Table of Contents

Práctica 2

Grupo 11

Nombre_1 Teresa González Gacía

Nombre_2 Miguel Oleo Blanco

```
close all; format compact; clear all
```

Ejercicio 1

La función my_coseno toma los siguientes parámetros de entrada:

N: número de muestras a representar

Ts: paso o periodo de muestreo (en s)

f: la frecuencia del coseno (en Hz)

phi: la fase del coseno (en radianes)

A: la amplitud del coseno (en V)

my_coseno devuelve:

t: el vector de referencia de tiempos;

y: el valor del coseno para dicho vector.

La función debe probarse para N=1000, Ts=10 microsegundos, f=1 KHz, phi=0 rad y A= 1 V (ojo con las unidades), por lo que la llamada será la siguiente:

[t, y] = my_coseno(N, Ts, f, phi, A);

Debe representar la función con el eje de tiempos etiquetado en milisegundos

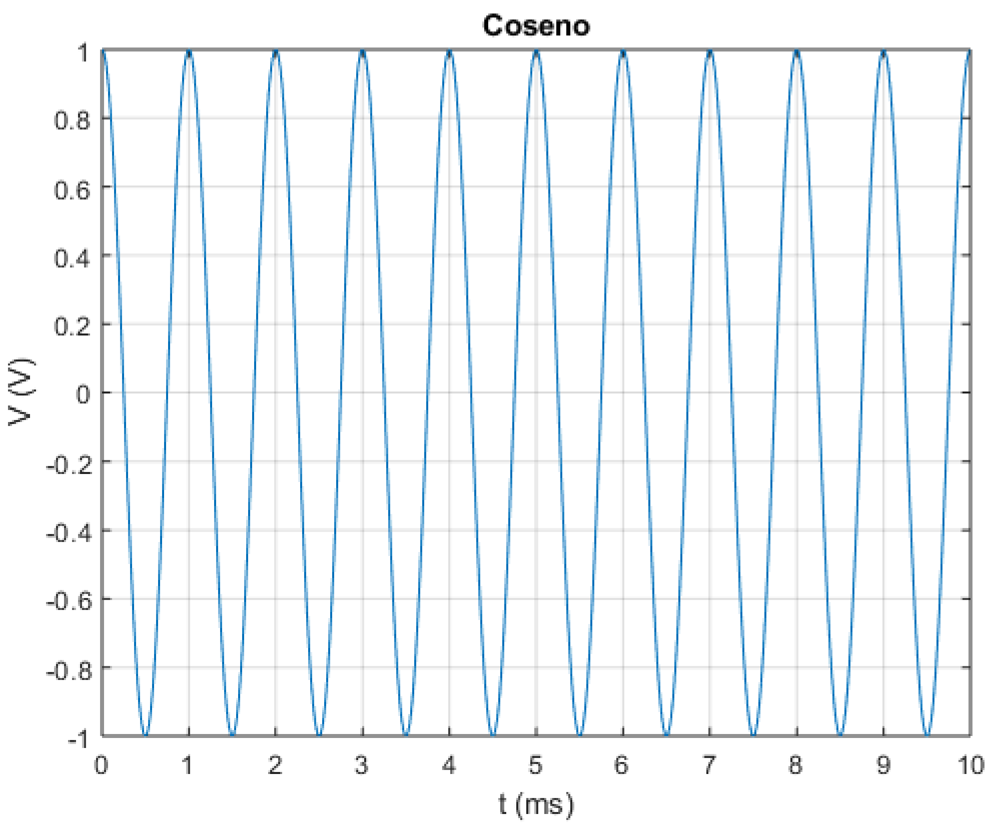
```
%Configuración de los parámetros de entrada según se indica en el
 enunciado
N=1000;
Ts=10*10^-6;
f=1000;
phi=0;
```

```
A=1;

% Llamada a my_coseno

[y,t] = my_coseno(N, Ts, f, phi, A);

% Representación temporal
t=t*1000; %Lo multiplicamos para dajarlo en ms;
figure;
plot(t,y); %configure la llamada a plot adecuadamente
title('Coseno');
xlabel('t (ms)');
ylabel('V (V)');
grid on
```



Incluya aquí su análisis ¿Cuál es el período de la señal?

