
```
clc , clear all , close all
```

Modulacion Julian Nieto

```
%fs = 800 ; Ts = 1/fs ;timeSig = 0.0012 ; n = fs * timeSig ; t = 0:
    Ts : n - Ts; % t = 0: Ts :(n-1) * timeSig ; % t = linspace(0,p,f); %
    muestreo y long de la signal
% fs = 800 ; Ts = 1/fs ; t = linspace(0,Ts,fs) ; % timeSig = 1 ; t =
    0 : Ts : timeSig - Ts; % t = linspace(0,Ts,fs);
archivo = 'inputAudio.wav' ; [moduladora , fs] = audioread(archivo) ;

% convertir a mono
moduladora = moduladora(:,1) ;

t = [0 : length( moduladora) -1] / fs; % representar el arreglo
    en segundos ya que la longitud de la entrada est? en muestras 0 :
    length( moduladora) -1 -> por el indexado de matlab
% para poner en terminos de segundo [n] * 1 / fs = [n] * [seg / n] =
    seg
t = t(:) ; %5 transponer t a columna[:,1]

%
% fMod = 1000 ; TMod = 1/fMod ; aMod = 1;
% tMod = linspace(0,TMod ,fMod);
% moduladora = aMod*sin(2*pi*fMod*t);

% figure(1); plot(t , moduladora) ; title(' Moduladora')

% fCarri = 100000 ; TCarri = 1 / fCarri ; aCarri= 1; % fs = 1000000 ;
    ojo se suele usar lo frec como moduladora y la input como carry; si
    es al contrario cambia la frec d emuestreo

fCarri = 100000 ; TCarri = 1 / fCarri ; aCarri= 0.62; % se escala la
    amplitud
DCOffset = 0.5 ; % se considera un
    dc offset par aajustarlo a la amplitud de la senal de entrada
portadora = aCarri *cos(2 * pi * fCarri * t) + DCOffset ;

portadora = portadora(:) ; % transponer a columna la portadora
%figure(2) ; plot(t , portadora) ; title(' Portadora') ; grid on

% ajustar tamano de arreglos
diferencia = length(portadora) - length(moduladora) ; ajuste =
    zeros(diferencia,1);
moduladora = [ moduladora ; ajuste ] ;

modulada = portadora .* moduladora ;
% figure(2) ;plot(t , modulada ) ; title(' Senal Modulada') ; grid on
```

```

nfft = length(modulada) ; % longitud de la senal a analizar en fft en
    tiempo
nfft = 2 ^ nextpow2(nfft) ; % el doble de resolucion en resolucion de
    potencias https://www.mathworks.com/help/matlab/ref/nextpow2.html

dftModulada = fft(modulada , nfft); % dft de la senal

mModulada = abs(dftModulada); % magnitud
mModulada = mModulada(1 :nfft / 2); % debido a la simetria d ela fft
mModuladaNormal = mModulada / max(mModulada);

p = unwrap(angle(dftModulada)); % fase

nFft = fs * (0 : nfft/ 2 - 1) / nfft; % nFft = fs * (0 : nfft/ 2) /
    nfft -> tiene una sample mas
% frec = (0 : length(dftModulada)-1)* 100/length(dftModulada); % frec
    = f *(0:(t./2))./t;
%fn = 0 : (length(dftModulada) - 1) * 100 / length(dftModulada); %fn =
    0 : fs/n : fs - fs/n ;
% plot(frec,mModulada) ; title('Magnitud') ; grid on ;

demodulacion

yDemodulada = moduladora .* (portadora .^(2)) ;

nfftDem = length(yDemodulada) ; nfftDem = 2 ^ nextpow2(nfftDem) ;
dftDemodulada = fft(yDemodulada , nfftDem); % dft de la senal
    demodulada

mDemodulada = abs(dftDemodulada); mDemodulada =
    mDemodulada(1 :nfftDem / 2);
mDemoduladaNormal = mDemodulada / max(mDemodulada);

nFftDem = fs * (0 : ( nfftDem / 2 - 1 )) / nfftDem ;

% % filtrar la senal reconstruida / demodulada
% b = filtroLow();
% demodFiltrada = filter(b,1,yDemodulada) ;

% filtrar con detector de envolvente
[upEnv,loEnv] = envelope(yDemodulada) ;
% senalRecons =

figure('Name','Modulacion Am') ;
subplot(5,1,1); plot(t , moduladora) ; title(' Moduladora') ; grid on
    ; xlabel('tiempo (seg)') ;

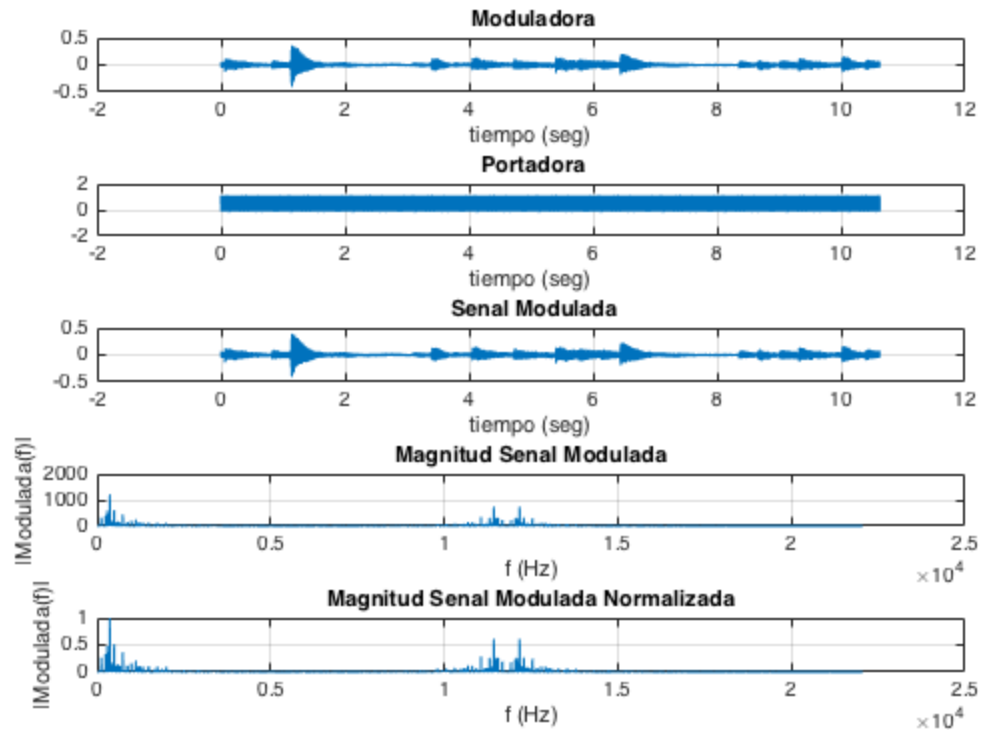
subplot(5,1,2); plot(t , portadora) ; title(' Portadora') ; grid on ;
    xlabel('tiempo (seg)');

subplot(5,1,3); plot(t , modulada ) ; title(' Senal Modulada') ;
    grid on ; xlabel('tiempo (seg)');

subplot(5,1,4); plot( nFft, mModulada) ; title('Magnitud Senal
    Modulada') ; grid on ; xlabel('f (Hz)') ; ylabel('|Modulada(f)|')

