

**TRABAJO FINAL LABORATORIO**



# **ECUALIZADOR PARAMÉTRICO DE AUDIO**

**LUCÍA LARRAONA  
ALVARO HERNÁNDEZ**

**TDSÑ 2018-2019**

## **ÍNDICE**

<b>1. FUNCIÓN DISTORSIONADORA Y COMPENSADOR IDEAL.....</b>	<b>3</b>
<b>2. FILTRADO.....</b>	<b>5</b>
<b>3. SEÑAL DE PRUEBA.....</b>	<b>11</b>

## 1. FUNCIÓN DISTORSIONADORA Y COMPENSADOR IDEAL

Para poder comenzar el trabajo, se nos asigna una sala sobre la que tendremos que trabajar y diseñar el sistema ecualizador.

Como bien se cita en el enunciado, obtenemos los parámetros que definen la sala mediante la función *room*, como mostramos a continuación:

```
[Z, P, K] = room ( 'l.larraona@alumnos.upm.es' , 'alvaro.h.montes@alumnos.upm.es');
[B,A] = zp2tf (Z,P,K);
```

Dicha función genera 3 variables, Z, P y K, que no son más que los polos, los ceros, y la ganancia de la función de transferencia de la sala.

Obtenemos:

### Z

$0.5372 + 0.0088i$	Z(1)
$0.5372 - 0.0088i$	Z(2)
$0.6314 + 0.3645i$	Z(3)
$0.6314 - 0.3645i$	Z(4)
$-1.4093 + 0.0000i$	Z(5)

### P

$0.5067 + 0.1358i$	P(1)
$0.5067 - 0.1358i$	P(2)
$-0.3177 + 0.0000i$	P(3)
$0.0000 + 0.0000i$	P(4)
$0.0000 + 0.0000i$	P(5)

### K

0.8071

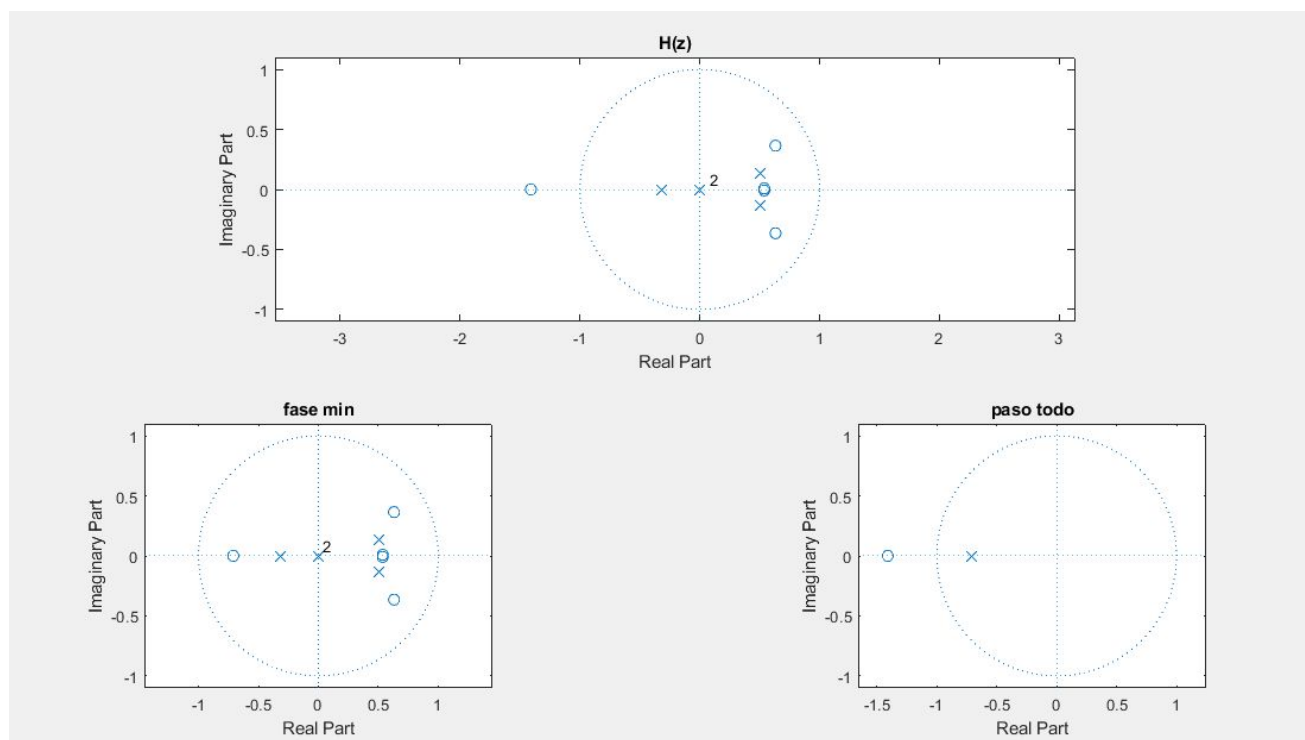
Haciendo uso de los conocimientos obtenidos en el tema 3 de la asignatura sabemos que para poder calcular el sistema compensador perfecto, debemos dibujar el diagrama de polos y ceros y calcular el sistema inverso al dado, dividiéndolo en uno de fase mínima y uno paso todo.