En la gráfica se puede observar fácilmente como el periodo es de 1 ms.

¿Con qué parámetro de entrada de my_coseno está relacionado?

el periodo del coseno está relacionado con el parámetro f, ya que 1ms=1/f.

Ejercicio 2

t debe ir de 0 a 3 ms

La frecuencia de muestreo es 1 MHz. ¿Cuál es la relación de este parámetro con los parámetros de entrada de la función my_coseno?

Fs está relacoionado con el parámetro de entrada f, ya que Fs >> f, por lo que se cumple que Fs>2f (Teorema de Nyquist-Shannon) y por lo cual no se produce solapamiento.

x1: coseno con amplitud 1 V, frecuencia 10 KHz y ningún desfase

x2: coseno con amplitud 2 V, frecuencia 23 KHz y un desfase positivo de 2 rad;

x3: coseno con amplitud 2 V, frecuencia 7KHz y un desfase negativo de 1 rad.

x = x1 + x2 + x3;

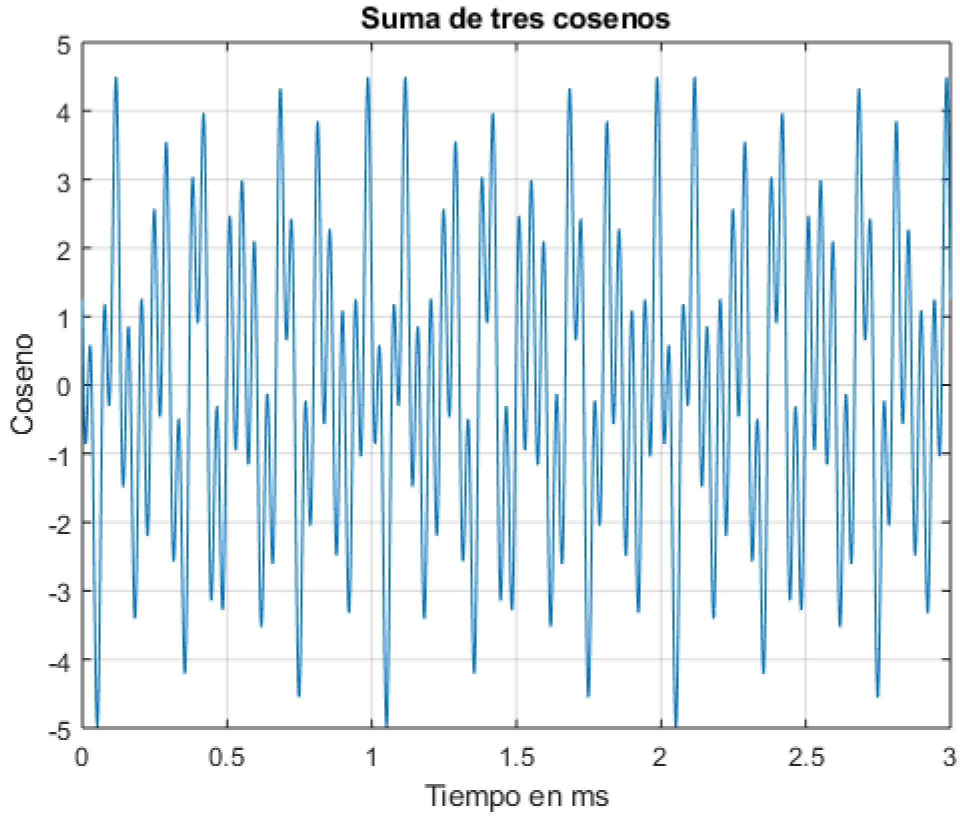
```
    fs=10^6;

% Genere x1 a continuación
    [cos,temp]=my_coseno(3001,1/(10^6),10*10^3,0,1);

% Genere x2 a continuación
    [cos1,temp1]=my_coseno(3001,1/(10^6),23*10^3,2,2);

% Genere x3 a continuación
    [cos2,temp2]=my_coseno(3001,1/(10^6),7*10^3,1,2);
% Calcule la señal compuesta como la suma de esos 3 cosenos

    coseno=cos+cos1+cos2;
% Represente la señal obtenida en el dominio del tiempo entre 0 y 3 ms
% Etiquete el eje de abscisas y el de ordenadas y muestre rejilla
figure
plot(temp*1000,coseno) % El tiempo lo multiplicamos por 1k para que se
 vea en ms
hold on
title('Suma de tres cosenos')
xlabel('Tiempo en ms')
ylabel('Coseno')
grid on
```



Incluya aquí su análisis ¿Cuál es el período de x?

