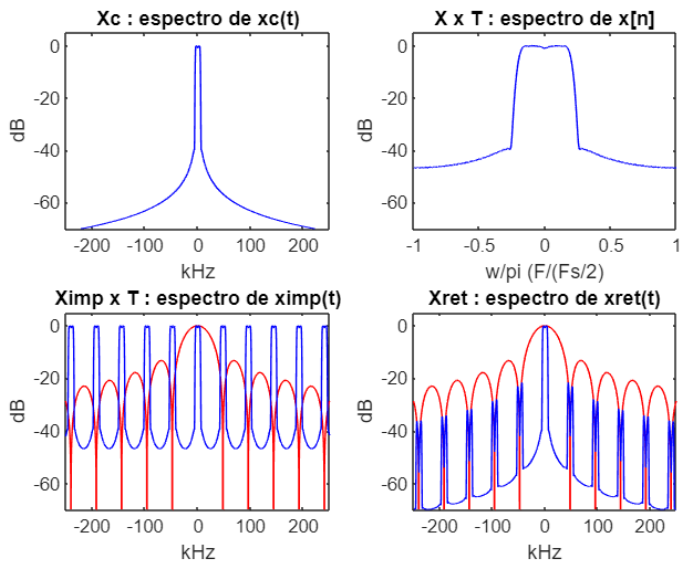




**Cuestión 7.** Repetir la simulación para una señal  $x_c(t)$  con una ancho de banda  $F_p=5\text{kHz}$ . Contestar de nuevo a las dos cuestiones anteriores.

practica12\_p7





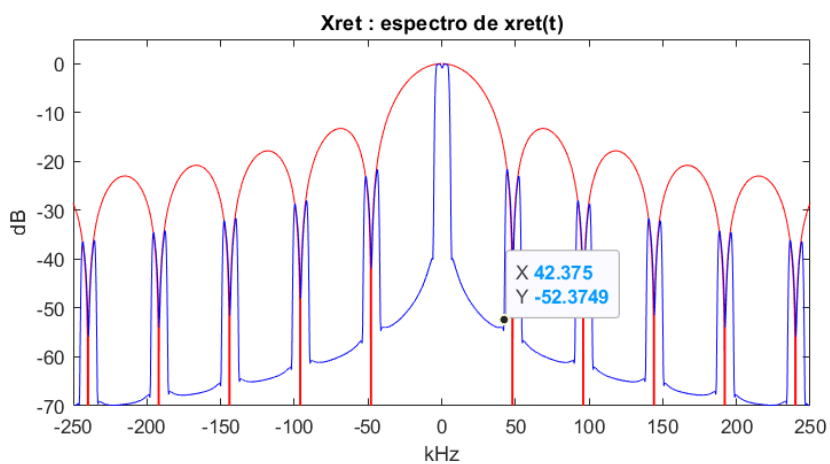
- Frecuencia de la primera banda imagen según teoría:

$$\frac{2\pi}{T} - \Omega_p = 48000 - 5000 = 43000 \text{ Hz}$$

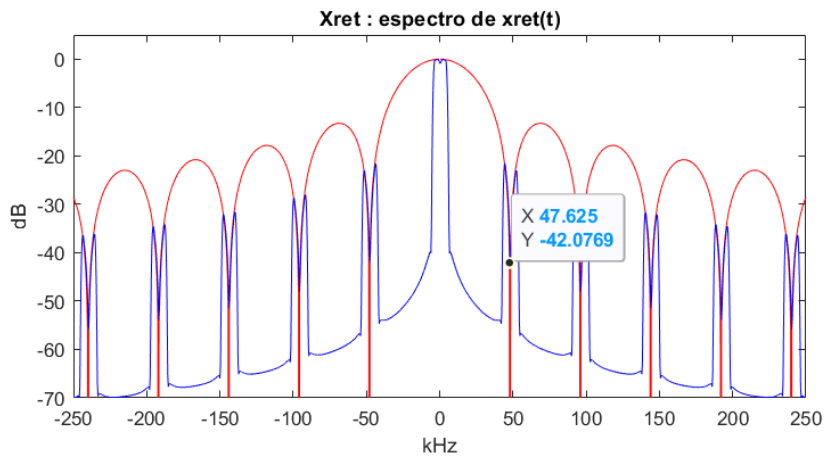
- Frecuencia central banda imagen según teoría:

La frecuencia central de la banda imagen será la frecuencia de muestreo, dado que esta frecuencia es la frecuencia a la que se va repitiendo la señal: 48000 Hz

- Frecuencia inicio banda imagen según gráfica: 42375 Hz



- Frecuencia central banda imagen según gráfica 47625 Hz



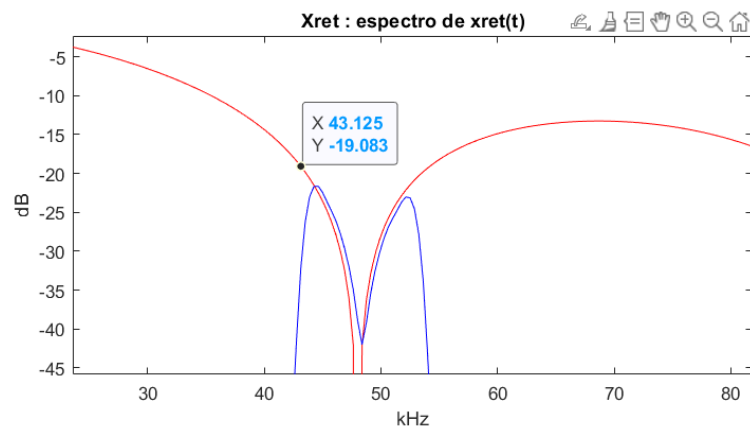
- Atenuación teórica:

% De manera teórica:

```
Fs=48000;
Fp=5000;
omega=(Fs-Fp)*2*pi;
Ts=1/Fs;
H0=abs(sin(omega*Ts/2)/(omega*Ts/2));
Att=-20*log10(H0) %dB
```

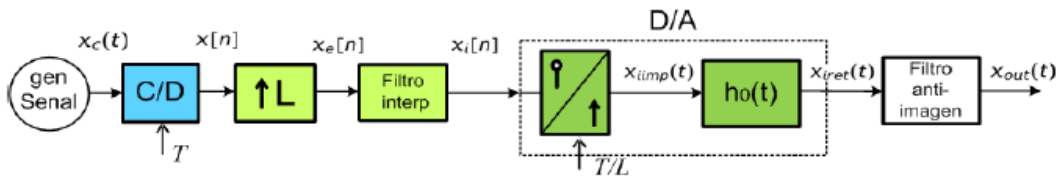
Att = 18.8456

Si miramos en la gráfica, obtenemos que el valor de la atenuación es: 19.083 dB



### 1.3. Conversión C/D ideal y conversión D/A con sobremuestreo

A continuación se estudia el efecto del sobremuestreo en la conversión D/A. Para ello consideramos el sistema de la figura 3 en el que se ha incluido un sistema interpolador por un factor  $L$  previo a la conversión. Se ha añadido un filtro analógico anti-imagen para completar el sistema.



**Figura 3. Sistema13**

Los parámetros para la simulación del *sistema13* se encuentran en el fichero *parametros13.m*. En el listado 3 se muestran los parámetros añadidos respecto a los dos sistemas anteriores.

**Listado 3. *parametros13.m***

```
%-----
% parametros 13 (además de los de parametros11)
%-----
% Parametros del Sistema
%-----
L=2;          % Factor de sobremuestreo para D/A

% Especificacion del filtro interpolador
ordenInterp=150; % orden del interpolador de sobremuestreo

% Especificacion del filtro anti-imagen
Rp=3;         % Rizado máximo en banda de paso: dB (+-)
Rs=30;        % Rizado máximo en banda de rechazo (dB)
Fpaso=Fp;     % Limite de la banda de paso
Fstop=L*Fs-Fp; % Comienzo de la banda de rechazo
```