El valor Z(5) es el único que se encuentra fuera del círculo unidad. Es por ello que tendremos que introducir un polo y cero recíprocos para compensarlo, con valor  $1/Z(5)$ .

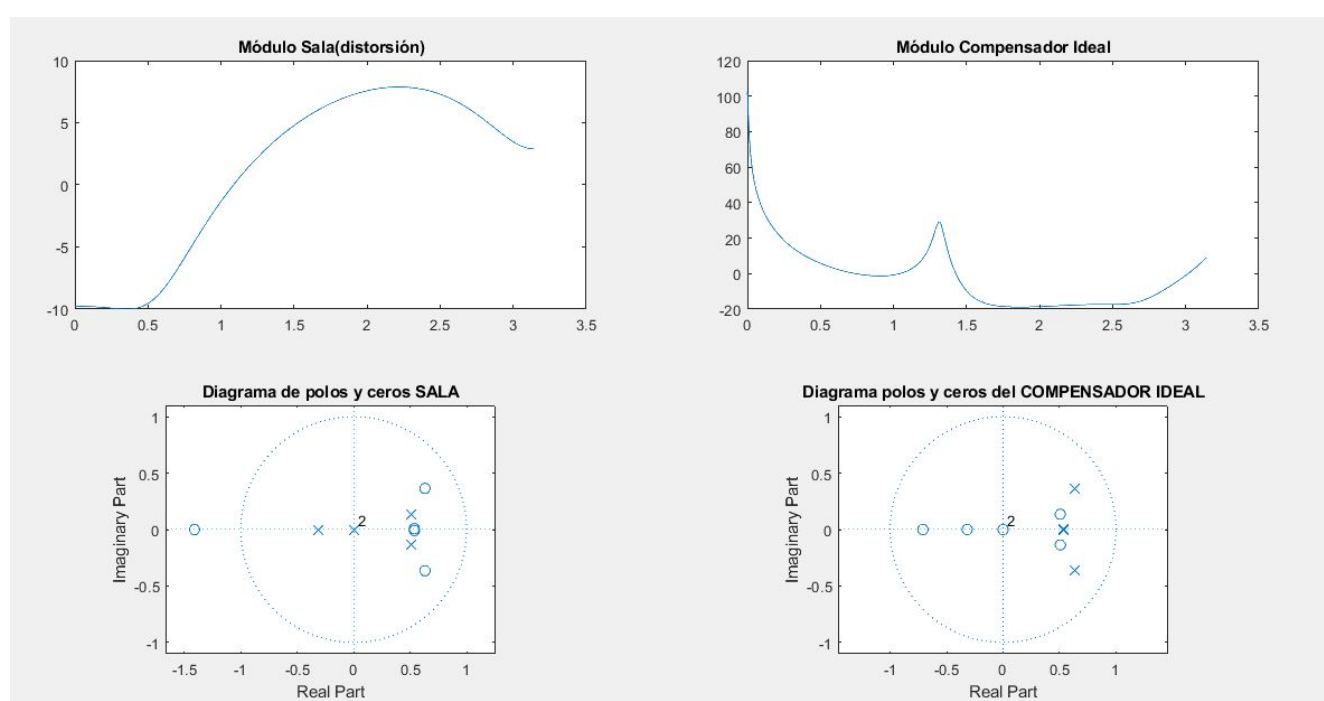
Los demás valores se encuentran todos dentro del círculo unidad por lo que no será necesario introducir más valores recíprocos adicionales.