A través de la gráfica se puede observar que el periodo de x es de 1 ms;

¿Corresponde con la frecuencia fundamental esperada?

Si corresponde con lo esperado, ya que la frecuencia fundamental de la suma de los tres cosenos, es el mínimo común multiplo entre las tres frecuencias fundamentales de cada coseno.

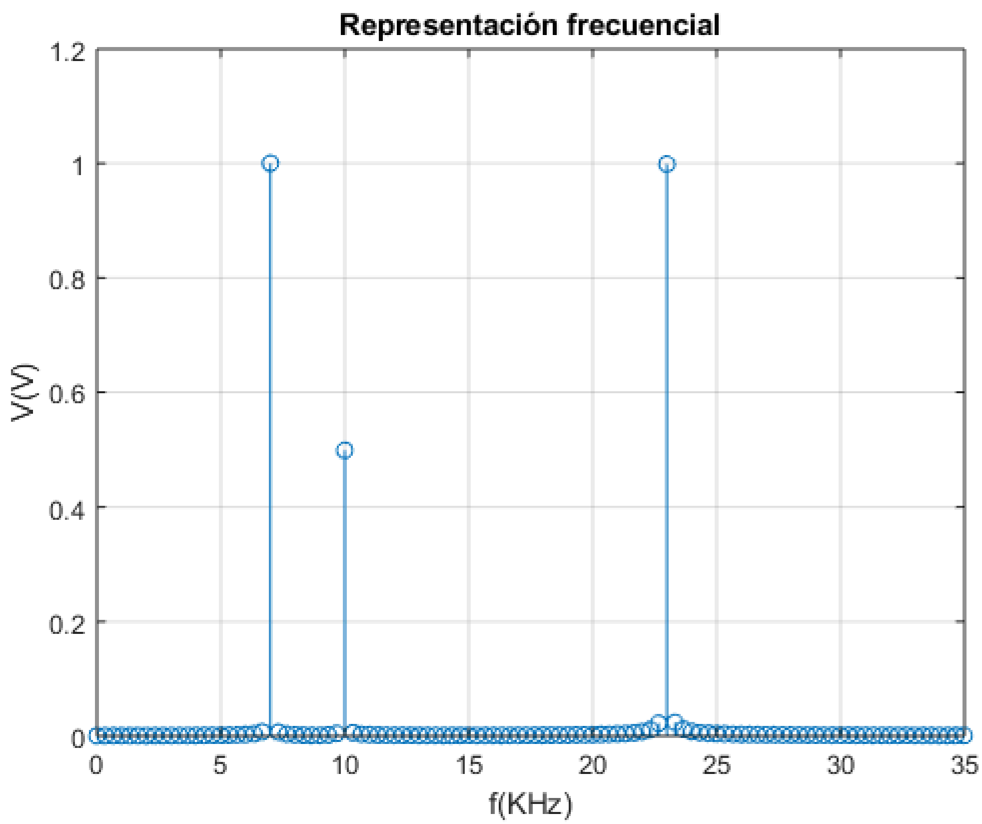
Debido a esto -> mcm(7,10,23)Khz = 1KHz 1ms.