```

```
subplot(5,1,5); plot( nFft, mModuladaNormal) ; title('Magnitud Senal
Modulada Normalizada') ; grid on ; xlabel('f (Hz)') ; ylabel('|
Modulada(f)|')
```

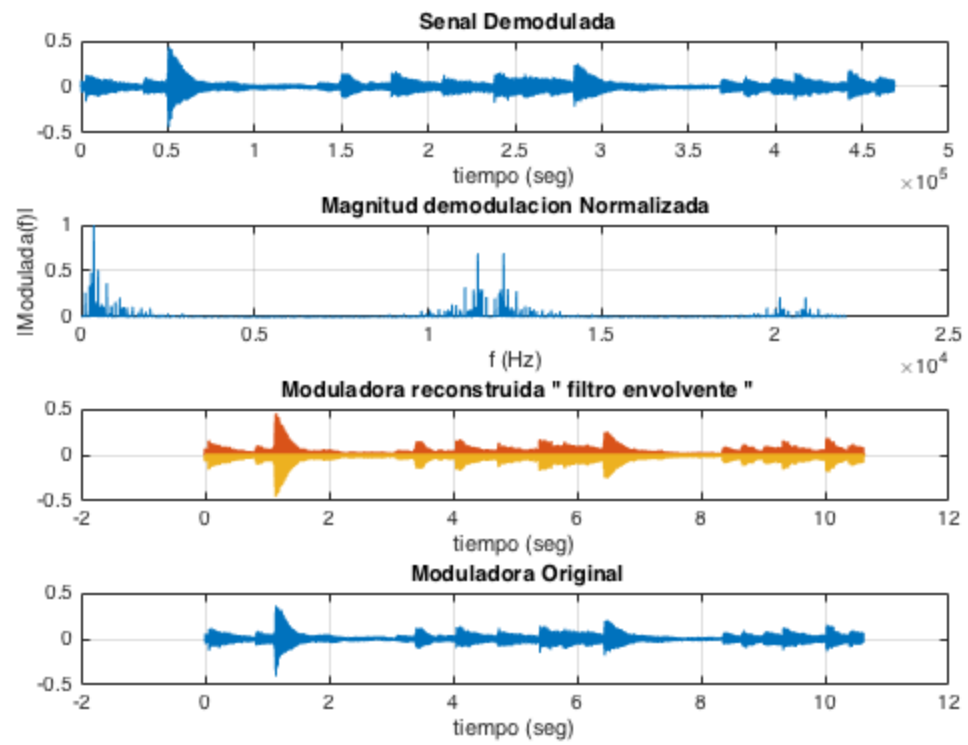


```
figure ('Name', 'Demodulacion DSB. sc') ;
subplot(4,1,1); plot(yDemodulada) ; title(' Senal Demodulada') ;
grid on ; xlabel('tiempo (seg)') ;

subplot(4,1,2); plot( nFftDem, mDemoduladaNormal) ; title('Magnitud
demodulacion Normalizada') ; grid on ; xlabel('f (Hz)') ; ylabel('|
Modulada(f)|') ;

subplot(4,1,3);plot(t,modulada) ; hold on ;
plot(t,upEnv,t,loEnv,'linewidth',1.5) ; title(' Moduladora
reconstruida " filtro envolvente "') ; grid on ; xlabel('tiempo
(seg)') ;

subplot(4,1,4); plot(t , moduladora) ; title(' Moduladora Original') ;
grid on ; xlabel('tiempo (seg)') ;
```



```
_Sonido_ sound(modulada,fs)

filename = 'senalModulada.wav'; audiowrite(filename,modulada,fs);
```

Published with MATLAB® R2015b