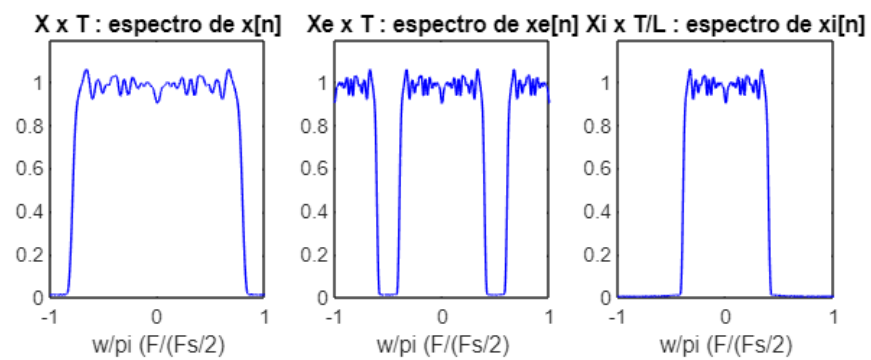
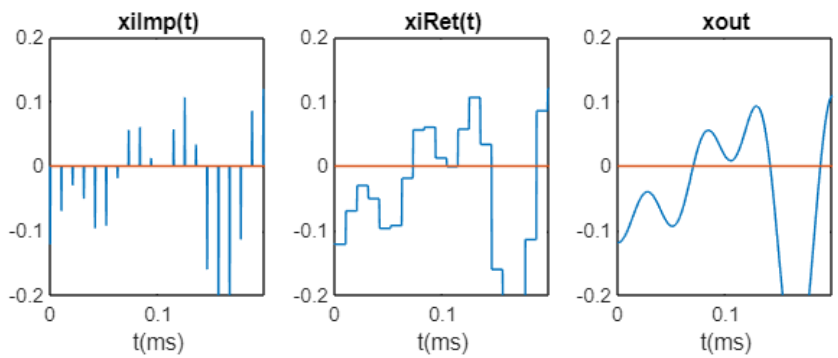
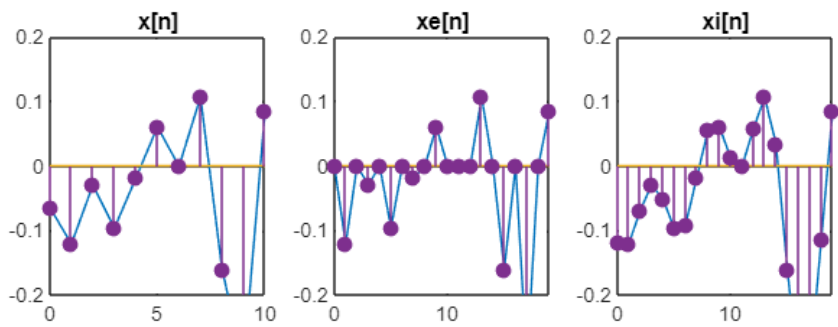
Simular el *sistema13* ejecutando el comando *practica13*. Como resultado deben aparecer tres gráficas.

En la primera se representa un tramo temporal de las 6 señales involucradas. En la segunda se muestran los espectros de las 6 señales en escala lineal y en la tercera los espectros en escala logarítmica.

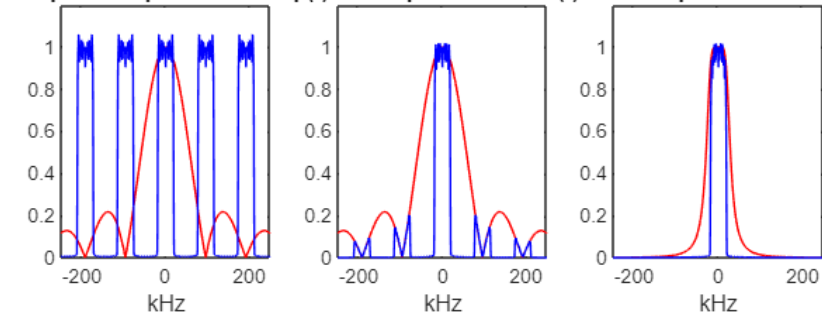
En la representación gráfica del espectro de  $x_{imp}(t)$  y de  $x_{ret}(t)$  se ha superpuesto en color rojo la respuesta en frecuencia del filtro de retención de orden cero.

En la representación gráfica de  $x_{out}(t)$  se ha superpuesto en color rojo la respuesta en frecuencia del filtro anti-imagen.

