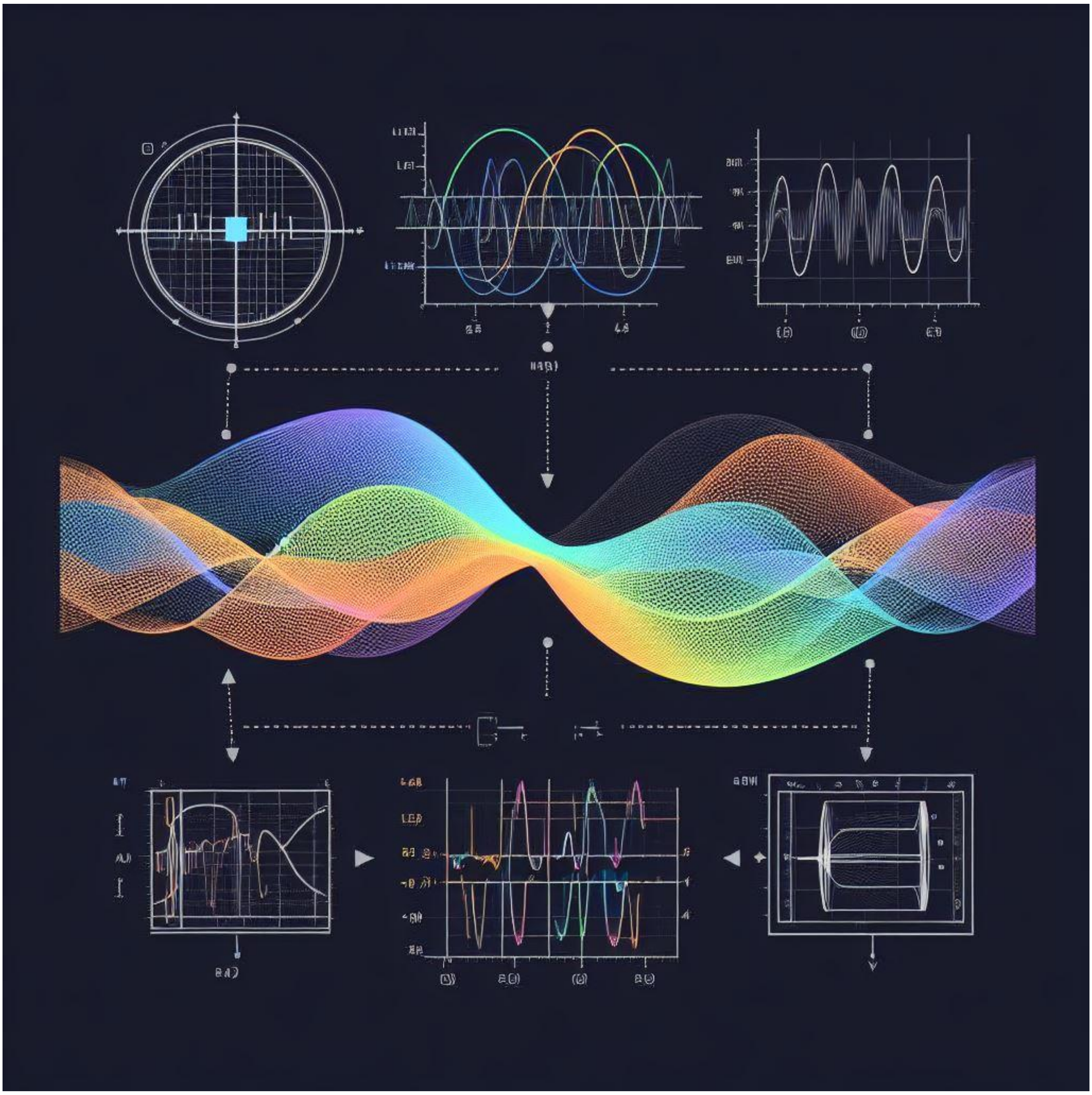
Introducción al Procesamiento de Señales

Práctica con Utilitario 1

Santiago Nihany - 03012/3

Ignacio Faccipieri - 02502/3



Ejercicio 1

Lista de comandos

- ➔ *clear all; close all; clc;*: Se utilizan para limpiar el espacio de trabajo actual. *clear all* elimina todas las variables del espacio de trabajo, *close all* cierra todas las ventanas de gráficos abiertas y *clc* limpia la ventana de comandos.

- ➔ *fileparts(mfilename('fullpath'));*: Esta línea obtiene el directorio en el que se encuentra el script actual.

- ➔ *addpath();*: Agrega la ruta al directorio que contiene las funciones relacionadas con el ejercicio 1.

- ➔ *[n, x] = senial(numAl);*: Llama a la función *senial* (brindada por la cátedra) con el valor *numAl* (variable donde se almacena el número de alumno a utilizar) como argumento y asigna los resultados a las variables *n* y *x*. *n* es un vector que representa los instantes de tiempo n y *x* un vector con los valores de amplitud en esos respectivos instantes. Deben tener la misma longitud, ya que sus valores se corresponden directamente.

- ➔ *stem(n, x);*: Dibuja un gráfico de la señal en el dominio del tiempo, donde *n* representa los instantes y *x* representa las amplitudes.

- ➔ *X = TFTD(n, x)*: Calcula la Transformada de Fourier en Tiempo Discreto (TFTD) de la señal *x* y almacena el resultado en la variable *X*.

- ➔ *ds = 0.001; s = [-2:ds:2]*: Define una variable *ds* con un valor de 0.001 y crea un vector *s* que varía de -2 a 2 con un paso de *ds*. La variable *s* es utilizada como parámetro de la función *plot()* para graficar la TFTD de *x*.

➔ *plot(s, X)*: Dibuja un gráfico de la transformada de la señal en el dominio de la frecuencia utilizando la función *plot*.

➔ *"title(' ')", "xlabel(' ')", "ylabel(' ')", y "grid on":* Comandos utilizados para el etiquetado del gráfico: agregarle un título, etiquetas a los ejes y una cuadrícula.

Funciones utilizadas

Para la resolución del ejercicio 1 necesitamos representar los sistemas que se nos brindan. Esto fue resuelto haciendo uso de las siguientes funciones:

- **Sist1, Sist2:**
    - Funciones utilizadas para lograr representar los distintos sistemas dados:

$$\boxed{\begin{aligned} y[n] &= \frac{1}{2}x[n] + \frac{1}{2}x[n-1] \\ y[n] &= \frac{1}{2}x[n] - \frac{1}{2}x[n-1] \end{aligned}}$$

En cada función se recorren todos los instantes "**n**" brindados y se evalúa **x[n]** en dicho instante para que luego el vector **y** que va a devolver la función**,** tome cierto valor en el mismo instante dependiendo de cuánto valga x[n].

- **Sist3, Sist4:**

    - Para los siguientes sistemas dados

$$\boxed{\begin{aligned} y[n] &= \frac{1}{4}x[n] + \frac{1}{4}x[n-1] + \frac{1}{2}y[n-1] \\ y[n] &= \frac{1}{4}x[n] - \frac{1}{4}x[n-1] - \frac{1}{2}y[n-1] \end{aligned}}$$

Se utiliza la misma lógica que se utilizó en las funciones **Sist1** y **Sist2** con la diferencia de que no sólo se evalúa **x[n]** sino que también se evalúa la misma función **y[n]** en un instante anterior. Esto quiere decir que la nueva función y[n] no solo depende del valor de x[n], sino también depende de cuánto valía y[n] en el instante anterior.

- **Transformadas de Fourier de Tiempo Discreto (TFTD)**

Otra función utilizada para lograr la resolución del ejercicio 1, fue la **Transformada de Fourier de Tiempo Discreto** (TFTD). Utilizamos la explicación otorgada en el Trabajo Práctico Nº 5 sobre la aproximación de la TFTD de la función cajon:

$$X(e^{j2\pi s}) = \sum_{n=-\infty}^{\infty} x[n]\; e^{-j2\pi sn}$$

Sin embargo, en una implementación numérica, la sumatoria no puede realizarse de $-\infty$ a ∞, sino de un determinado valor N_1 a N_2. Si se toma $N_1 = -K$, $N_2 = K$, y se define $M = 2K+1$, la "aproximación" de la TFTD resulta:

$$\hat{X}(e^{j2\pi s}) \approx \sum_{n=-K}^{K} x[n]\; e^{-j2\pi sn} = \sum_{n=-\infty}^{\infty} \sqcap_M[n]x[n]\; e^{-j2\pi sn}$$

Es decir, $\hat{X}(e^{j2\pi s})$ es la TFTD de la señal $\sqcap_M[n]x[n]$.

Cuando la señal $x[n]$ es de soporte finito, como en el caso de un cajón discreto, eligiendo adecuadamente el valor de K, se tiene que $\hat{X}(e^{j2\pi s}) = X(e^{j2\pi s})$.

Teniendo en cuenta esto, pudimos obtener una función generalizada de la expresión de la TFTD y codificarla en MATLAB.

```
function X= TFTD(n,x)

   ds = 0.001; s = [-2:ds:2]; X = zeros(size(s));
   for k = 1:length(s)
      X(k)=sum(x.*exp(-1i*2*pi*s(k)*n));
   end
end
```

En esta función se calcula la TFTD de una función mediante la definición de la Transformada de Fourier de Tiempo Discreto, evaluando los términos en distintos instantes k (instantes dados por un vector “s” el cual comienza en -2 y finaliza en 2 con un paso de 0.001).

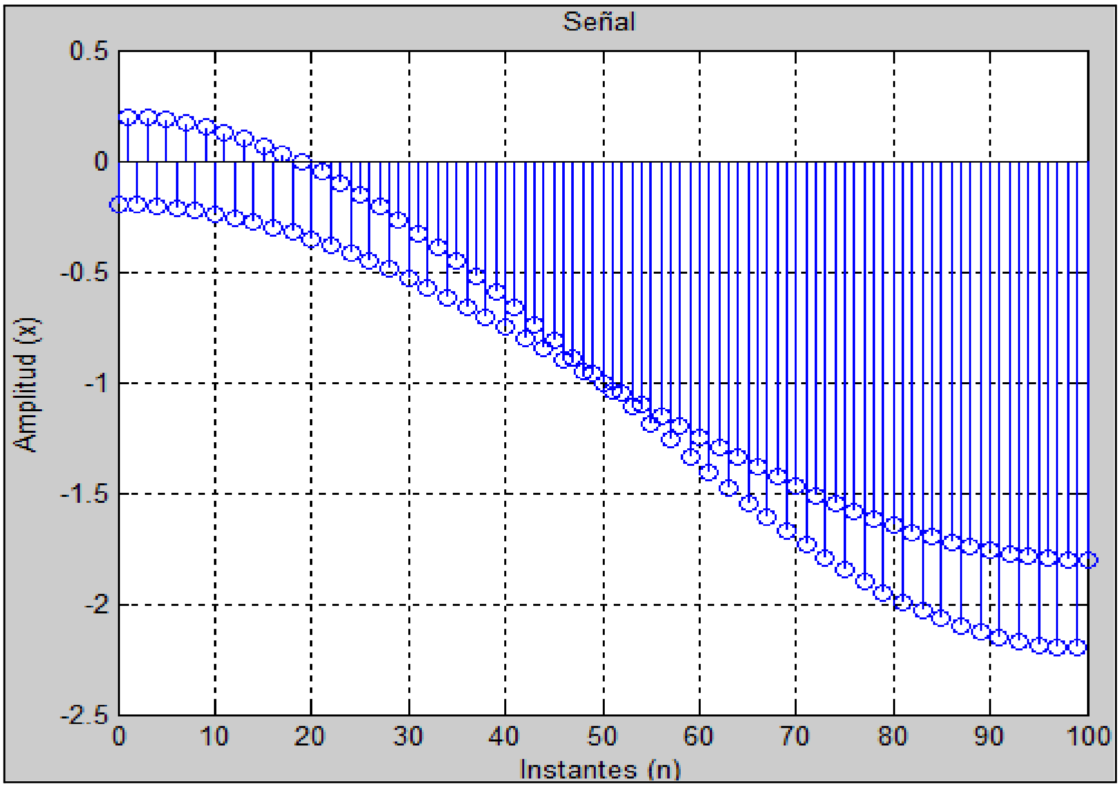
<u>Inciso 1</u>

- **senial.m**

Esta función se encuentra dentro de la consigna del trabajo, y con ella, se pide realizar: su gráfico, su TFTD y el gráfico de su transformada.