El filtro de fase mínima estará formado por los polos y los ceros que queden dentro de la circunferencia unidad más el cero que añadimos para compensar (que inevitablemente queda también ubicado dentro)

Y el filtro paso todo está únicamente formado por el cero que se encontraba fuera y su polo recíproco, obteniendo así ganancia 1.



El sistema compensador ideal lo conseguimos calculando el inverso del filtro de fase mínima. Es decir,  $1/H_{fmin}$ , gráficamente:



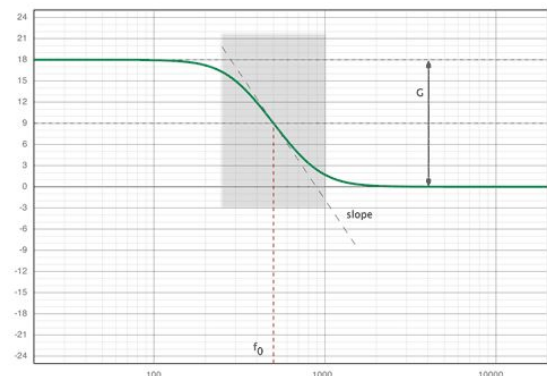
## 2. FILTRADO

El sistema de filtrado para compensar la distorsión que vamos a diseñar, está formado por 5 filtros en cascada.

¿Qué filtros son necesarios?

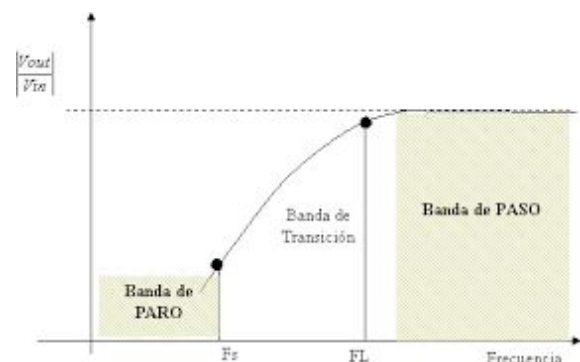
En este caso utilizamos tres tipos de filtros diferentes, cada uno con una función determinada. A continuación explicamos brevemente el funcionamiento de dichos filtros para mejor entendimiento del lector:

En primer lugar contamos con los filtros tipo *lowshelf* cuya función consiste en ajustar la ganancia de las frecuencias más bajas y el cual deja de tener efecto partir de su frecuencia de corte  $f_0$ .



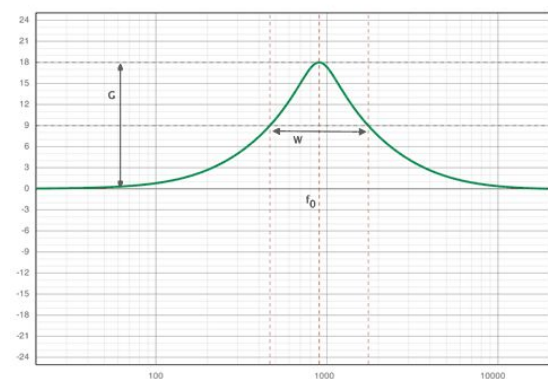
a

Por otro lado usamos los filtros tipo *highshelf*, que desempeñan la función contraria al anterior. Ajustan ganancia de las frecuencias más altas a partir de una frecuencia dada  $f_0$ .

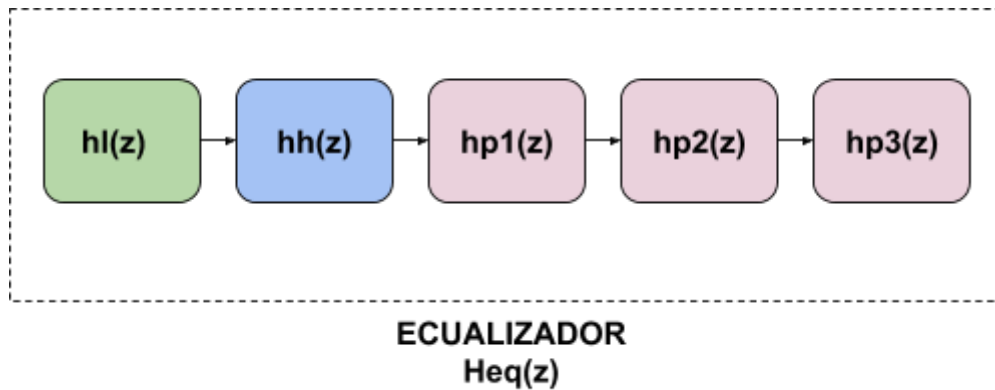


la

En último lugar hacemos uso de los filtros tipo *peak*, que funcionan atenuando las frecuencias próximas a una frecuencia central, en una región que se asemeja a una campana de gauss.