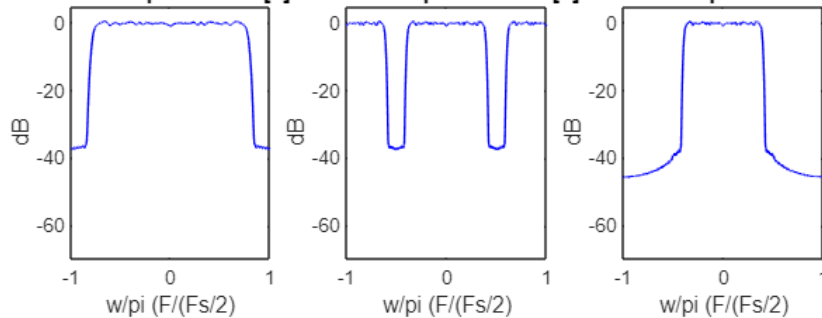
*practica13*



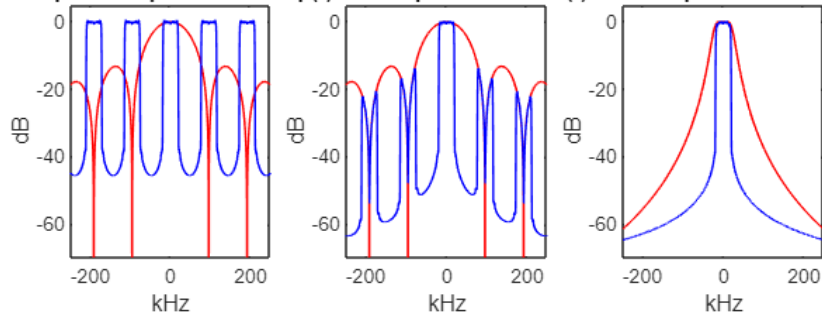
Xilmp x T : espectro de  $x_{ilmp}(t)$  Xiret : espectro de  $x_{iret}(t)$  Xout : espectro de  $x_{out}(t)$



X x T : espectro de  $x[n]$  Xe x T : espectro de  $x_e[n]$  Xi x T/L : espectro de  $x_i[n]$

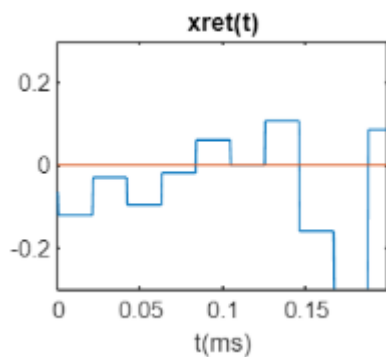


Xilmp x T : espectro de  $x_{ilmp}(t)$  Xiret : espectro de  $x_{iret}(t)$  Xout : espectro de  $x_{out}(t)$

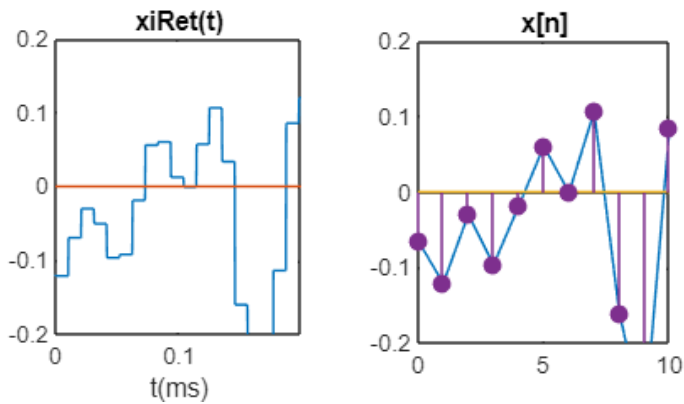


**Cuestión 8.** Describa las diferencias entre las señales  $x_{ret}(t)$  del sistema12 y  $x_{iret}(t)$  del sistema 13 en el dominio del tiempo y de la frecuencia.

Sistema 12

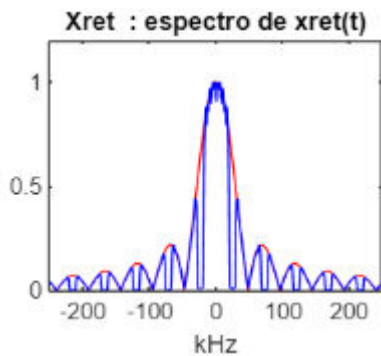


Sistema 13

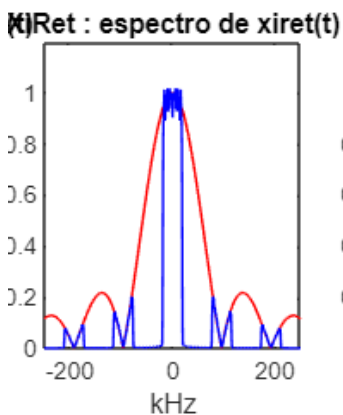


Al añadir el factor de sobremuestreo (Interpolación con el factor L), tenemos más muestras por lo que tenemos más información de ella, por lo tanto, conseguimos tener una señal final en el tiempo más completa. Respecto a los espectros de ambas señales:

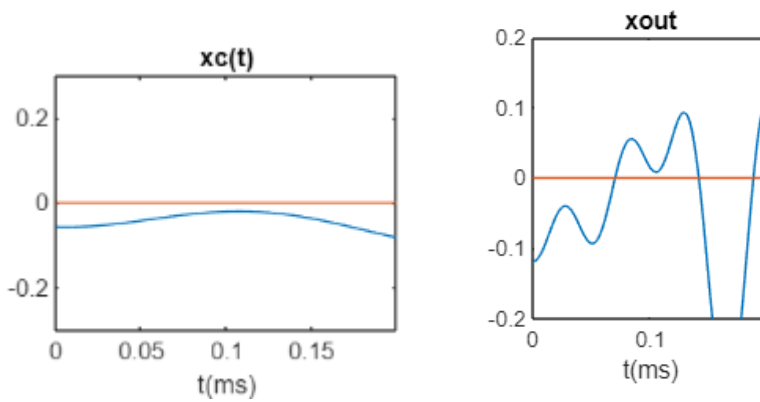
#### Sistema 12



#### Sistema 13



Dado que la señal del sistema 13 está sobremuestreada, en el espectro tenemos más información y vemos que el filtro de reconstrucción introduce menos imágenes que en el espectro del sistema 12. Esto siempre nos ocurrirá cuando sobremuestreemos la señal, cometeremos un error menor en las componentes espectrales así como en el dominio del tiempo. Si comparamos la señal original que estamos procesando:

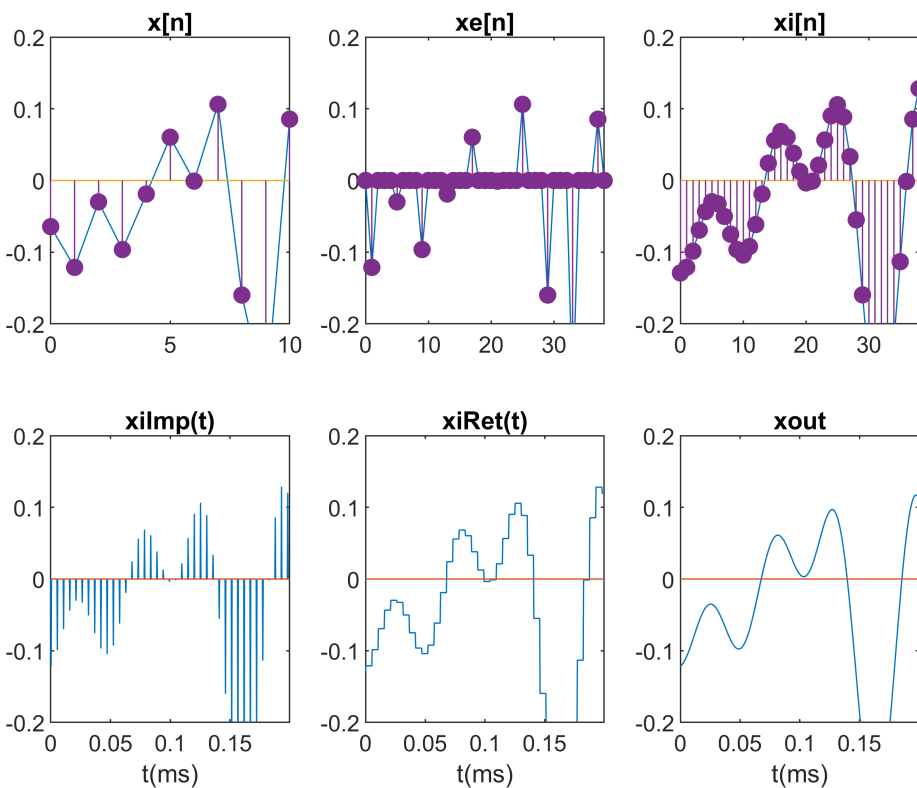


vemos que la señal más precisa es la sobremuestreada.