A continuación se proporciona el código para calcular la FFT de x

```
NFFT = length(coseno);         % Longitud de la FFT
X    = fft(coseno,NFFT)/NFFT;  % X como transformada de Fourier de x
df   = fs/NFFT;           % Resolución en frecuencia
lg   = floor(NFFT/2);     % Se redondea NFFT/2 hacia abajo, por si
 NFFT es impar
f    = 0:df:(lg-1)*df;    % Vector de frecuencias

%
figure
stem(f/1000,abs(X(1:lg))); %dividimos la frecuencia entre 1k para pasa
 de Hz a kHz
hold on                   % Configure la llamada a stem para
 representar en el eje X la frecuencia en KHz                    %
 Ajuste el eje de frecuencias para mostrar con claridad las líneas
 espectrales de la señal
```

```
xlabel('f(KHz)');          % Unidades del eje de abscisas
ylabel('V(V)');            % Unidades del eje de ordenadas
title('Representación frecuencial');
axis([0 35 0 1.2])  %Ajusto los ejes para que se vean mejor
grid on;
```



Análisis

Incluya aquí su análisis del módulo del espectro de x. Analice y justifique si los valores son los esperados tanto en el eje de abscisas como en el de ordenadas Mencionar amplitudes y frecuencias

El resultado de la gráfica de la representación en frecuencia concuerda con lo esperados

Primero: tenemos una delta a 7 KHz y con una amplitud de 1V. Concuerda ya que la f del cos = 7KHz y su amplitud es 2V (en frecuencia 2/2 = 1).

Segundo: tenemos una delta a 10 KHz y con una amplitud de 05V. Concuerda ya que la f del cos = 10 KHz y su amplitud es 1V (en frecuencia 1/2=0.5).

Tercero: tenemos una delta a 23 Khz y con una ampitud de 1V. Concuerda ya que la f del cos = 23KHz y su aplitud es 2V (en frecuencia 2/2 = 1).

Ejercicio 3

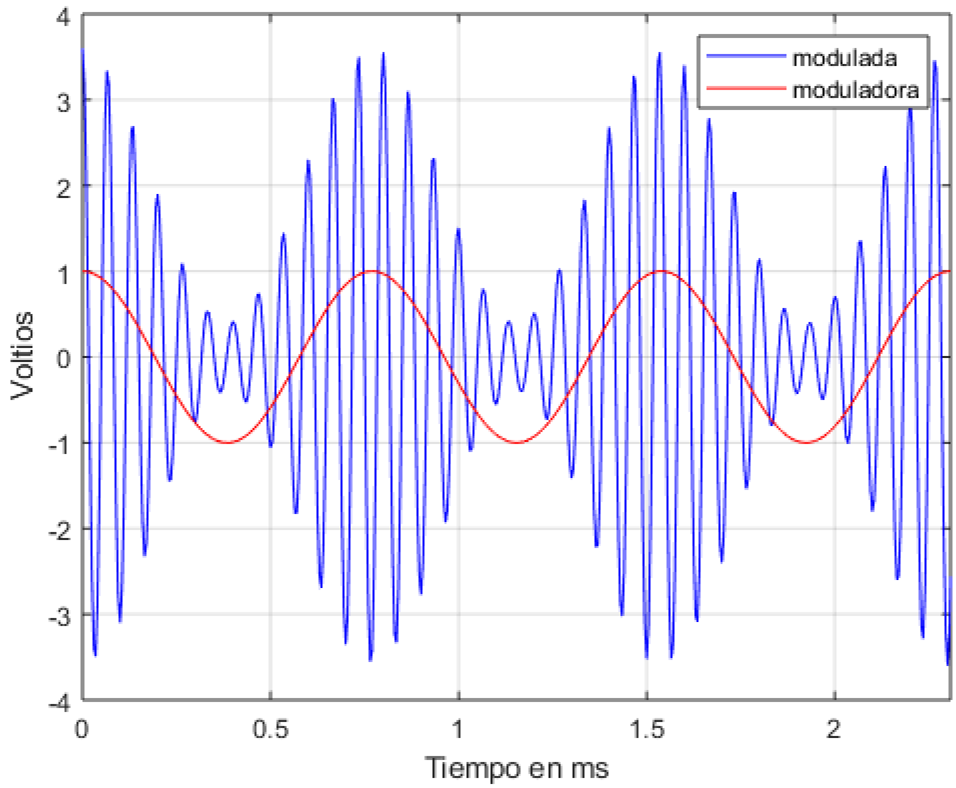
Programe la función modulaAM que toma los siguientes parámetros de entrada: fm: frecuencia de la señal moduladora (Hz); mu: índice de modulación (en porcentaje); fs: frecuencia de muestreo (Hz); fc: frecuencia de la señal portadora (en Hz); Ac: amplitud de la señal portadora (en V).

```
% Los parámetros de salida son:
% t: la referencia temporal (debe cubrir varios periodos de la
 moduladora);
% x_info_n: la señal moduladora;
% y_AM: la señal modulada en AM en el dominio del tiempo

% Configure los parámetros de entrada según se indican en el enunciado
fm = 1300;      % Frecuencia de la señal moduladora (Hz)
mu = 80;      % Indice de mdulación (%)
fs = 200000;      % Frecuencia de muestreo (Hz)
fc = 15000;      % Frecuencia de la señal portadora (Hz)
Ac = 2;      % Amplitud de la señal portadora (V)

% Llame a la función modulaAM
[t,x_info_n, y_AM] = modulaAM(fm, mu, fs, fc, Ac);

% Represente en la misma figura la señal modulada en AM (en azul) y la
 señal
% moduladora (en rojo). Las unidades del eje temporal deben ser ms.
% Etiquete los ejes adecuadamente, ajústelos si lo considera necesario
 e incluya una leyenda que permita identificar cuál es cada señal.
figure
title ('Señal modulada y señal moduladora');
plot(t*1000,y_AM,'b')
hold on
axis([0 3000/fm -4 4]);
plot(t*1000,x_info_n,'r');
xlabel('Tiempo en ms');
ylabel('Voltios');
legend ('modulada','moduladora');
grid on
```