Para nuestra sala en concreto, hemos utilizado 1 filtro *lowshelf*, 1 filtro *highshelf* y 3 filtros tipo *peak*. El esquema de la interconexión en cascada lo mostramos a continuación:



Hemos definido todos los filtros mediante sus respectivas funciones de transferencia. Para ello ha sido necesario calcular los coeficientes correspondientes para cada tipo de filtro.

Dicho cálculo lo hemos realizado atendiendo a la referencia *AudioEQCookbook*. Hemos diseñado una función para cada tipo de filtro, dentro de las cuales calculamos dichos coeficientes.

#### • **FILTRO LOWSHELF**

```

function [LShelf,w,hnuml, hdenl, ml, mldB] = LowShelf(f0l, dBGainLS, QLS, Fs,
M)

% definimos parámetros previos
w0l= (2*pi*f0l)/Fs;
A1 = sqrt (10^(dBGainLS/20));
alphal = sin(w0l)/(2*QLS);

% coeficientes

b0l = A1*((A1+1) - (A1-1) * cos(w0l) + 2*sqrt(A1)*alphal);
b1l= 2 *A1 *((A1-1) - (A1+1) * cos(w0l));
b2l = A1 * ( (A1+1)-(A1-1)*cos(w0l) - 2*sqrt(A1)* alphal);

a0l = (A1 +1) + (A1-1)* cos(w0l) + 2*sqrt(A1)*alphal;
a1l = -2*((A1-1) + (A1+1) * cos(w0l));
a2l = (A1+1) + (A1-1)*cos(w0l) - 2*sqrt(A1)*alphal;

hnuml = [b0l,b1l,b2l];
hdenl= [a0l,a1l,a2l];

[LShelf,w] = freqz(hnuml,hdenl,M);

ml = abs(LShelf);
mldB = 20*log10(ml);
end
  
```

- **FILTRO HIGHSHELF**

```
function [HShelf,w,hnumh,hdenh,mh,mhdB] = HighShelf(f0h,dBGainHS,QHS,Fs,M)

% parametros previos:
w0h= (2*pi*f0h)/Fs;
Ah= sqrt(10^(dBGainHS/20));
alphah = sin(w0h)/ (2*QHS);

%coeficientes
b0h = Ah*( (Ah+1) + (Ah-1)*cos(w0h) + 2*sqrt(Ah)*alphah );
b1h = -2*Ah*( (Ah-1) + (Ah+1)*cos(w0h));
b2h = Ah*( (Ah+1) + (Ah-1)*cos(w0h) - 2*sqrt(Ah)*alphah );

a0h= (Ah+1) - (Ah-1)*cos(w0h) + 2*sqrt(Ah)*alphah;
a1h = 2*( (Ah-1) - (Ah+1)*cos(w0h));
a2h=(Ah+1) - (Ah-1)*cos(w0h) - 2*sqrt(Ah)*alphah;

hnumh = [b0h,b1h,b2h];
hdenh = [a0h,a1h,a2h];

[HShelf, w] = freqz( hnumh,hdenh,M);
mh=abs(HShelf);
mhdB= 20* log10(mh);

end
```