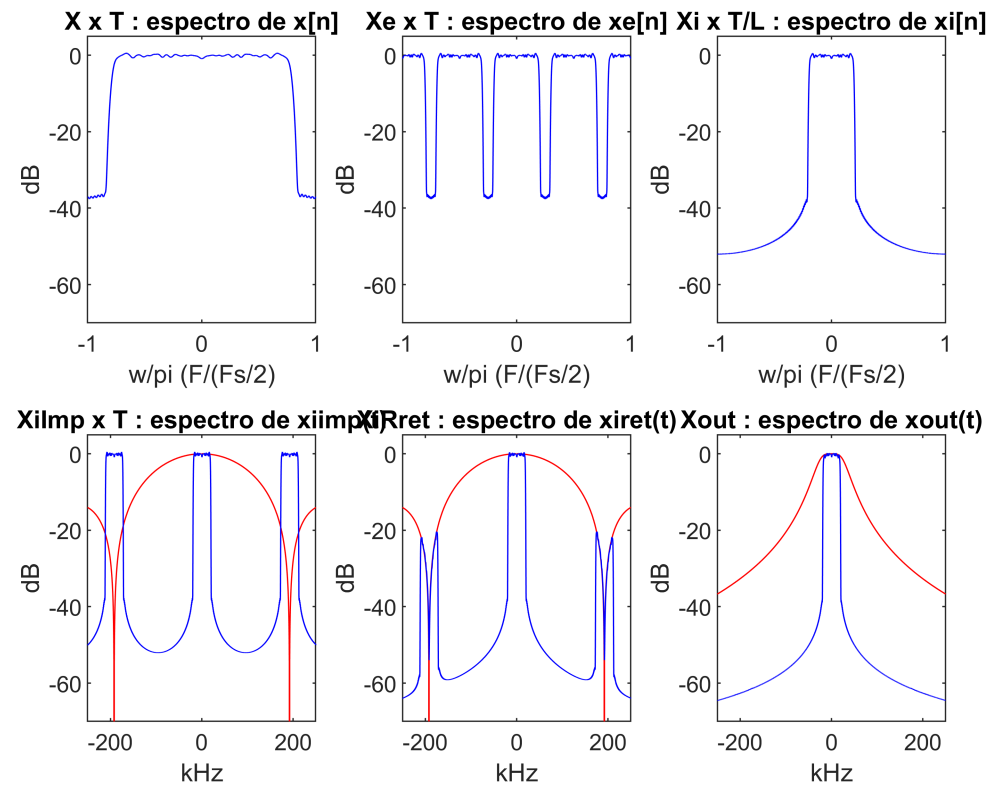
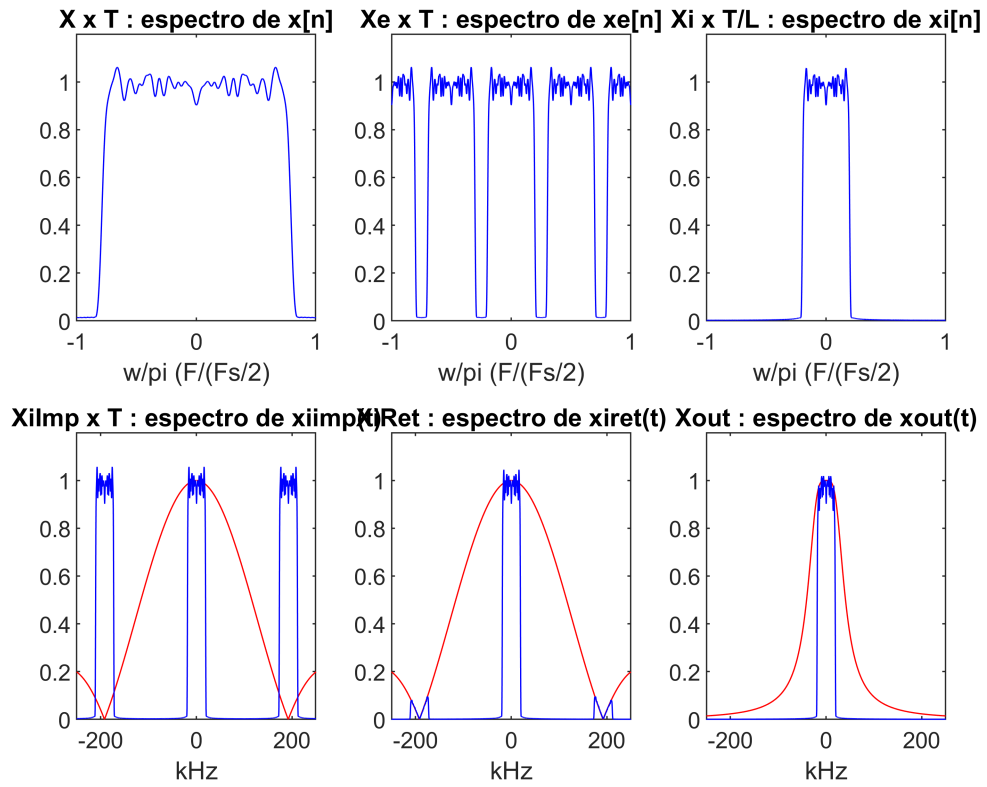
**Cuestión 9.** ¿Mejoraría o empeoraría la reconstrucción de la señal si se aumenta el valor de  $L$ ? Verificarlo repitiendo la simulación para  $L=4$ .