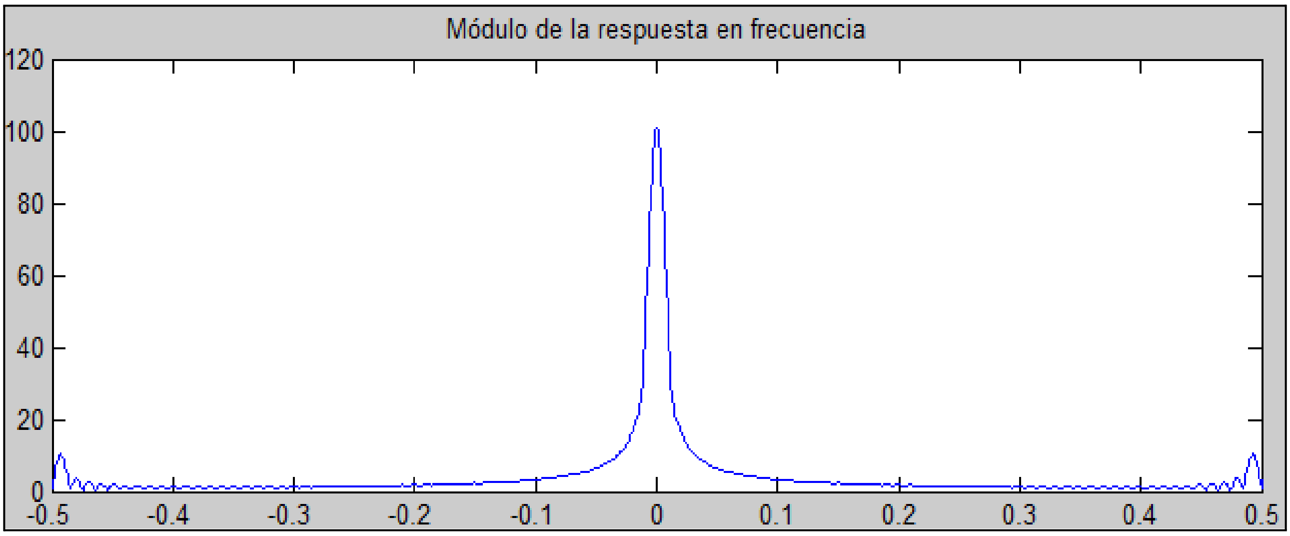
Esta función abarca 4 señales únicas pero sólo retorna 1. Para determinar cuál de las 4 señales debe retornar, se emplea la función *rem(número, 4)*. Esta función divide el número proporcionado por 4 y en función del resto de dicha división elije la señal. En este caso, el número que se debe elegir como entrada es un <u>número de alumno</u>: se utilizó el de Santiago (**030123**). El resto de dicha división da como resultado *3*, por lo tanto será seleccionada la señal *3* como salida de la función.

Para realizar el gráfico de dicha función se utilizó la función *"stem"*, ya descrita anteriormente. Este fue el resultado:

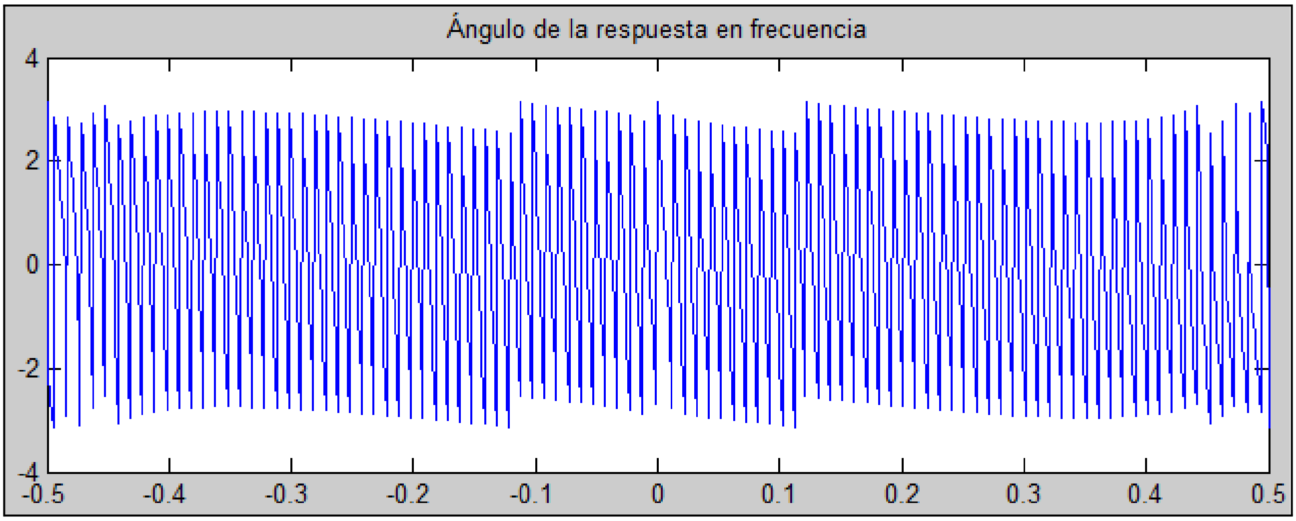


Se hizo uso de la función previamente mencionada y explicada, llamada *"TFTD"* para calcular la **Transformada de Fourier de Tiempo Discreto** de la función "*senial*". Una vez hecho esto, se la graficó utilizando la función "*plot*" y el resultado fue el siguiente:

Módulo de la respuesta en frecuencia de la señal



Fase de la respuesta en frecuencia de la señal



Para identificar los componentes de frecuencia de la señal, debemos analizar un período de la misma. Tras observar el gráfico podemos ver que la señal tiene período $T = 1$ y frecuencia fundamental $s = 1/T = 1$. Tomamos un período de la señal comprendido entre $-s/2$ y $s/2$: [-½ , ½]. Se pueden **identificar componentes de frecuencias bajas** (sus picos están cercanos a 0) como se ve en la imagen:

<u>Inciso 2</u>

<u>Sistema 1</u>

$$\boxed{y[n] = \frac{1}{2}x[n] + \frac{1}{2}x[n-1]}$$

Respuesta impulsional:

La respuesta impulsional describe la respuesta del sistema cuando se le aplica un impulso (delta) como entrada. En términos prácticos, para obtener la respuesta impulsional de un sistema, y[n] se reemplaza por h[n] y se le aplica una delta (x[n] pasa a ser una delta centrada en el mismo punto donde estaba la x original).

$$\boxed{h[n] = \frac{1}{2}\delta[n] + \frac{1}{2}\delta[n-1]}$$

Respuesta en frecuencia:

Podemos definir a la respuesta en frecuencia del sistema como la TFTD de la respuesta impulsional "*h*" calculada anteriormente.

A partir de la respuesta impulsional encontrada, se aplicaron las siguientes propiedades:

<u>Desplazamiento en tiempo y en frecuencia:</u> $\boxed{x[n-n_0] \xleftrightarrow{\text{TFTD}} e^{-j2\pi s n_0} X(e^{j2\pi s})}$

el <u>par transformado</u> de la delta de Kronecker $\boxed{\delta[n] \supset 1}$

y se obtuvo la respuesta en frecuencia del sistema:

$$H(e^{j2\pi s}) = \frac{1}{2} + \frac{1}{2} \cdot e^{-j2\pi s}$$

Aplicando la misma lógica y utilizando las mismas propiedades, se obtuvo para el **sistema 2:**

$$y[n] = \frac{1}{2}x[n] - \frac{1}{2}x[n-1]$$

Su **respuesta impulsional:**

$$h[n] = \frac{1}{2}\delta[n] - \frac{1}{2}\delta[n-1]$$

Y su **respuesta en frecuencia:**

$$H(e^{j2\pi s}) = \frac{1}{2} - \frac{1}{2} e^{-j2\pi s}$$

Ambos sistemas, 1 y 2, son sistemas **FIR** (Finite Impulse Response). Esto quiere decir que su respuesta impulsional tiene un número finito de términos no nulos. Podemos identificarlo observando que sus ecuaciones no contienen términos que involucren salidas en instantes anteriores.

Sistema 3

$$y[n] = \frac{1}{4}x[n] + \frac{1}{4}x[n-1] + \frac{1}{2}y[n-1]$$

Respuesta impulsional:

Para calcular la respuesta impulsional del sistema 3 y 4, se aplica la misma lógica que para los sistemas 1 y 2 (ya explicados): y[n] se reemplaza por h[n] y se le aplica una delta obteniendo así la siguiente expresión:

$$h[n] = \frac{1}{4}\delta[n] + \frac{1}{4}\delta[n-1] + u[n-1]\,\frac{3}{8}\left(\frac{1}{2}\right)^{n-1}$$

Para explicar el término final de la respuesta impulsional hallada, se adjunta una imagen donde se muestra el procedimiento realizado para llegar a dicha expresión: Se evalúa h[n] en distintos instantes hasta encontrar un patrón. Se ve que a partir de n=1 comienza a multiplicarse ⅜ por ½ elevado a (n-1), sin embargo es necesario multiplicar dicho patrón por un escalón que comience en el instante en el que comienza el patrón (n=1).

$$h[n] = \frac{1}{4}\delta[n] + \frac{1}{4}\delta[n-1] + \frac{1}{2}h[n-1]$$

$$h[0] = \frac{1}{4} \qquad h[1] = \frac{1}{4} + \frac{1}{2}\left(\frac{1}{4}\right) = \frac{3}{8}$$

$$h[2] = \frac{1}{2}\left(\frac{3}{8}\right) \qquad h[3] = \left(\frac{1}{2}\right)^2 \cdot \frac{3}{8} \qquad h[4] = \left(\frac{1}{2}\right)^3 \cdot \frac{3}{8}$$

Respuesta en frecuencia:

$$Y(e^{j2\pi s}) - \frac{1}{2}Y(e^{j2\pi s}).e^{-j2\pi s} = \frac{1}{4}X(e^{j2\pi s}) + \frac{1}{4}X(e^{j2\pi s}).e^{-j2\pi s}$$

$$\frac{Y(e^{j2\pi s})}{X(e^{j2\pi s})} = H(e^{+j2\pi s}) = \frac{1/4 + 1/4\, e^{-j2\pi s}}{1 - 1/2\, e^{-j2\pi s}}$$

Sistema 4

$$y[n] = \frac{1}{4}x[n] - \frac{1}{4}x[n-1] - \frac{1}{2}y[n-1]$$

Respuesta impulsional:

$$h[n] = \frac{1}{4}\delta[n] - \frac{1}{4}\delta[n-1] - \frac{3}{8}\left(-\frac{1}{2}\right)^{n-1} \cdot u[n-1]$$

Explicación de cómo se llegó al último término de la respuesta impulsional hallada:

$$h[n] = \frac{1}{4}\delta[n] - \frac{1}{4}\delta[n-1] - \frac{1}{2}h[n-1]$$

Evaluando h[n] en distintos instantes:

$$h[0] = \frac{1}{4} \qquad h[1] = -\frac{1}{4} - \frac{1}{2}\left(\frac{1}{4}\right) = -\frac{3}{8}$$

$$h[2] = -\frac{1}{2}\cdot\left(-\frac{3}{8}\right) \qquad h[3] = \left(-\frac{1}{2}\right)^{n-1}\left(-\frac{3}{8}\right)$$

Respuesta en frecuencia:

La respuesta en frecuencia del sistema 4 es idéntica a la respuesta en frecuencia del sistema 3, sólo cambian algunos signos a la hora del despeje. Se adjunta una imagen con la respuesta final hallada:

$$H\left(e^{j2\pi s}\right) = \frac{1/4 - 1/4\, e^{-j2\pi s}}{1 + 1/2\, e^{-j2\pi s}}$$

Ambos sistemas, 3 y 4, son sistemas **IIR** (Infinite Impulse Response). Esto quiere decir que su respuesta impulsional tiene un número infinito de términos no nulos. Podemos identificar esto observando que sus ecuaciones contienen términos que involucran salidas en instantes anteriores, como *y[n - 1]*.

Inciso 3

Sistema 1

Para representar el sistema 1 en MATLAB realizamos la siguiente función:

```
function y = sist1(n,x)
    y = zeros(size(n));
    for i = 1:length(n)
        if i == 1
            % Si es el primer elemento, x[n-1] se asume como 0
            y(i) = 1/2 * x(i);
        else
            y(i) = 1/2 * x(i) + 1/2 * x(i - 1);
        end
    end
end
```

Entradas:

- *n*: vector de instantes de la función.
- *x*: vector con valores de la función en los instantes *n*.

Salida:

- *y*: vector de la expresión salida de tamaño *n*.

En primer lugar, crea un vector de ceros de tamaño n y se lo asigna a la variable de salida *y*. Para completar la expresión de salida, realiza un *for* de "*n*" iteraciones y opera con los valores que de entrada de *x*.

Los sistemas posteriores están codificados de la misma manera pero cumpliendo de acuerdo a sus respectivas ecuaciones.

Respuesta **impulsional** obtenida: $h[n] = \frac{1}{2}\delta[n] + \frac{1}{2}\delta[n-1]$