- **FILTRO PEAK**

```
function [p,w,hnump,hdenp,mp,mpdB] = peak(f0P1,dBgainP1,QP1,Fs,M)

%parametros previos
w0P1 = (2*pi*f0P1)/Fs;
Ap1 = sqrt(10^(dBgainP1/20));
alphap1 = sin(w0P1)/(2*QP1);

% coeficientes
b0p1 = 1 + alphap1*Ap1;
b1p1 = -2*cos(w0P1);
b2p1 = 1 - alphap1*Ap1;

a0p1 = 1 + alphap1/Ap1;
a1p1 = -2*cos(w0P1);
a2p1 = 1 - alphap1/Ap1;
```

```

hnum = [b0p1 b1p1 b2p1];
hden = [a0p1 a1p1 a2p1];

[p,w] = freqz (hnum,hden,M);
mp = abs(p);
mpdB = 20* log10 (mp);

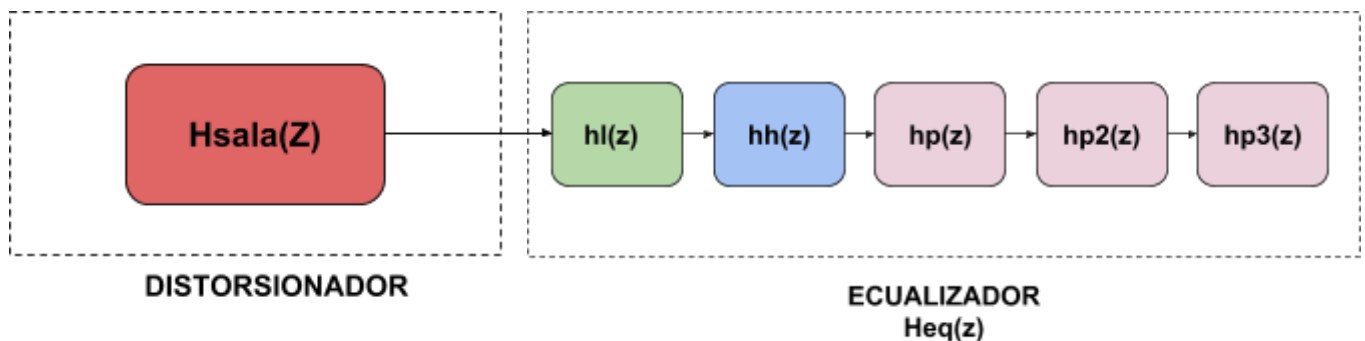
end

```

Todo este código tiene poca descripción ya que las fórmulas están basadas en el material proporcionado, los 3 últimos comandos de todos los filtros sirven para lo mismo, obtener la frecuencia mediante el comando `freqz` y obtener el módulo mediante el comando `abs`. En un último punto para trabajar en dB se hace  $20 \cdot \log$  (en base 10) del módulo.

Para llegar al compensador ideal es necesario conseguir que la suma, en decibelios, de la sala y el sistema ecualizador esté **cercana a cero**, o lo que es lo mismo, hacer que este compensador se ajuste lo más posible al ideal calculado en el primer apartado.

El esquema queda así:



Para conseguirlo, existen ciertos parámetros de los filtros que podemos modificar siempre que se ajusten a la serie de  $\frac{1}{3}$  de octava de la norma ISO 266.

En los filtros tipo shelving modificaremos tanto la frecuencia de corte como la ganancia y el ancho de banda. Para los filtros tipo peak realizaremos cambios análogos teniendo en cuenta que la frecuencia que modificamos es la frecuencia central.

Visto que no existe una forma analítica de ajustar estos parámetros, para conseguir la ecualización correcta, hemos seguido un proceso de prueba y error.

Aun así, la manera de ajustarlo sin dar demasiados pasos en falso, tiene una base teórica.

Los datos que se ajustan son los tres mencionados previamente. Ahora explicaremos más en detalle cómo equilibrar cada uno:



- **$f$ :** La frecuencia a la que actúa el filtro, cada filtro lo hemos situado en una posición diferente para conseguir actuar sobre los puntos necesarios, aquellos que tuvieran la ganancia lejana a cero.  
Teniendo la representación gráfica es muy sencillo, pues simplemente se trata de identificar las zonas que deben de ser modificadas e introducir un parámetro  $f$  acorde con la sensibilidad del filtro.
- **$dBGain$ :** Equivale a la ganancia que introduce el filtro a una determinada frecuencia, la cual debe situarse entre los -15, y 15 dB siendo los últimos los valores más altos.  
Viendo la representación gráfica, si la ganancia a una determinada frecuencia se alejaba del valor cero, escoges aproximadamente la ganancia que sería capaz de compensarla, para conseguir obtener una línea recta de valor constante en  $dB=0$ .
- **$Q$ :** Este ultimo parámetro define el área de actuación del filtro, de esta forma una variación de  $Q$  hace que el efecto en dB para una frecuencia  $f$ , aplique a más o menos valores de frecuencia en sus regiones cercanas.  
En la gráfica tenias que compatibilizar este valor para los extremos a los que las frecuencias no llegaban, para que de esta manera el filtro aplicase su efecto también en esos puntos.