ANÁLISIS

Como se puede observar, la señal resultante no sufre sobremodulación, y era lo que esperábamos puesto que el índice de modulación es 0.8.

Hemos ajustado los ejes para que se puedan ver 3 periodos de muestreo. Además, hemos tenido en cuenta el valor máximo de la envolvente (3.6) y mínimo (0.4) al ajustar el eje y.

```
% Compare el periodo de la envolvente de la señal modulada con el
 periodo
% de la moduladora.
% Como se puede ver, el periodo de la modulada es menor porque se
 encuentra
% a mayor frecuencia.
%
% Compruebe si el índice de modulación es el esperado.
%
% m=Amax-Amin/2*Vmedio= (3.6-0.4)/2*2)=0.8 CORRECTO
```

Calcule la TF de la señal info

```
    NFFT = length(x_info_n);          % Longitud de la FFT
    X    = fft(x_info_n,NFFT)/NFFT;   % X como transformada de Fourier
 de x
    df   = fs/NFFT;                % Resolución en frecuencia
    lg   = floor(NFFT/2);          % Se redondea NFFT/2 hacia abajo, por si
 NFFT es impar
    f    = 0:df:(lg-1)*df;
```

```
% Calcule la TF de la señal modulada

     NFFT = length(y_AM);         % Longitud de la FFT
    X2    = fft(y_AM,NFFT)/NFFT;  % X como transformada de Fourier de
 x
    df2   = fs/NFFT;           % Resolución en frecuencia
    lg2   = floor(NFFT/2);     % Se redondea NFFT/2 hacia abajo, por
 si NFFT es impar
    f2    = 0:df2:(lg2-1)*df2;
% Utilizamos la funcion FFT proporcionada por Matlab

% Represente en la misma figura el espectro del módulo de la señal
 modulada en AM (en azul) y de la señal
figure
plot(f/1000,abs(X(1:lg)),'r');
axis([0 18 0 3]);

hold on
plot(f/1000,abs(X2(1:lg2)),'b');
title ('Señal moduladora y señal modulada');
xlabel('f(KHz)');
ylabel('V(V)');
grid on
legend ('moduladora', 'modulada');
```

ANÁLISIS

Para ajustar los ejes hemos tenido en cuenta las frecuencias esperadas de la señal modulada al ser desplazadas debido a la frecuencia de la portadora (entre 14 y 18 KHz). También se muestra el espectro de la señal de información. Su amplitud esperada es 0.5, por lo que nos coincide. Sin embargo, no sabemos por qué la amplitud de la moduladora nos sale ligeramente atenuada y en vez de salirnos 1 en el piloto y 0.8 en las bandas laterales nos sale unos valores un poco más pequeños.