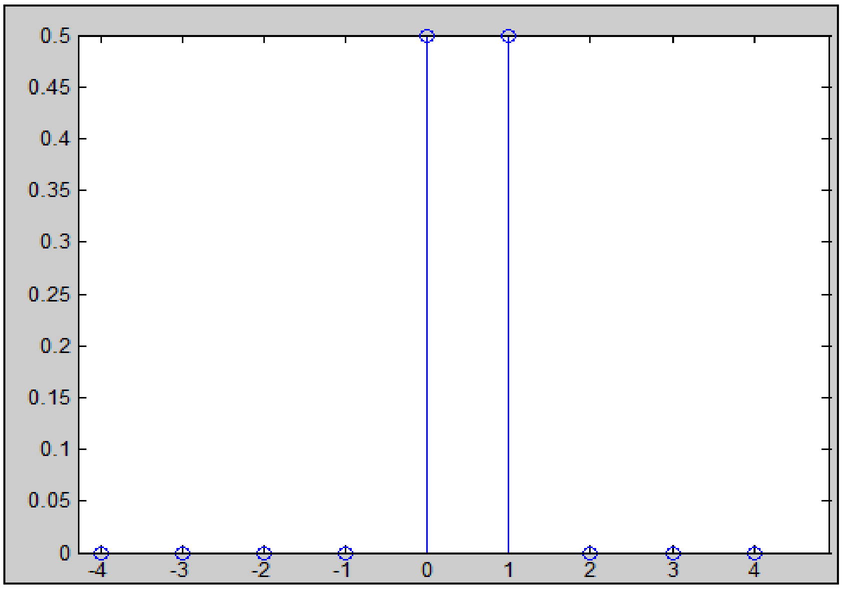
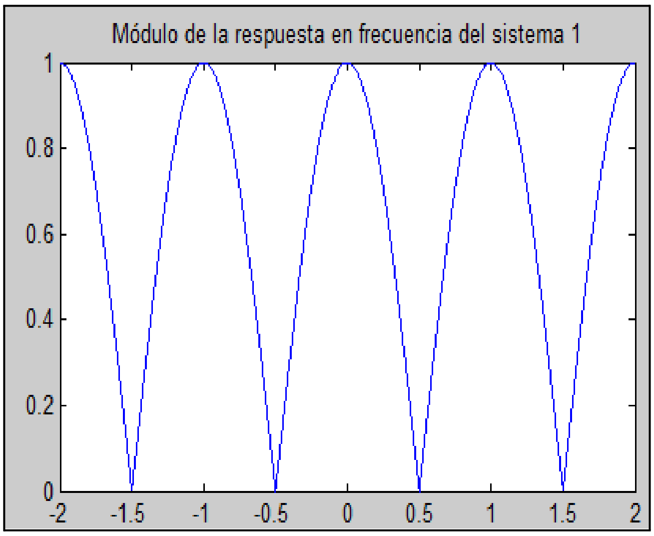
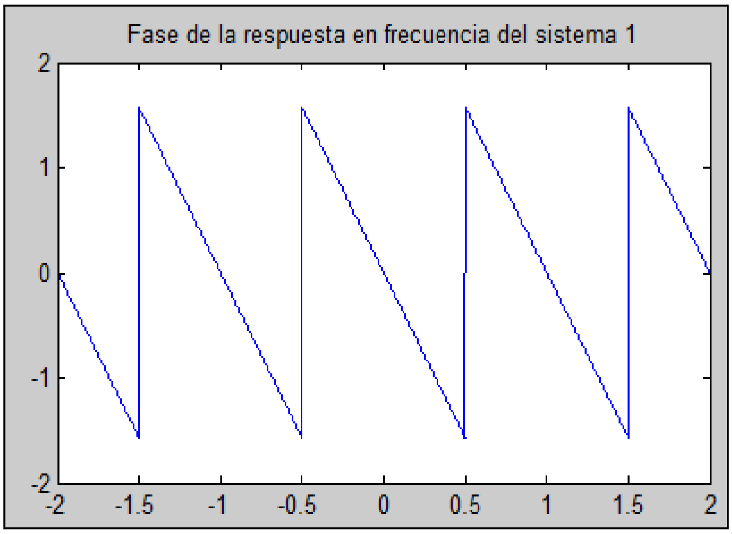


Gráfico de la respuesta impulsional obtenida

Respuesta en **frecuencia** obtenida: $H(e^{j2\pi s}) = \frac{1}{2} + \frac{1}{2} \cdot e^{-j2\pi s}$





Sistema 2

Para representar el sistema 2 en MATLAB realizamos la siguiente función:

```
function y = sist2(n,x)
    y = zeros(size(n));
    for i = 1:length(n)
        if i == 1
            % Si es el primer elemento, x[n-1] se asume como 0
            y(i) = 1/2 * x(i);
        else
            y(i) = 1/2 * x(i) - 1/2 * x(i - 1);
        end
    end
end
```

Respuesta **impulsional** obtenida: $h[n] = \frac{1}{2}\delta[n] - \frac{1}{2}\delta[n-1]$

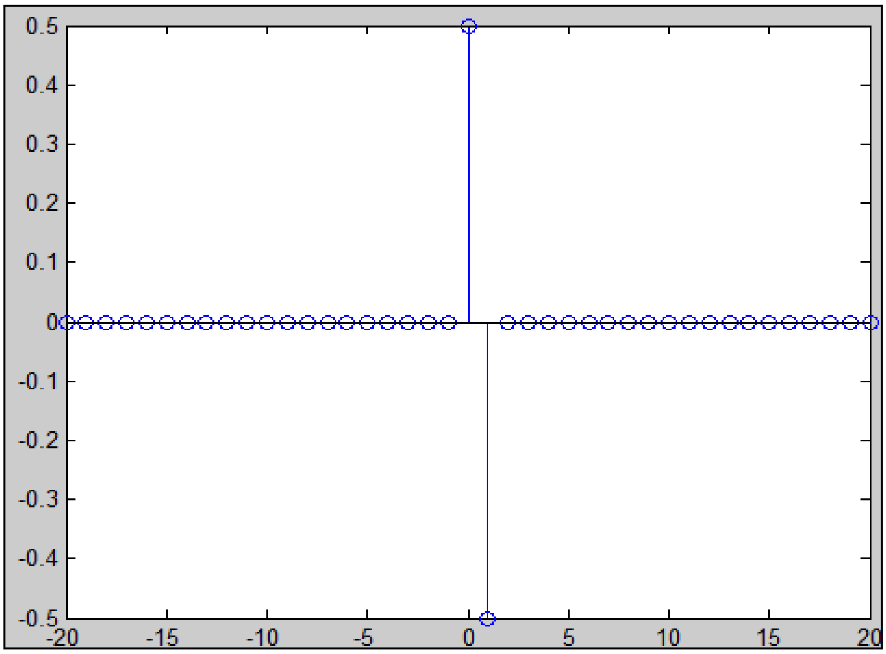
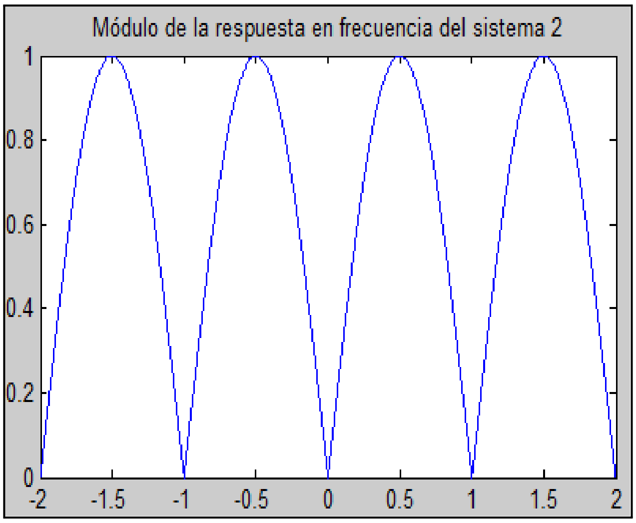
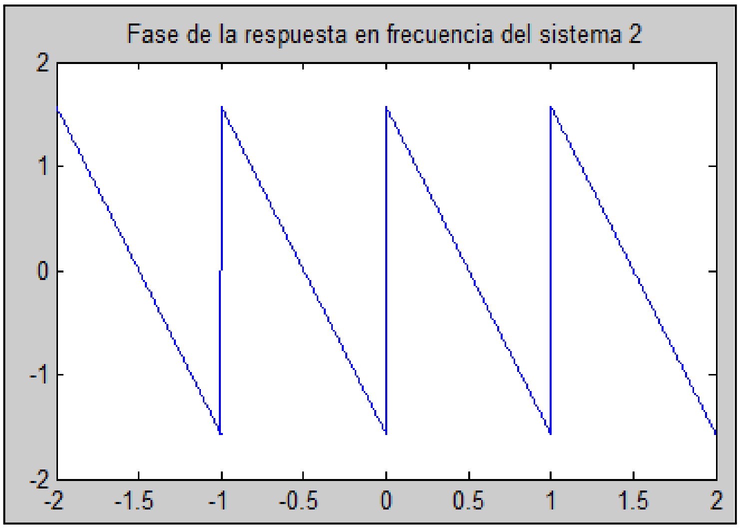


Gráfico de la respuesta impulsional obtenida

Respuesta en **frecuencia** obtenida: $H(e^{j2\pi s}) = \frac{1}{2} - \frac{1}{2}e^{-j2\pi s}$





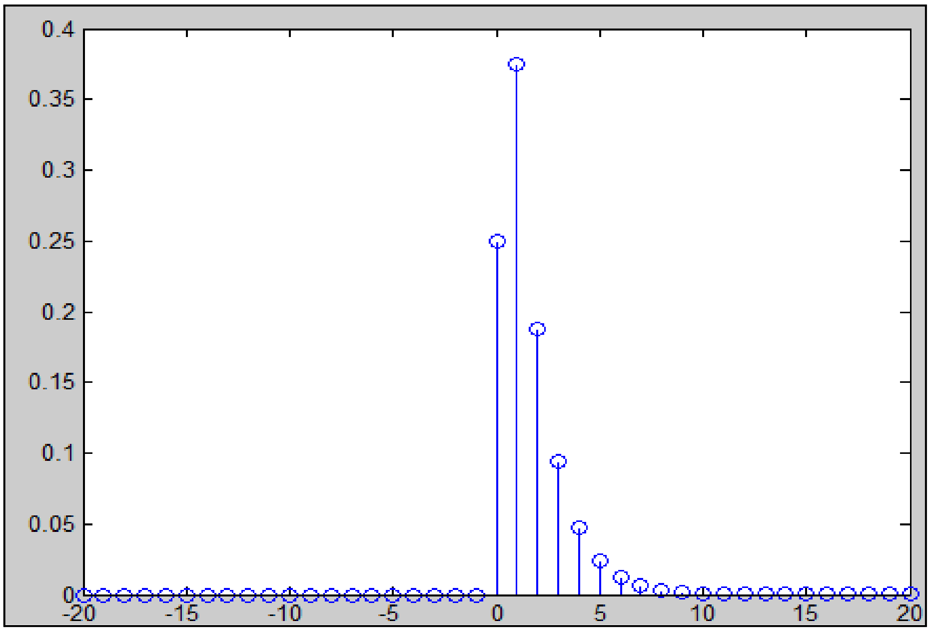
Sistema 3

Para representar el sistema 3 en MATLAB realizamos la siguiente función:

```
function y = sist3(n,x)
    y = zeros(size(n));
    for i = 1:length(n)
        if i == 1
            % Si es el primer elemento, x[n-1] y y[n-1] se asumen como 0
            y(i) = 1/4 * x(i);
        else
            y(i) = 1/4 * x(i) + 1/4 * x(i - 1) + 1/2 * y(i - 1);
        end
    end
end
```

Respuesta **impulsional** obtenida:

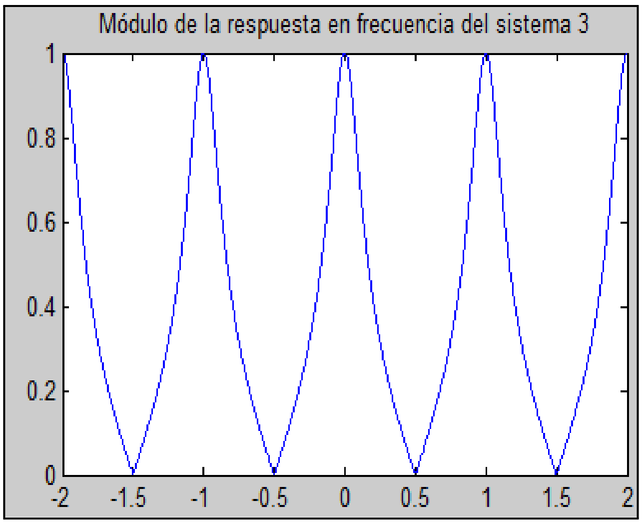
$$h[n] = \frac{1}{4}\delta[n] + \frac{1}{4}\delta[n-1] + u[n-1]\,\frac{3}{8}\left(\frac{1}{2}\right)^{n-1}$$

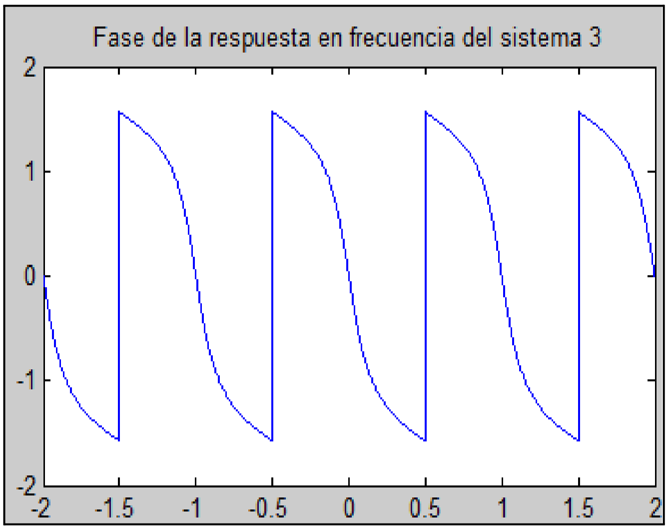


Gráfica de la respuesta impulsional obtenida

Respuesta en **frecuencia** obtenida:

$$H(e^{+j2\pi s}) = \frac{1/4 + 1/4\, e^{-j2\pi s}}{1 - 1/2\, e^{-j2\pi s}}$$



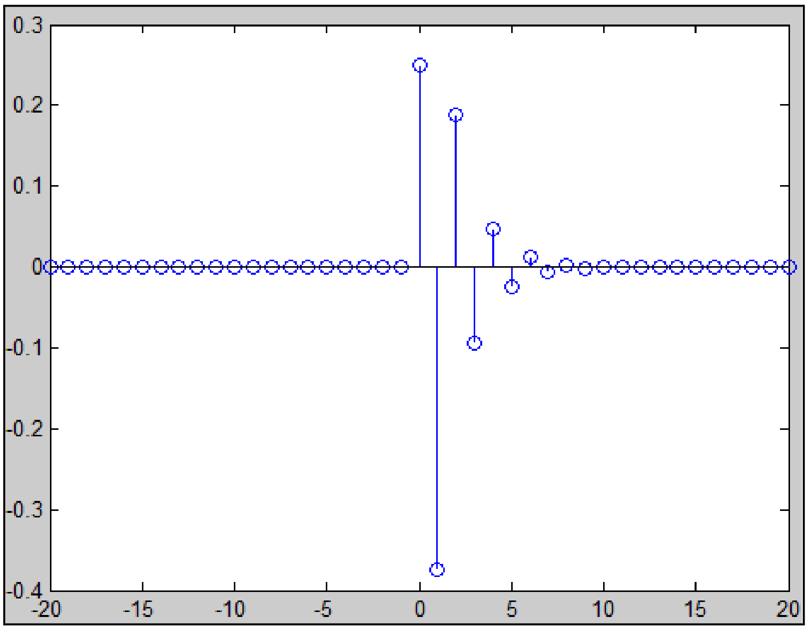


Sistema 4

Para representar el sistema 4 en MATLAB realizamos la siguiente función:

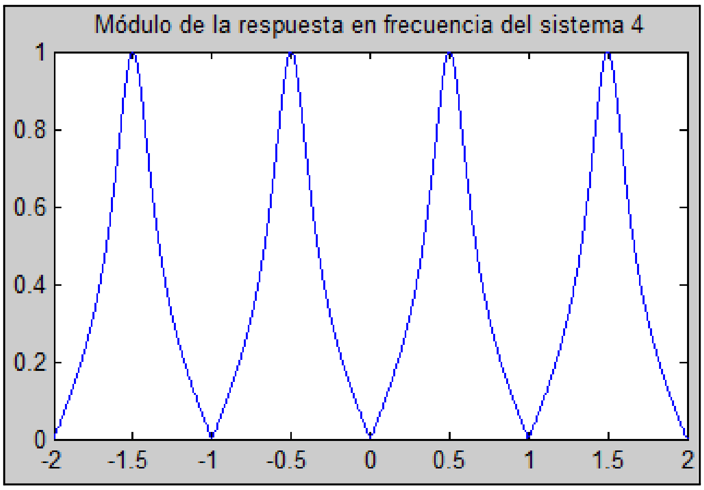
```
function y = sist4(n,x)
    y = zeros(size(n));
    for i = 1:length(n)
        if i == 1
            % Si es el primer elemento, x[n-1] y y[n-1] se asumen como 0
            y(i) = 1/4 * x(i);
        else
            y(i) = 1/4 * x(i) - 1/4 * x(i - 1) - 1/2 * y(i - 1);
        end
    end
end
```

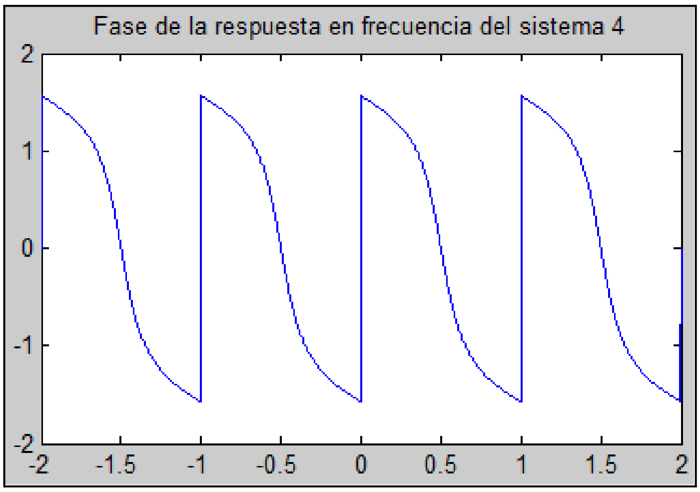
Respuesta **impulsional** obtenida: $h[n] = \frac{1}{4}\delta[n] - \frac{1}{4}\delta[n-1] - \frac{3}{8}\left(-\frac{1}{2}\right)^{n-1} \cdot u[n-1]$



Gráfica de la respuesta impulsional obtenida

Respuesta en **frecuencia** obtenida: $H(e^{j2\pi s}) = \dfrac{1/4 - 1/4\, e^{-j2\pi s}}{1 + 1/2\, e^{-j2\pi s}}$

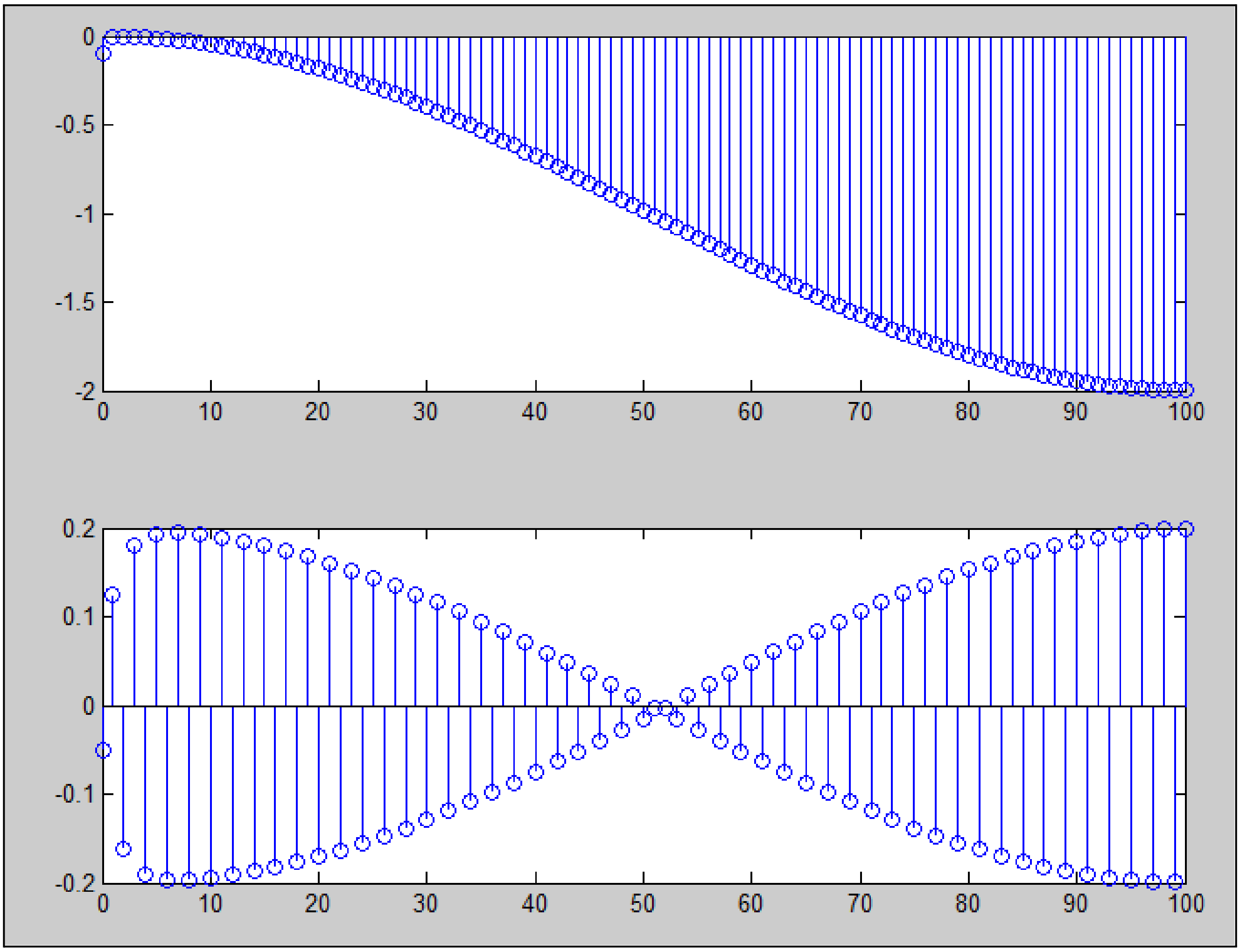




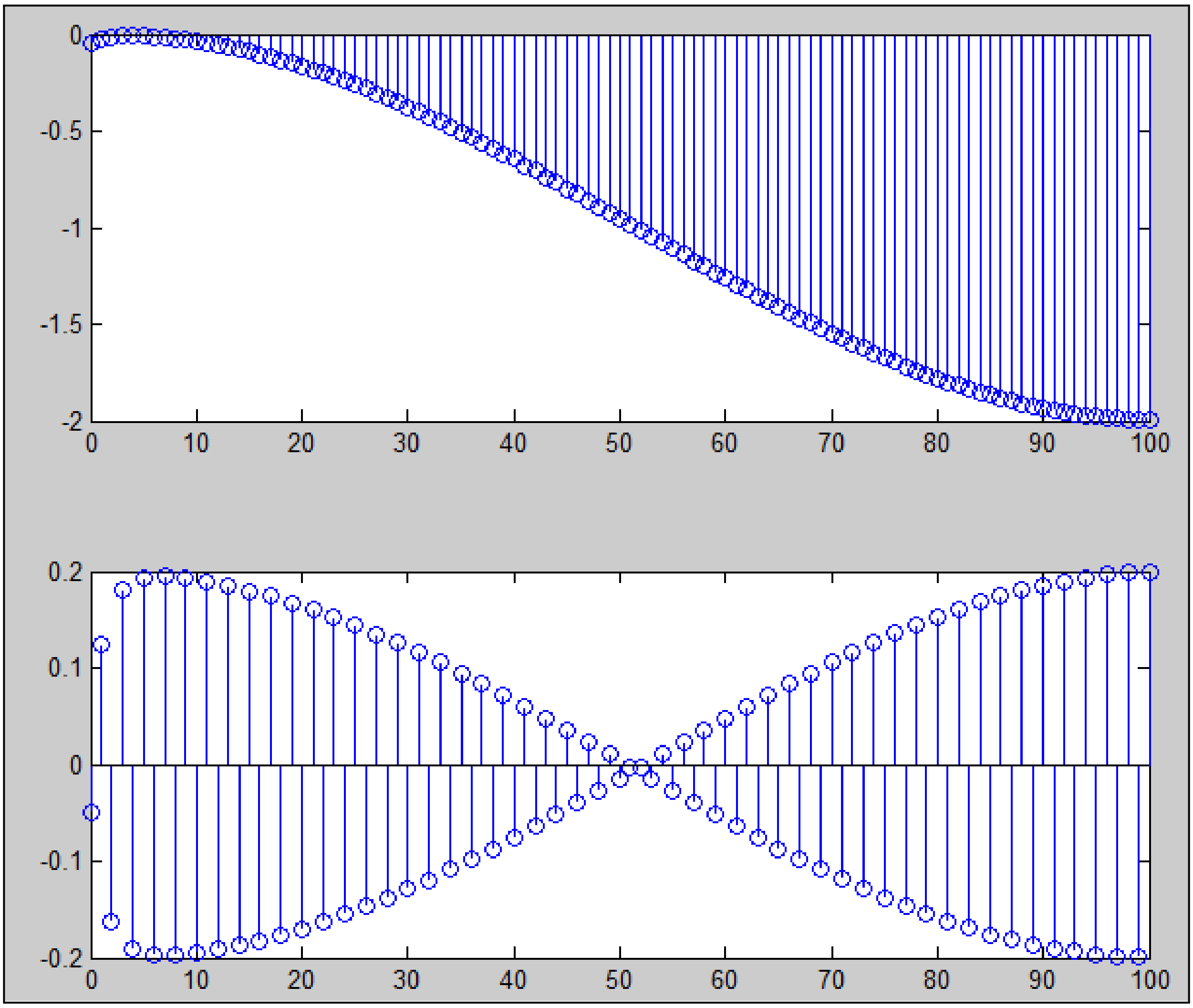
Inciso 4

En el inciso 3 se codificaron funciones que implementan cada uno de los sistemas dados. A partir de dicha implementación, se aplicó la señal “senial.m” dada por la cátedra y se obtuvieron las señales de salida de cada sistema.

Los siguientes 2 gráficos muestran dichas señales de salida obtenidas para el **sistema 1** y el **sistema 2** respectivamente



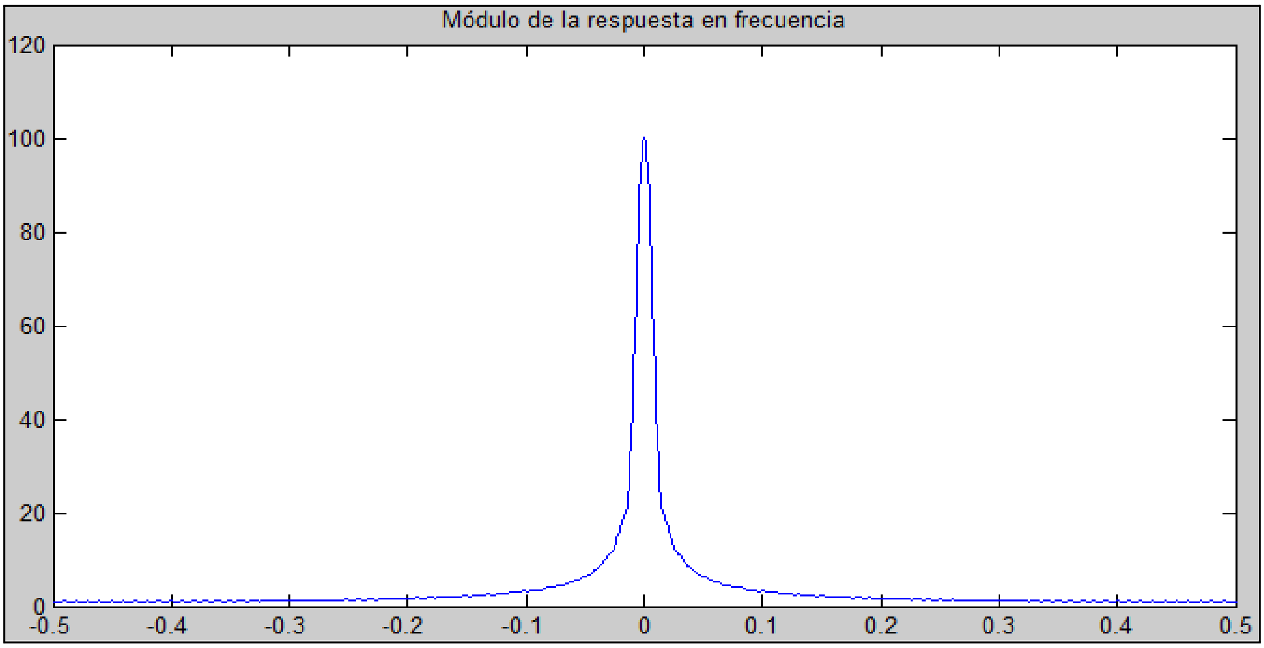
Gráficos de las señales obtenidas para el sistema 3 y 4 respectivamente