Tras numerosos intentos, el resultado de la programación con los datos finales ha sido el siguiente:

```
% filtro LOWSHELF
f0l = 6300; %frecuencia de corte
dBGainLS= 10; % ganancia
QLS=0.6;
[LSshelf,w,hnuml, hdenl, ml, mldB] = LowShelf(f0l, dBGainLS, QLS, Fs, M);

% filtro HIGHSHELF
f0h= 20000;
dBGainHS = -4;
QHS = 0.5;
[HSshelf,w,hnumh,hdenh,mh,mhdB] = HighShelf(f0h,dBGainHS,QHS,Fs,M);

% filtro tipo PEAK (1)
f0P1 =8000 ;
dBgainP1 = 4.5;
QP1 = 0.5;
[p,w,hnump,hdenp,mp,mpdB] = peak(f0P1,dBgainP1,QP1,Fs,M);
```

```

% filtro tipo PEAK (2)
f0P2 = 10000;
dBgainP2 = -3;
QP2 = 0.5;
[p2,w,hnump2,hdenp2,mp2,mpdB2] = peak(f0P2,dBgainP2,QP2,Fs,M);

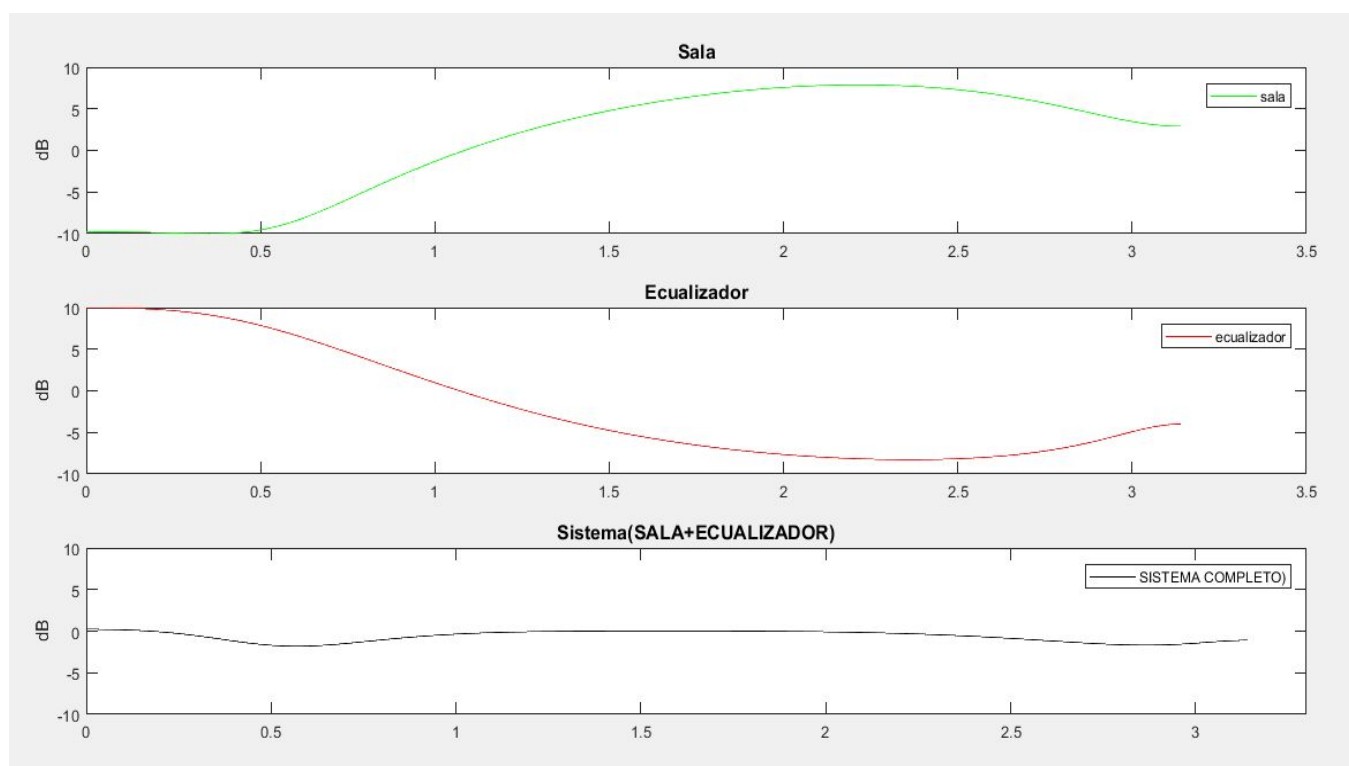
% filtro tipo PEAK (3)
f0P3 = 16000;
dBgainP3 = -7.5;
QP3= 0.3;
[p3,w,hnump3,hdenp3,mp3,mpdB3] = peak(f0P3,dBgainP3,QP3,Fs,M);