Ejercicio 4

Programe la función demodulaAM que toma los siguientes parámetros de entrada: y_mod: Señal modulada (en el tiempo); fs: Frecuencia de muestreo (Hz); fc: Frecuencia de la señal portadora (en Hz); filtro: Filtro paso bajo ajustado a la frecuencia de la señal moduladora.

Los parámetros de salida son:

t: la referencia temporal; x_demod_before_filt: la señal demodulada, en el dominio del tiempo, antes del filtrado x_demod_after_filt: la señal demodulada, en el dominio del tiempo, después del filtrado

```
ts=1/fs;
t = 0:ts:N;
% Cargue el filtro

load LowPassFilter
% Demodule la señal modulada en el Ejercicio 3 llamando a demodulaAM
[t,demod_final,demod]=demodulaAm(y_AM,fs,fc,LowPassFilter);
% Represente en el dominio del tiempo la señal demodulada (antes y
 después de aplicar el filtrado) y la señal moduladora.
```

```
% El eje de tiempos debe estar en milisegundos.
% Las tres señales se representarán en 3 plots independientes de la
 misma imagen (subplot).
% La primera en verde, la segunda en rojo, con trazo discontinuo, y la
 tercera en rojo, con trazo continuo
% Etiquete los ejes adecuadamente, ajústelos si lo considera necesario
 e incluya una leyenda que permita identificar cuál es cada señal.

figure
t=t*1000; %milisegundos

subplot(3,1,1)
plot(t,demod,'g');
hold on
title('demodulada antes del filtro');
xlabel('t(ms)');
ylabel('V(V)');
grid on;
axis([0 5 0 4]);

subplot(3,1,2)
hold on
plot(t,demod_final,'r-');
title('demodulada después del filtro');
xlabel('t(ms)');
ylabel('V(V)');
grid on;
axis([0 5  -4 4])


subplot(3,1,3)
hold on
plot(t,x_info_n,'r')
title('Moduladora');
xlabel('t(ms)');
ylabel('V(V)');
grid on;
axis([0 5 -1 1]);
```

ANÁLISIS

La demodulación que hemos utilizado es una demodulación coherente (aunque podría servir el detector de envolvente ya que no hay sobremodulación). En la misma hemos multiplicado la señal modulada por un coseno con la misma frecuencia que la portadora, por lo que dicha señal llamada "demodulada" se encuentra en una frecuencia el doble de mayor y por ello en la primera gráfica se ve la parte oscura ya que altas frecuencias son variaciones rápidas en el tiempo. Por otro lado, cargamos un filtro paso bajo con frecuencia de corte la frecuencia de laseñal de información, por loque filtra la señal "demodulada" limitándola en banda por lo que "demod_final" se queda con frecuecia muy baja (1300 Hz) y en el tiempo desaparece la zona oscura. La señal moduladora o de información se representa sin la continua. Hemos ajustado los ejes para que se vean varios periodos de muestreo y teniendo en cuenta los valores máximos y mínimos de la envolvente.