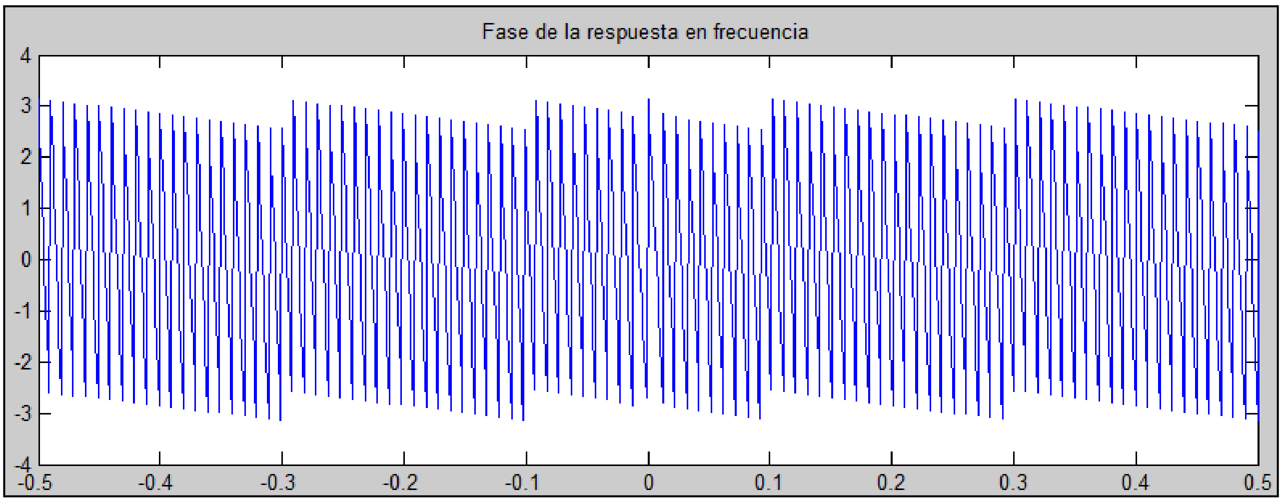
Inciso 5

TFTD S1:

Módulo de la respuesta en frecuencia de la salida del sistema 1

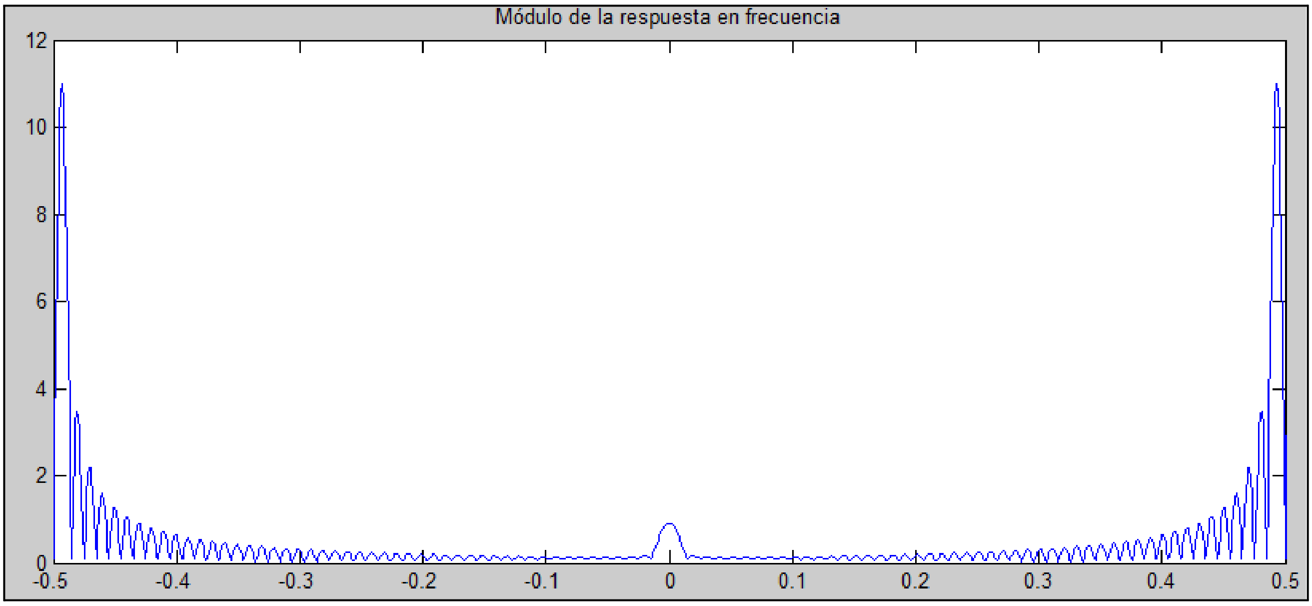


Fase de la respuesta en frecuencia de la salida del sistema 1

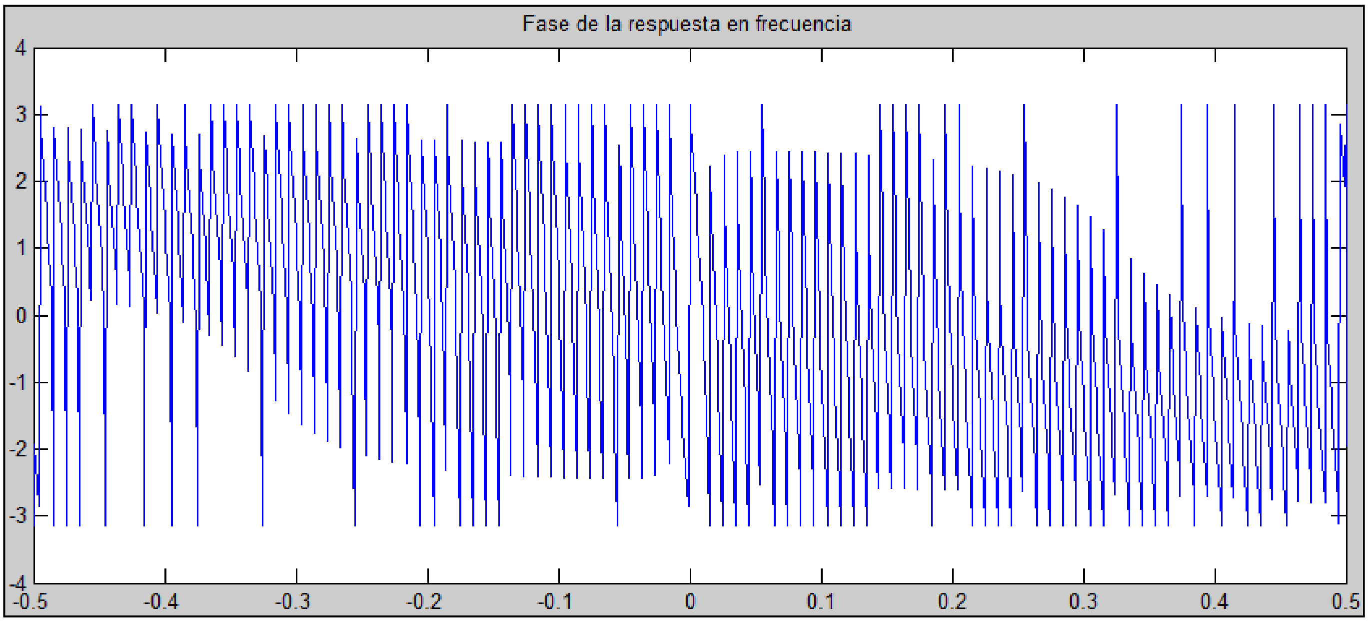


TFTD S2:

Módulo de la respuesta en frecuencia de la salida del sistema 2

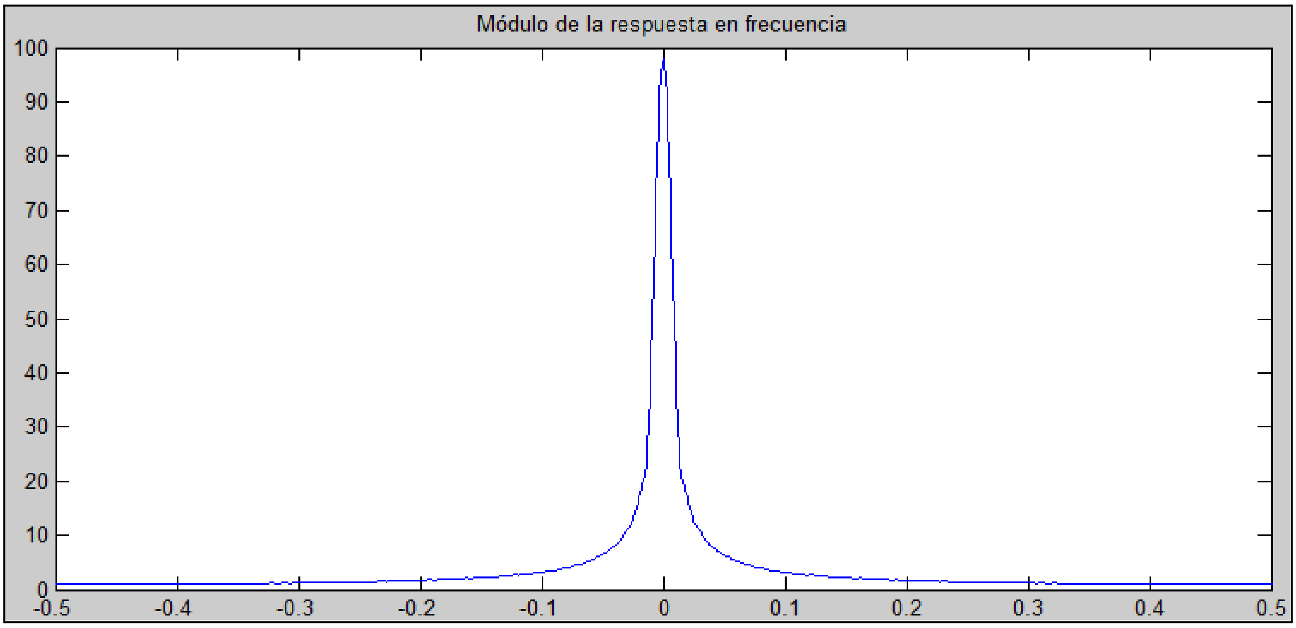


Fase de la respuesta en frecuencia de la salida del sistema 2

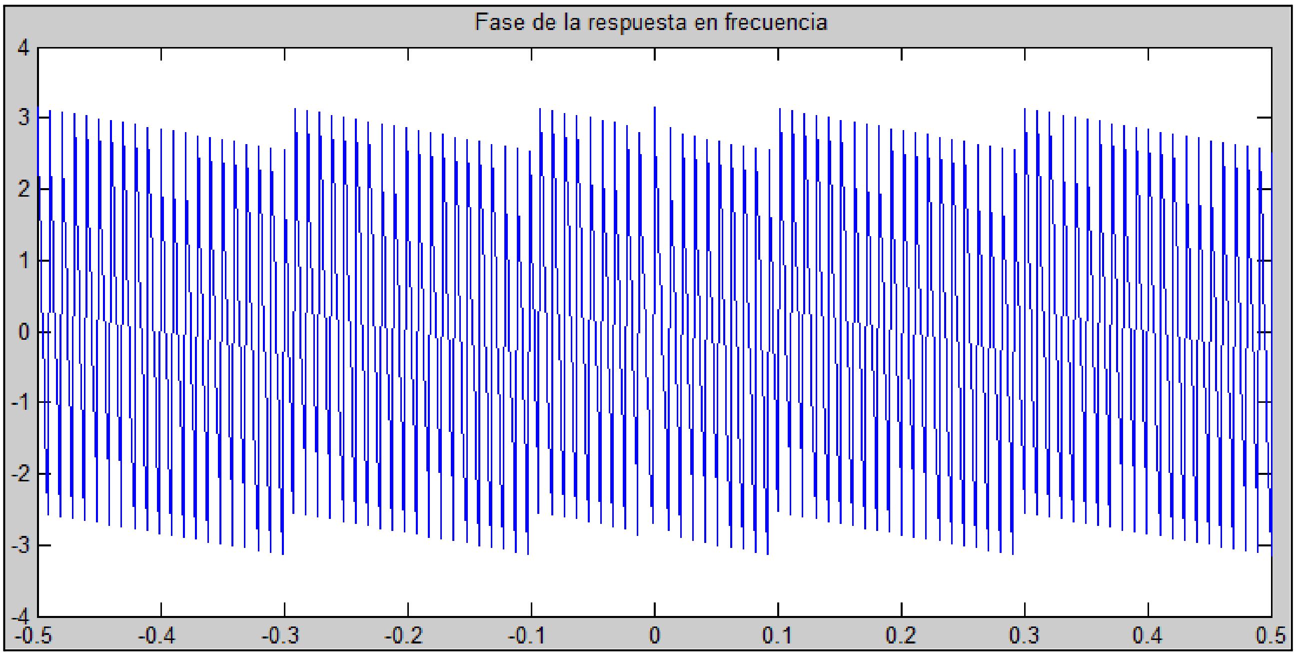


TFTD S3:

Módulo de la respuesta en frecuencia de la salida del sistema 3

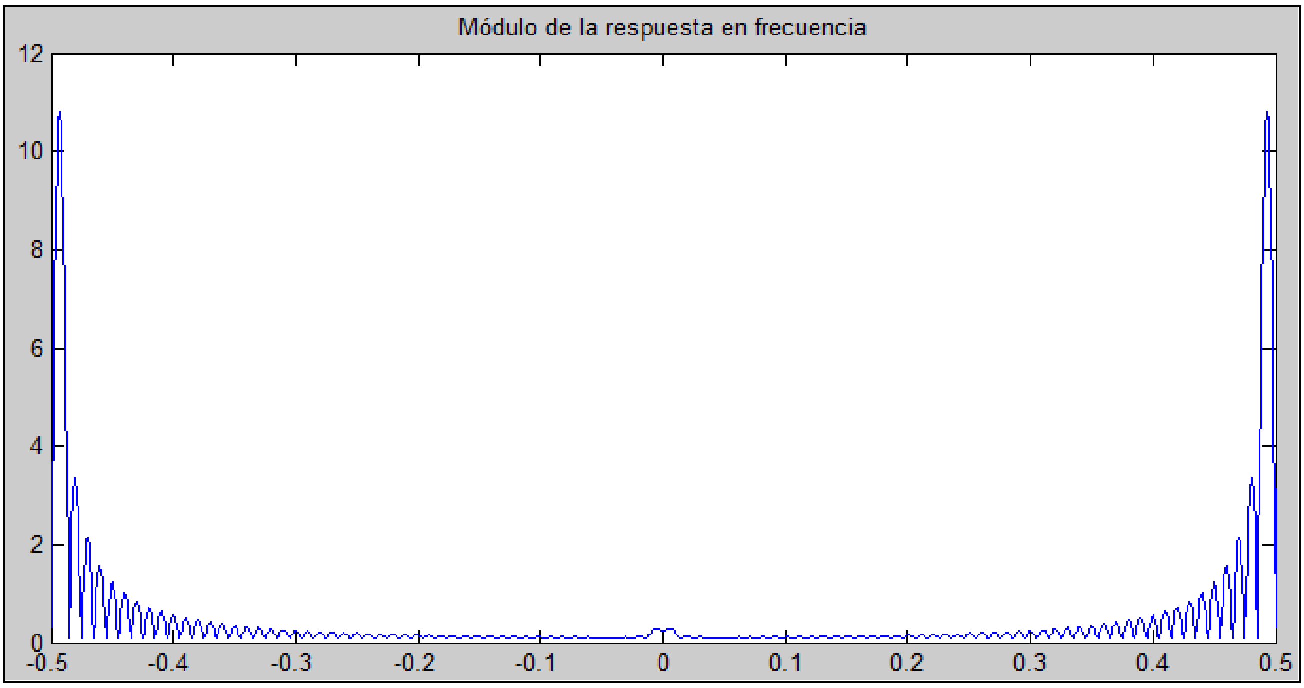


Fase de la respuesta en frecuencia de la salida del sistema 3

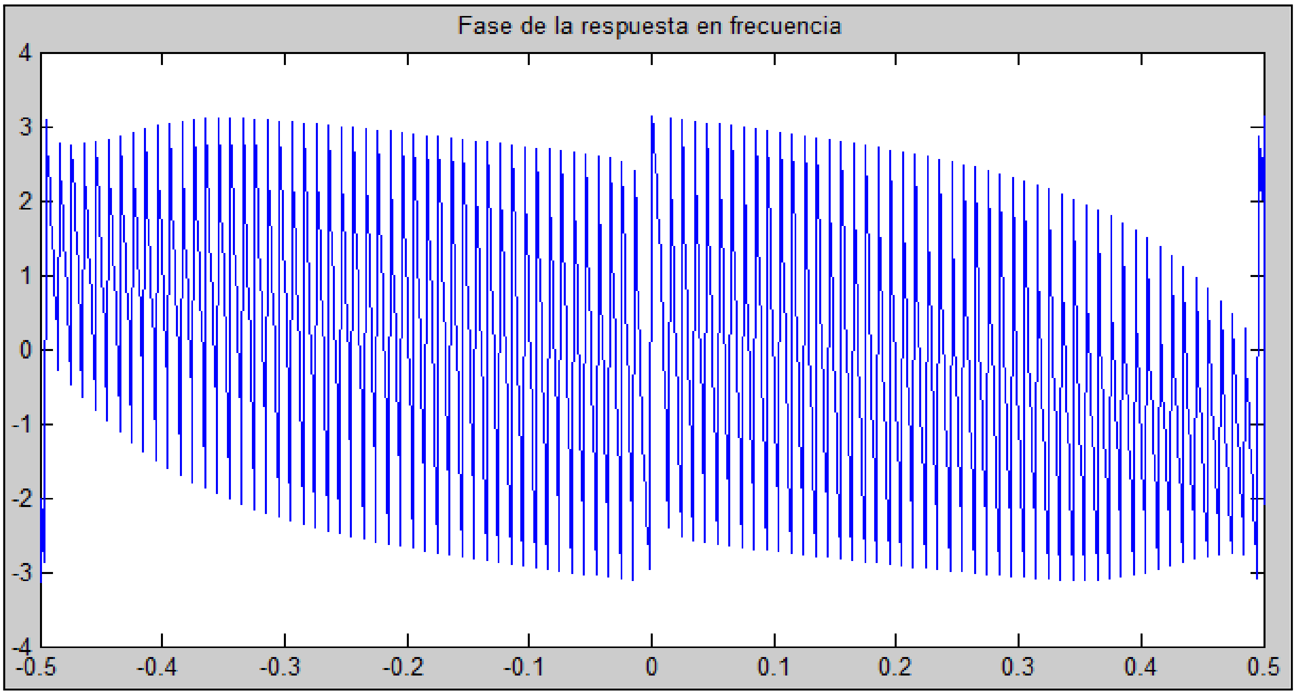


TFTD S4:

Módulo de la respuesta en frecuencia de la salida del sistema 4



Fase de la respuesta en frecuencia de la salida del sistema 4



Nuevamente, como ya se hizo para el inciso 1, se identificaron los componentes de frecuencia de la señal analizando un período de la misma. Observando el gráfico podemos ver que la señal tiene período $T = 1$ y frecuencia fundamental $s = 1/T = 1$. Tomamos un período de la señal comprendido entre $-s/2$ y $s/2$: [-½ , ½]. Este análisis es válido para los cuatro sistemas representados.
Sin embargo presentan distinticiones: tanto para el gráfico de la Transformada de Fourier de Tiempo Discreto del sistema 1 como para el gráfico del sistema 3 se pueden **identificar componentes de frecuencias bajas** (sus picos están cercanos a 0).
Por el contrario para el gráfico de la TFTD del sistema 2 y del sistema 4, utilizando la misma lógica, se pueden **identificar componentes de frecuencias altas** (sus picos están más cercanos a 0,5 y -0,5).

Ejercicio 2

Lista de comandos

- Se utilizaron los mismos comandos ya explicados en el ejercicio 1 y, además, se hizo uso del comando "*sound(x,fs)*" que permite escuchar un sonido almacenado en la variable "*x*" a una frecuencia "*fs*".

Funciones utilizadas

- Se hizo uso de la función "*ec1*" la cual implementa un sistema con la ecuación en diferencias obtenida para la función dada "*hcanald.m*".
- Se utilizaron las funciones "*filtro1*", "*filtro2*", "*filtro3*" las cuales, como su nombre lo indica, son los filtros aplicados a la señal de audio dada. Más adelante se explica de manera más detallada la diferencia entre éstos.

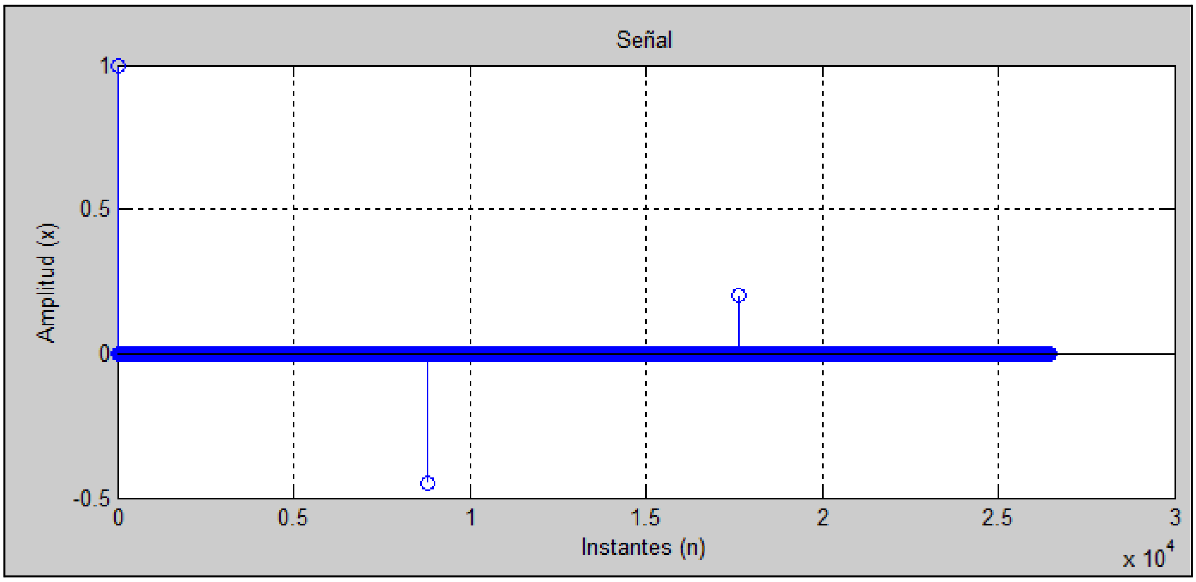
<u>Inciso 1</u>

- **hcanald**

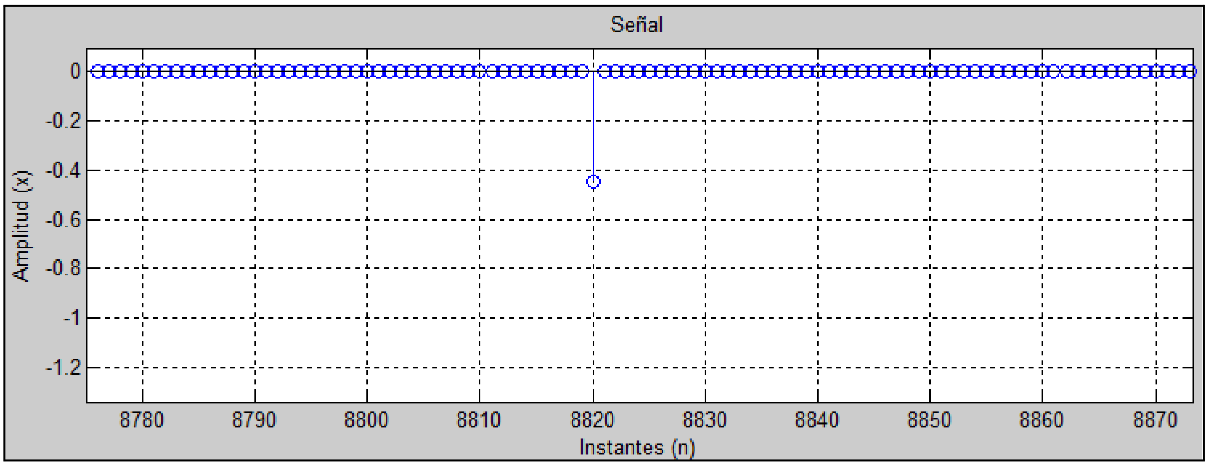
La señal *hA[n]* retornada por la función **hcanald.m** representa la respuesta impulsional de un sistema SLID FIR que modela el comportamiento de un canal digital con dos ecos.

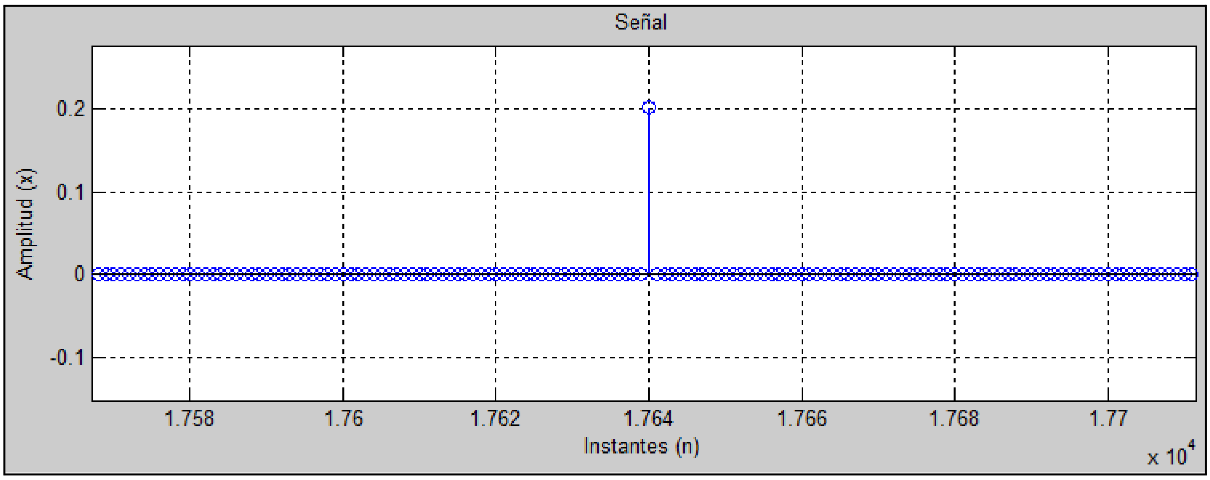
Esta función abarca 4 señales únicas pero sólo retorna 1. Para determinar cuál de las 4 señales debe retornar, se emplea la función *rem(número, 5)*. Esta función divide el número proporcionado por 5 y en función del resto de dicha división elije la señal. En este caso, el número que se debe elegir como entrada es un <u>número de alumno</u>: se utilizó el de Ignacio (**025023**). El resto de dicha división da como resultado *3*, por lo tanto será seleccionada la señal *3* como salida de la función.