```

De forma más esquemática:

	Paso Bajo	Paso Alto	Peak 1	Peak 2	Peak 3
Frecuencia	6300	20000	8000	10000	16000
Ganancia	10	-4	4.5	-3	-7.5
Q	0.6	0.5	0.5	0.5	0.3

El resultado obtenido se ajusta lo más posible al teórico dentro de una cota de  $[-1.5, 0.5]$ .  
Lo mostramos a continuación:



### 3. SEÑAL DE PRUEBA

Procedemos a comprobar que el ecualizador funciona correctamente utilizando la señal de prueba.

Para poder utilizar el fichero de prueba que se nos proporciona es necesario remuestrear la señal a la frecuencia de muestreo del ecualizador (48kHz), ya que dicho fichero está a 44.1 kHz.

Para ello hacemos uso de la función *resample* pero usando la sintaxis siguiente:

```
y = resample(x,p,q,n,beta)
```

No podemos usar *resample* sin esta sintaxis puesto que, esta función, de forma predeterminada, realiza el cambio de la frecuencia de muestreo haciendo uso de un filtro a la hora de interpolar que introduce una atenuación de aproximadamente 60 dB y que no es suficiente para este proyecto.

Utilizamos entonces el parámetro *beta*, para poder ajustar la ganancia del filtro interno y conseguir que se encuentre en un rango de entre 96 y 100 dB.

Para completar este procedimiento utilizamos la función *rat* que genera dos vectores N y D tales que N/D se aproxima al cambio fraccionario que necesitamos para remuestrear la señal.

El parámetro n debe ser de entorno 10.

```
% Remuestreo de la señal
% utilizamos rat para calcular los parámetros p y q de la función
resample

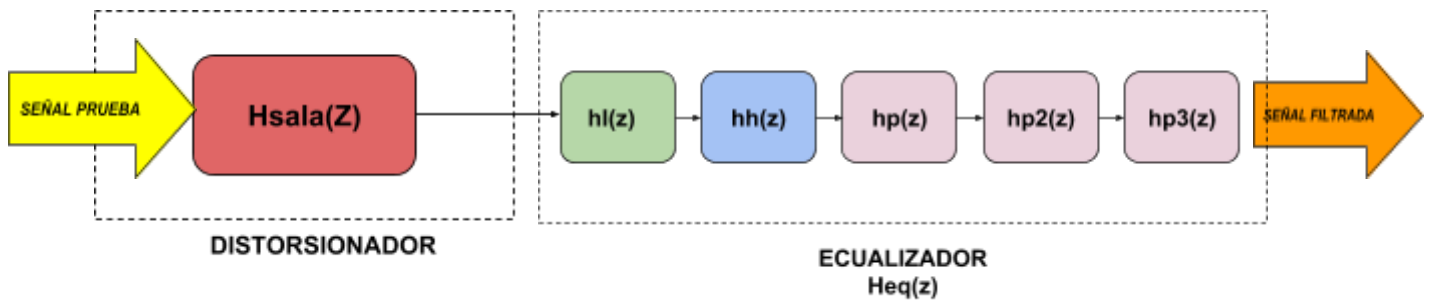
[n,d]=rat (fsenal/48);
p=d;
q=n;
%el parámetro n nos dicen que debe ser 10

%hemos escogido beta(10) tal que el filtro tenga una atenuación entre
%[96,100]dB

%finalmente, señal de prueba convertida a la frec de muestreo del
ecualizador

y = resample(x(:,1),p,q,10,10);
```

Introducimos en el esquema previo una representación del paso de la señal de prueba:



Mediante la función *filter* simulamos las señales de salida que conseguiríamos al hacer que la señal de prueba atravesara cada uno de los filtros.