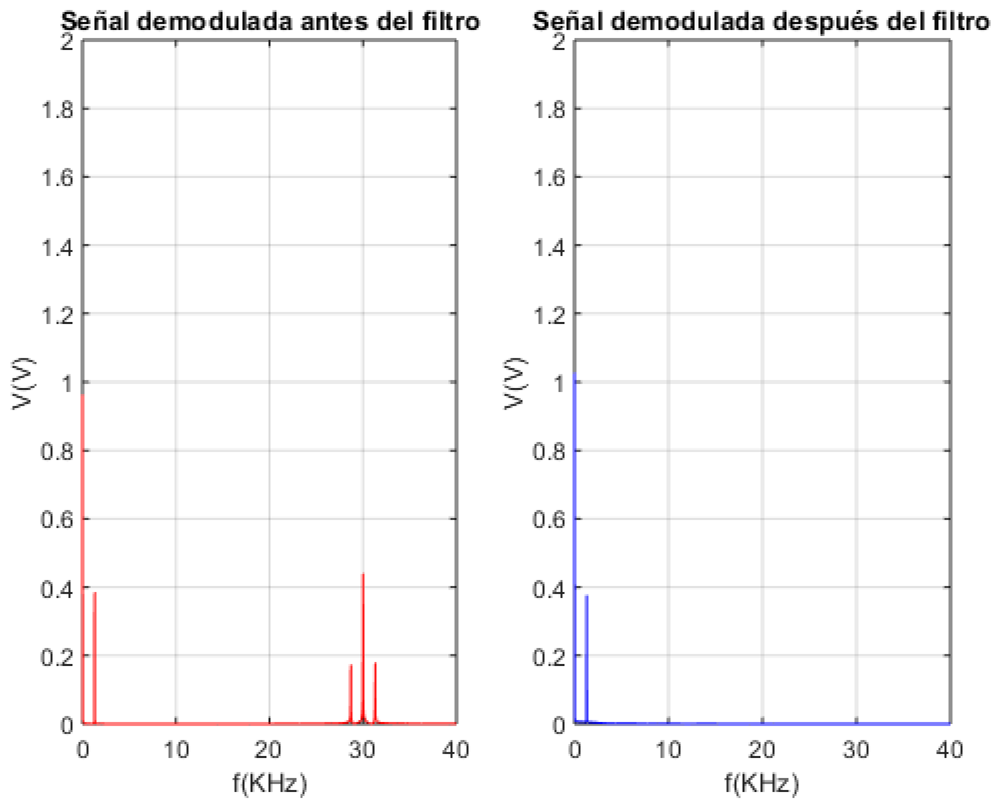
Represente en el dominio de la frecuencia la señal demodulada antes y después de aplicar el filtrado. El eje de frecuencias estará en KHz. Las dos señales se representarán en el mismo plot. La primera en verde y la segunda en rojo, con trazo discontinuo. Etiquete los ejes adecuadamente, ajústelos si lo considera necesario e incluya una leyenda que permita identificar cuál es cada señal.

```
   NFFT = length(demod);         % Longitud de la FFT
   X    = fft(demod,NFFT)/NFFT;  % X como transformada de Fourier de
 x
   df   = fs/NFFT;           % Resolución en frecuencia
   lg   = floor(NFFT/2);     % Se redondea NFFT/2 hacia abajo, por si
 NFFT es impar
   f    = 0:df:(lg-1)*df;


   NFFT = length(demod_final);         % Longitud de la FFT
   X2   = fft(demod_final,NFFT)/NFFT;  % X como transformada de
 Fourier de x
   df2   = fs/NFFT;           % Resolución en frecuencia
   lg2   = floor(NFFT/2);     % Se redondea NFFT/2 hacia abajo, por
 si NFFT es impar
   f2    = 0:df:(lg2-1)*df;

% de nuevo hemos calculado la TDF de las dos señales mediante el
 comando
% FFT teniendo en cuenta que Matlab no divide la señal entre el numero
% de puntos (NFFT). Cabe destacar que la longitud de ambas señales es
 la
% misma.
   figure
   subplot(1,2,1);
   plot(f/1000,abs(X(1:lg)),'r');
   axis([0 40 0 2]);
   grid on
   title ('Señal demodulada antes del filtro');
   xlabel('f(KHz)');
   ylabel('V(V)');

   subplot(1,2,2)
   plot(f2/1000,abs(X2(1:lg2)),'b');
   axis([ 0 40 0 2]);
   title ('Señal demodulada después del filtro');
   xlabel('f(KHz)');
   ylabel('V(V)');
   grid on
```



ANÁLISIS

Como se esperaba en el espectro de frecuencias de la "demodulada" nos encontramos con que las deltas se desplazan al doble de la frecuencia de la portadora; sin embargo después del filtro en la "demod_final" nos quedamos solo con el espectro de frecuencias menor que la frecuencia de la señal de información.

Published with MATLAB® R2018b