Para realizar el gráfico de dicha función se utilizó la función *"stem"*, ya descrita anteriormente. Este fue el resultado:



Se adjuntan dos imágenes adicionales del gráfico previo, con enfoque en las deltas, con el propósito de facilitar un análisis más detallado que abarque su amplitud y ubicación central.





A partir del gráfico y de su análisis, se obtuvo la ecuación de la respuesta impulsional *hA[n]* del sistema:

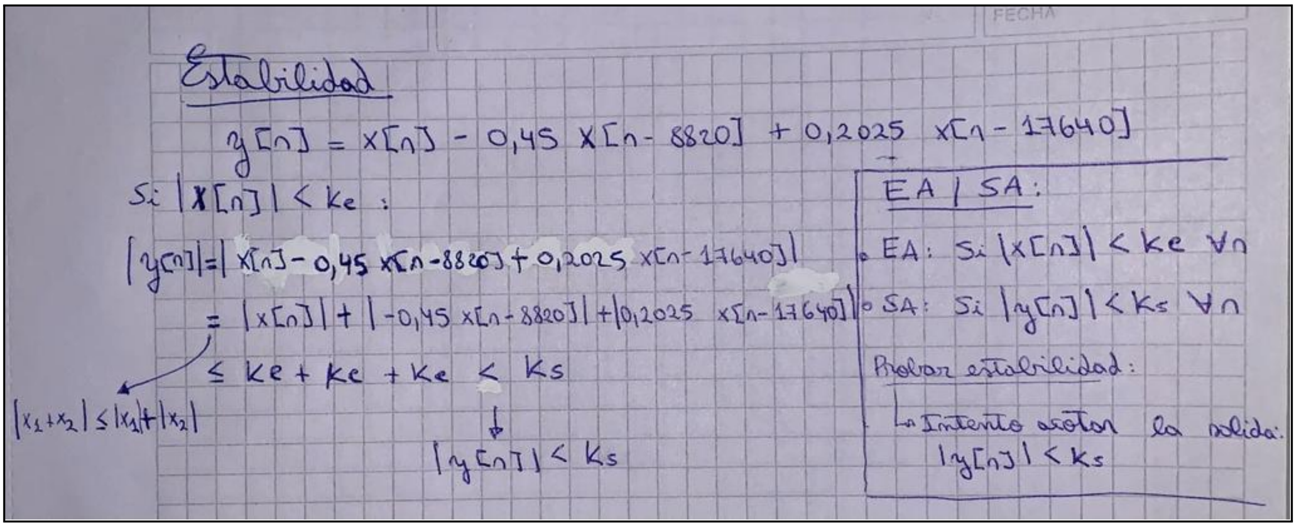
$$\boxed{\mathbf{hA[n] = d[n] - 0.45\, d[n - 8820] + 0.2025\, d[n - 17640]}}$$

Se compone por 3 deltas de Kronecker. La primera en n=0 con amplitud 1, la segunda en n= 8820 con amplitud -0,45 y la tercera en n=17640 con amplitud 0,2025.

A partir de la ecuación de la respuesta impulsional, podemos obtener la ecuación en diferencias del sistema, reemplazando *deltas* por *x[n]*, y h[n] por *y*:

$$\boxed{y[n] = x[n] - 0.45\, x[n - 8820] + 0.2025\, x[n - 17640]}$$

Podemos demostrar la estabilidad del sistema de la siguiente forma:



Estabilidad

$y[n] = x[n] - 0{,}45\, x[n-8820] + 0{,}2025\, x[n-17640]$

Si $|x[n]| < k_e$:

$|y[n]| = |x[n] - 0{,}45\, x[n-8820] + 0{,}2025\, x[n-17640]|$

$= |x[n]| + |-0{,}45\, x[n-8820]| + |0{,}2025\, x[n-17640]|$

$\leq k_e + k_e + k_e < k_s$

$|x_1 + x_2| \leq |x_1| + |x_2|$

$|y[n]| < k_s$

EA / SA:

- EA: Si $|x[n]| < k_e \ \forall n$
- SA: Si $|y[n]| < k_s \ \forall n$

Probar estabilidad:

↳ Intento acotar la salida: $|y[n]| < k_s$

Inciso 2

Función *ec1(n,x)* que representa el sistema:

$$y[n] = x[n] - 0.45\, x[n - 8820] + 0.2025\, x[n - 17640]$$

```
function y =ec1(n,x)
    y = zeros(size(n));  % Inicializar la señal de salida

    for i = 1:length(n)
        y(i) = x(i);

        if i > 8820
            y(i) = y(i) - 0.45 * x(i - 8820);
        end

        if i > 17640
            y(i) = y(i) + 0.2025 * x(i - 17640);
        end
    end
end
```

Entradas:

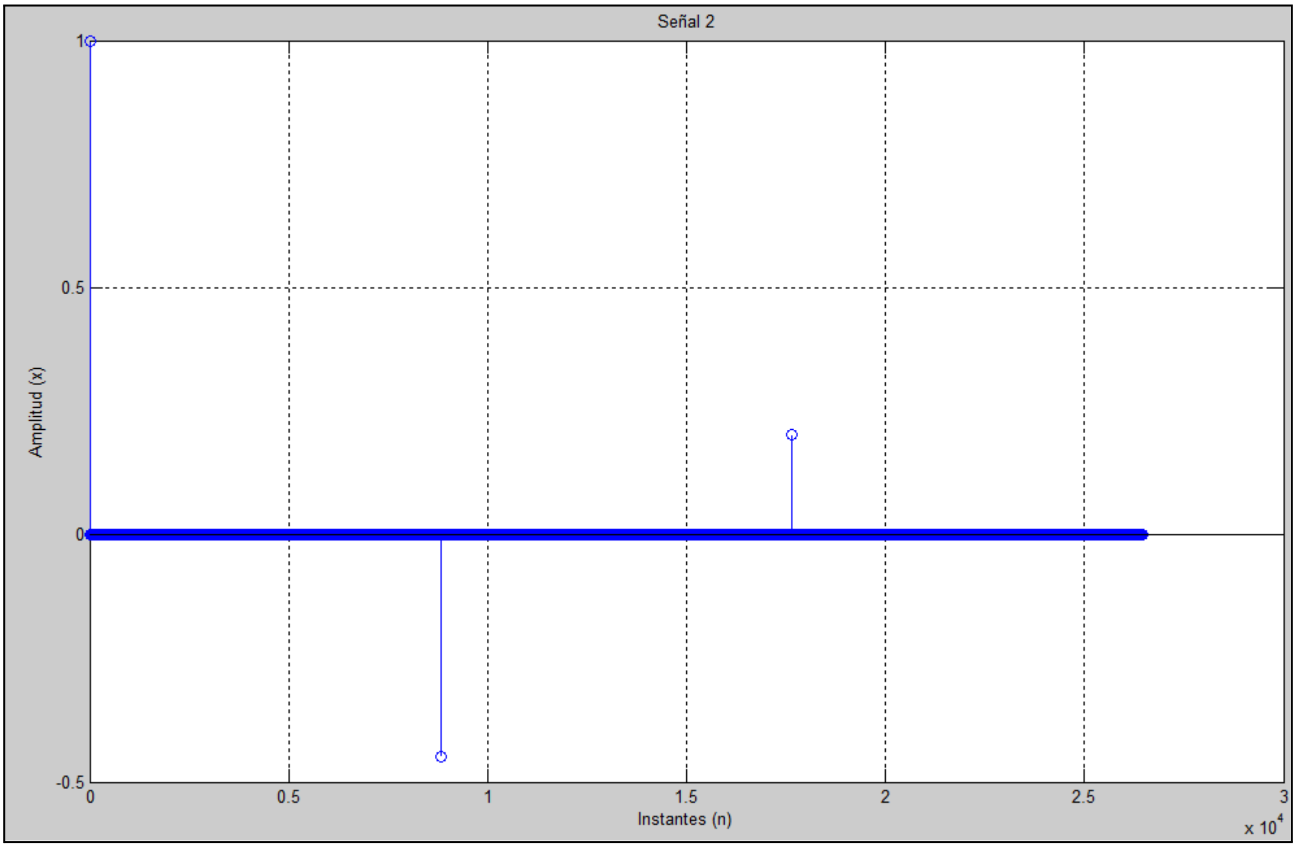
- *n*: vector de instantes de la función.
- *x*: vector con valores de la función en los instantes *n*.

Salida:

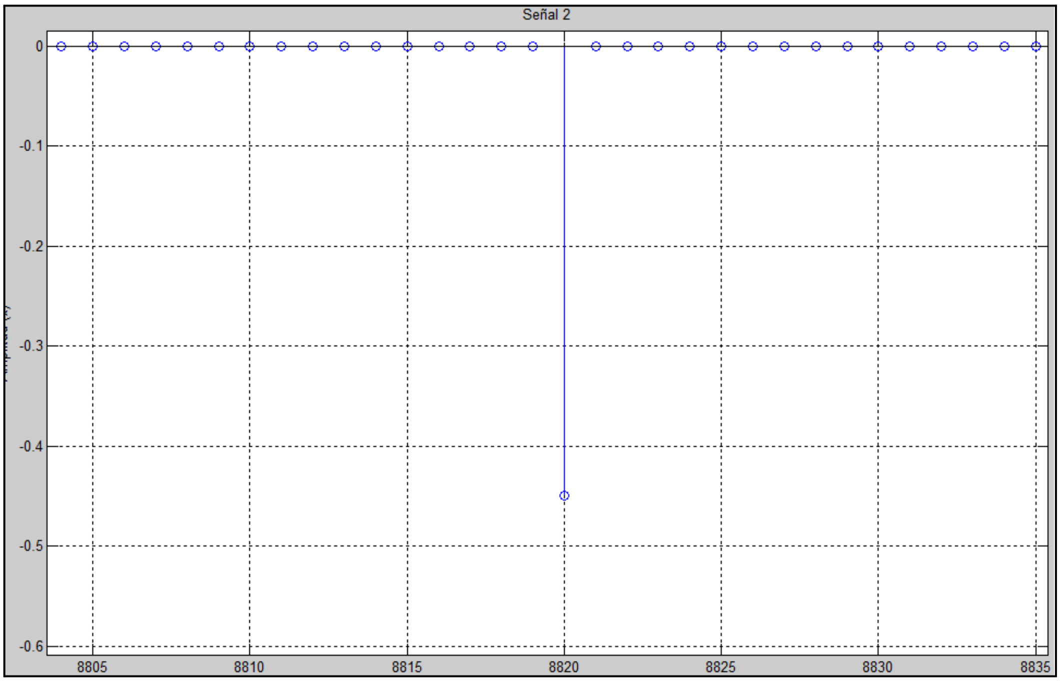
- *y*: vector de la expresión salida de tamaño *n*.

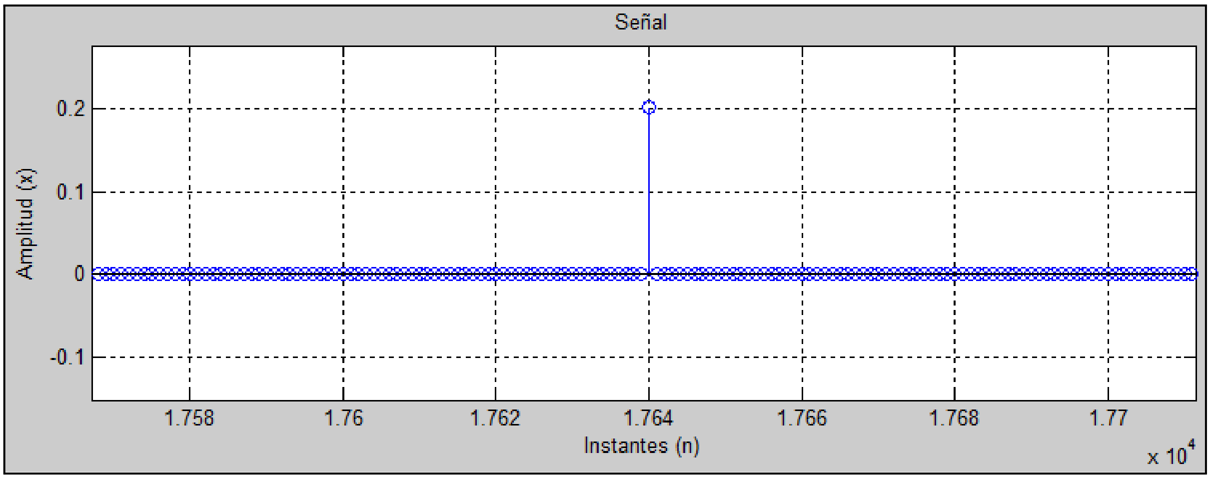
Esta función compone la ecuación descrita anteriormente, agregando términos dependiendo en el instante que se encuentra el *for*.

Al graficar su respuesta impulsional obtenemos el siguiente gráfico:



Se verifican las mismas 3 deltas que las del inciso 1: están centradas en los mismos puntos y su amplitud es igual. Como anteriormente se adjuntan imágenes para que puedan observarse con mayor detalle la amplitud y la posición de las deltas.