Adjuntamos el extracto del código en el que realizamos la conexión en cascada:

```
Y= filter(B,A,y);           % a traves de la sala (distorsion)

Y1 = filter(hnumh,hdenh,Y);   %a traves del filtro paso alto

Y2 = filter(hnuml,hdenl,Y1);  %a traves del paso bajo

Y3 = filter(hnump,hdenp,Y2);  %a traves del 1 filtro peak

Y4 = filter(hnump2,hdenp2,Y3); %a traves del 2 filtro peak

Y5 = filter(hnump3,hdenp3,Y4); %a traves del 3 filtro peak
```

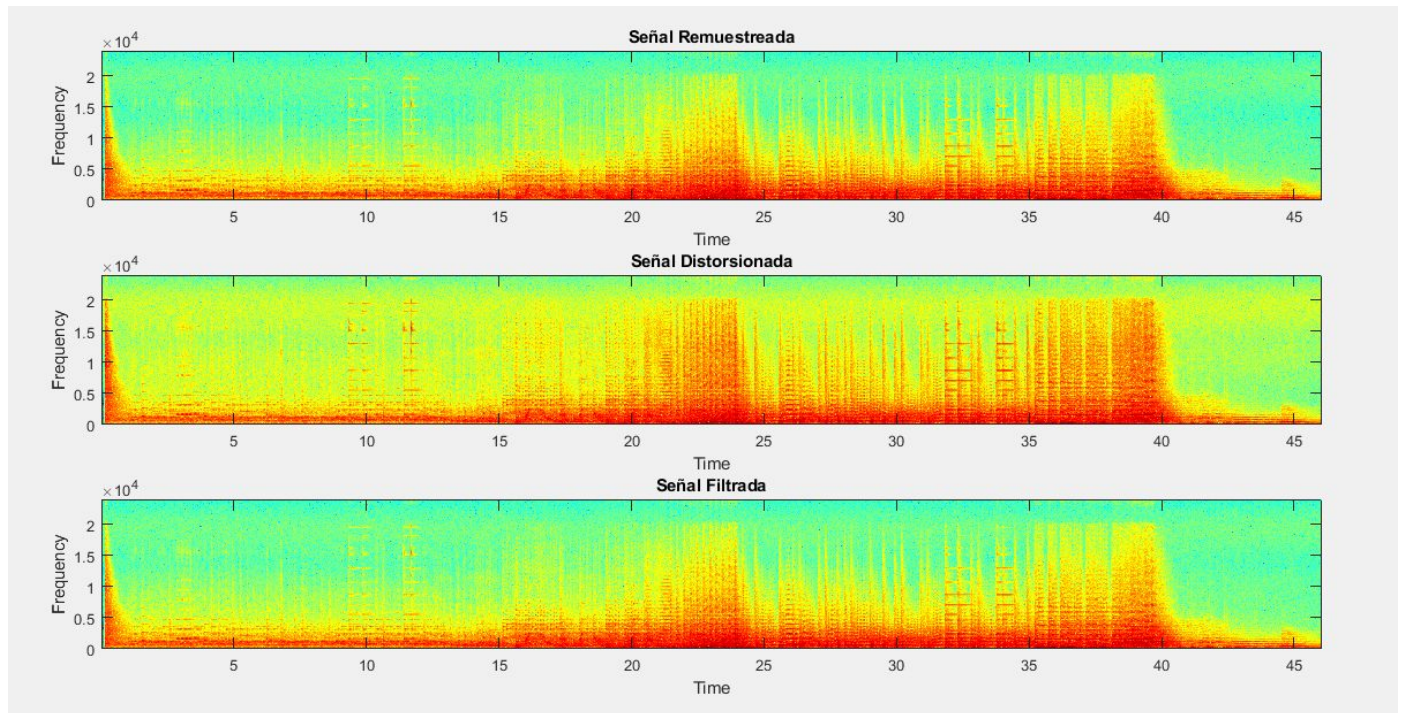
La señal 'y' correspondería con la señal de prueba remuestreada que es la que introducimos al inicio del esquema.

La señal 'Y' es la señal distorsionada al atravesar la sala.

La señal 'Y5' es la señal final, una vez que atraviesa todos los filtros.

Y las demás variables Yi son los resultados parciales que tiene la señal de entrada al estar atravesando los filtros del ecualizador.

La representación de dichas señales las hemos realizado con la función specgram:



**Referencias:**

<http://www.musicdsp.org/files/Audio-EQ-Cookbook.txt>

[https://en.wikipedia.org/wiki/Equalization\\_\(audio\)](https://en.wikipedia.org/wiki/Equalization_(audio))

<http://www.disca.upv.es/adomenec/IASPA/tema4/ES-Filtros.html>

[https://es.mathworks.com/help/matlab/learn\\_matlab/help.html](https://es.mathworks.com/help/matlab/learn_matlab/help.html)