Al aumentar el sobremuestreo, mejoraría la reconstrucción porque tendríamos más muestras de la señal de entrada y eso hace que, en frecuencia, las muestras estén más lejos por lo que sería menos probable que se produzca aliasing y a la salida obtendríamos la señal de entrada buena.

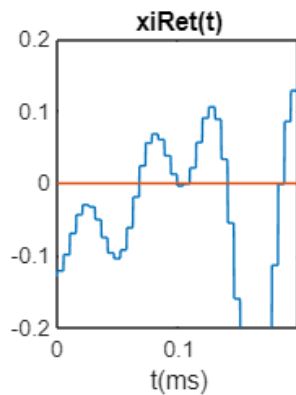
```
practica13_9 % cambiamos L=4
```



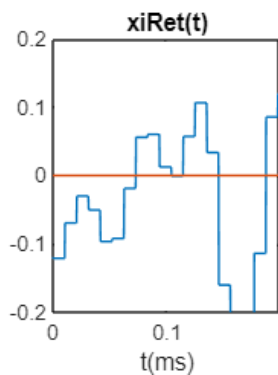




Para  $L=4$



Para  $L=2$



Podemos comprobar que para  $L=4$  tiene una mejor reconstrucción que para  $L=2$ , además de que tendría más forma de señal continua.

**Cuestión 10.** Teclear `help expande` para ver el formato de la rutina de expansión. Siguiendo ese formato, diseñar una rutina propia que realice la expansión de una señal por un factor  $L$ . Probar su funcionamiento con una señal sencilla y posteriormente añadirla a la librería. Renombrar previamente la rutina original para no perderla. Ejecutar la simulación del sistema 13 de nuevo usando la rutina diseñada y confirmar que funciona correctamente. (NO SE SI ES CORRECTO)

`help expande`

-----  
Expansion  
-----

`xe = expande(x, L)`  
-----

x: vector de señal de entrada  
xe: vector de señal de salida expandido  
L: factor de expansión  
-----

Por lo que, la señal  $x$  es la señal de entrada cualquiera y  $L$  es el factor de expansión.

```

% -----
% Expansion
% -----
% xe = expande(x, L)
% -----
% x:   vector de señal de entrada
% xe:  vector de señal de salida expandido
% L:   factor de expansión
% -----

function xe = expande(x, L)
N=length(x);
xe=zeros(1,L*N);
for k=0:N-1,
    xe(1+k*L)=x(1+k);
end

```

La función `expande`, lo que hace es multiplicar la señal discreta de entrada y multiplicarlo con un factor  $N$  y lo que hace es introducir ceros de por medio. Si lo hacemos nosotros, tendremos que meter muestras intermedias que sean cero, de tal manera:

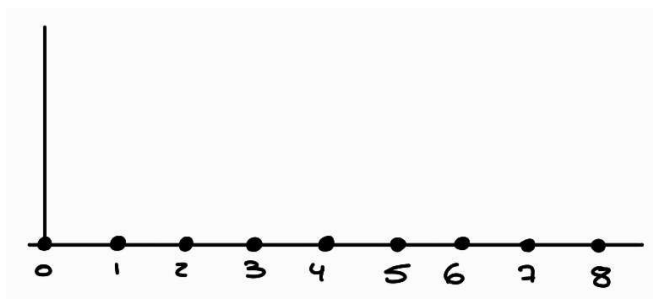
```

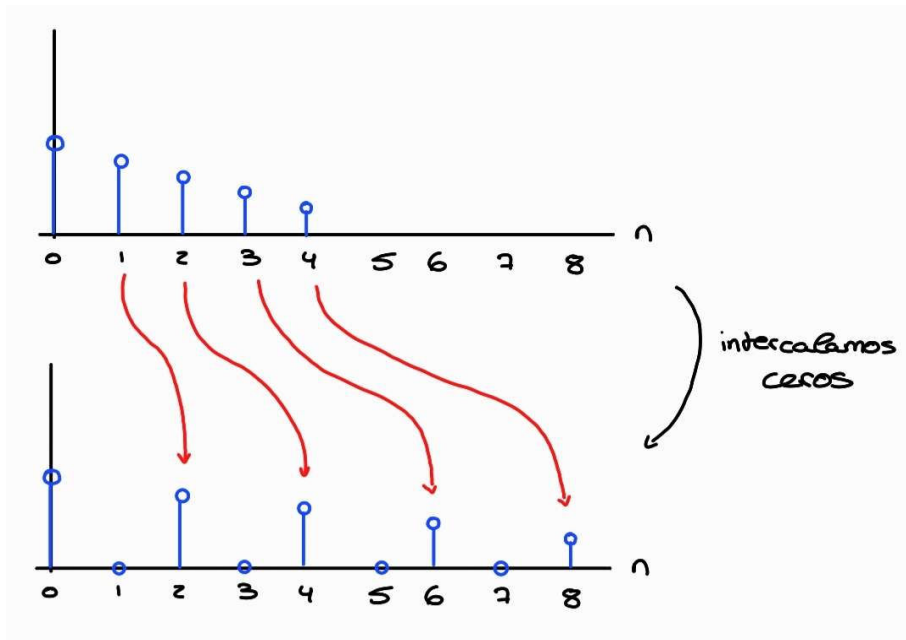
% -----
% Expansion
% -----
% xe = expande(x, L)
% -----
% x:   vector de señal de entrada
% xe:  vector de señal de salida expandido
% L:   factor de expansión
% -----

function xe=expande_10(x,L)
xe=zeros(1,L*length(x));
n=1:length(x); % crea un array que va desde 1 hasta el tamaño de la señal de
% entrada (empieza en 1 porque los arrays en matlab siempre empiezan en 1)
np=1:L:length(xe);
xe(np)=x(n); % nos saltamos las muestras en las que hemos introducido el
% cero y vamos a poner las muestras que son distintas a cero en las posiciones
end

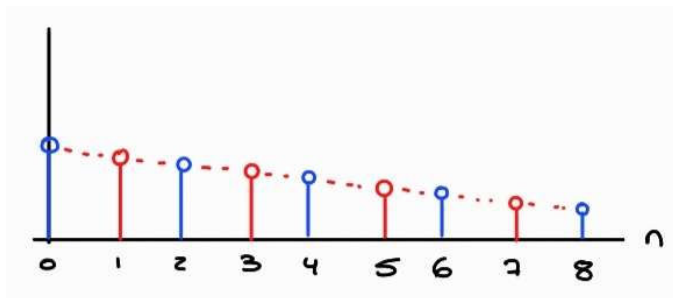
```

Con esto, cada muestra que tengamos de la señal de entrada, cogeremos la señal  $1$  y nos la llevamos a su punto determinado, y así con todo:





Estamos intercalando  $L-1$  ceros entre dos muestras y con la interpolación crearemos nuevas muestras como resultado de la operación, es decir, creando las muestras azules:



Con la interpolación aumentamos la frecuencia de muestreo aparente, ya que estamos introduciendo ceros entre muestras y muestras, es decir, aumentamos la frecuencia de muestreo del sistema discreto para alejar las componentes en el espectro. Esto es sólo posible cuando la frecuencia de muestreo original es buena (verifica Nyquist, en ideal, en una real lo que debemos hacer es que el solape sea menor, es decir, siempre al muestrear perdemos información a no ser que la señal sea muy limitada en banda (no es lo normal), ya que las señales reales son no limitadas en banda (tienen espectros infinitos)).