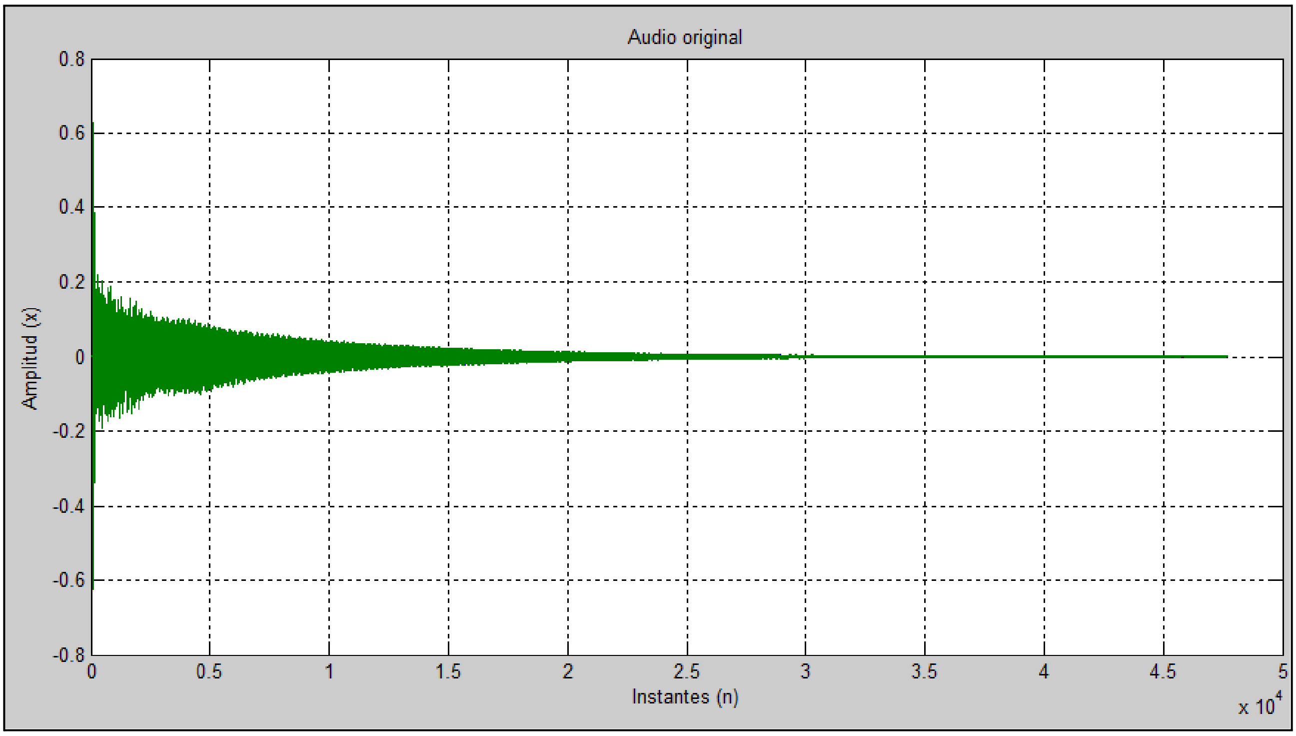




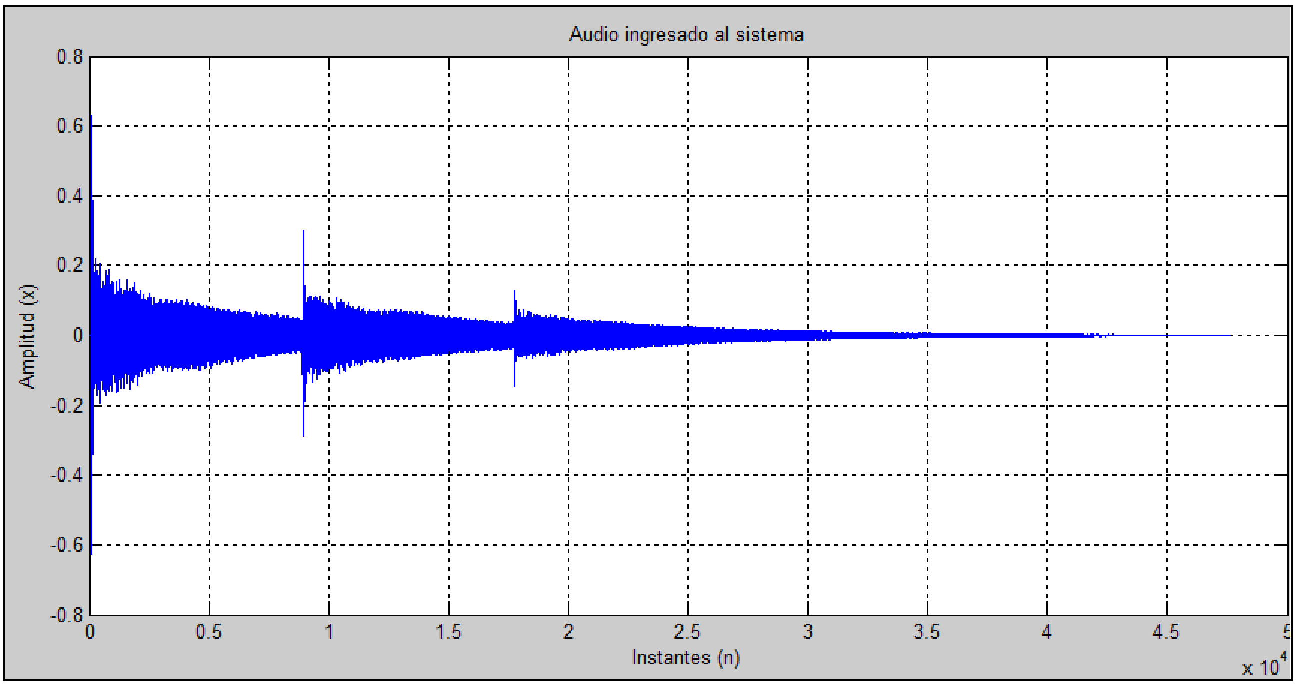
Inciso 3

Al ingresar el audio al sistema, se escucha el audio original junto con dos ecos o retardos. Esto puede verse expresado gráficamente:

*Gráfica del **audio original***



*Gráfica de su **ingreso al sistema** (señal de salida en el tiempo).*

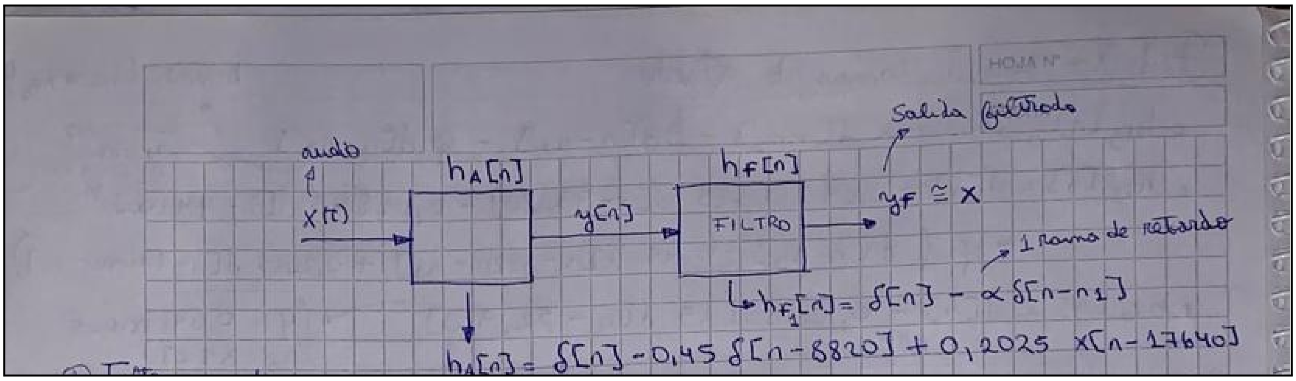


Inciso 4

Con el fin de reducir los ecos detectados, se hizo uso de diferentes **filtros** (la manera analítica en la que se halló dónde estaban ubicados los ecos que se fueron generando al pasar por los distintos filtros, son mostrados luego de los gráficos de todos los filtros).

En primer lugar, debemos entender cómo funcionan estos filtros y cual es la manera de conseguir su ecuación en diferencias. El audio *x(t)* ingresa al primer sistema con respuesta impulsional *hA[n]* y salida *y[n]*. Luego, esa señal deberá ser filtrada por un segundo sistema, de tal manera que el audio original pueda ser reconstruido.

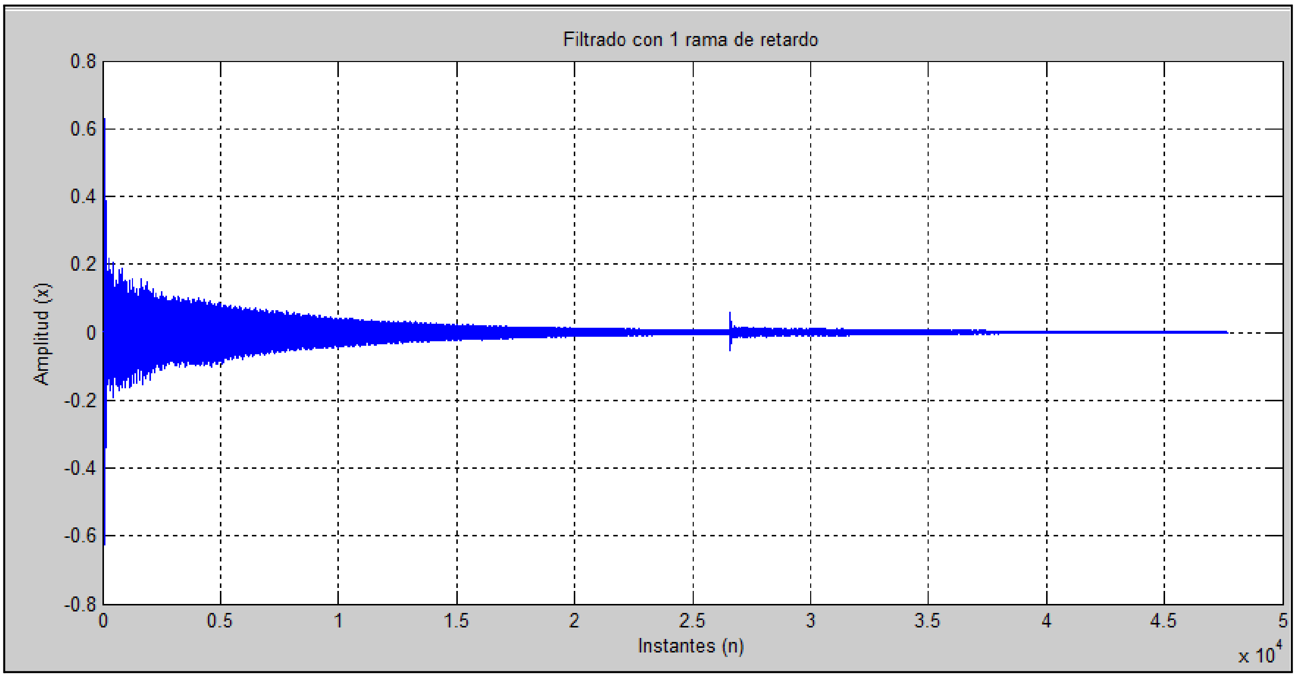
Nuestro sistema total, tendrá una salida llamada *yf*, y una respuesta impulsional llamada *ht*. Podemos definir *y = x conv hA*. También *yf = y conv hf* , donde *hf* es la respuesta impulsional del filtro. Por lo tanto, *yf = x conv hA conv hf*. Finalmente, podemos definir *ht = hA conv hf*.



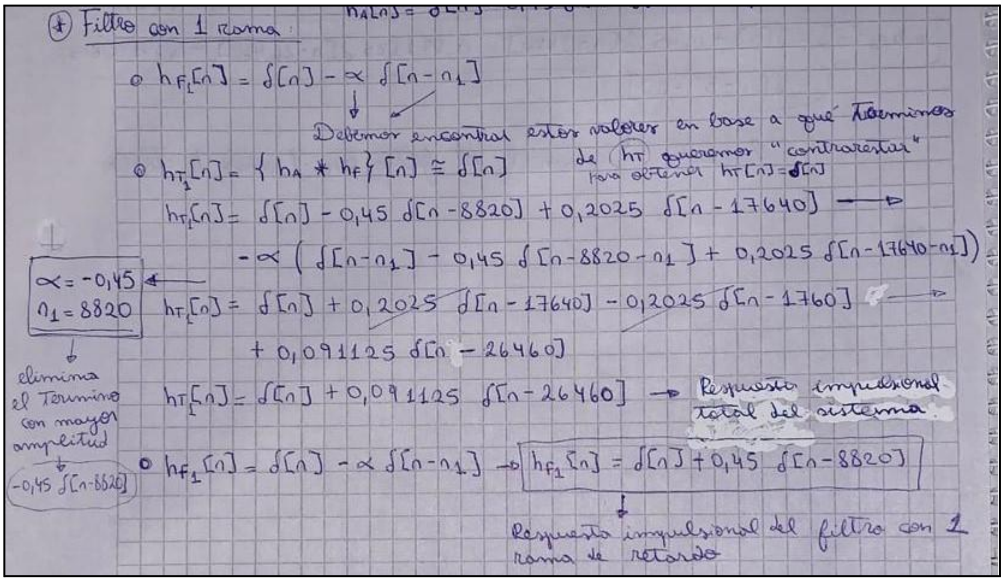
Según la definición de filtro, la expresión de la respuesta impulsional para **n** ramas de retardo será:

hf**n**[n] = d[n] - a1 d[n - n1] - a2 d[n - n2] - … - a**n** d[n-n**n**]

En primera instancia se utilizó un filtro de **una rama** el cual fue implementado en una función y se la llamó "*filtro1*" en el código de matlab. Éste se encarga de reducir los ecos ya vistos, sin embargo, puede observarse (tanto gráfica como analíticamente) que éste añade un eco de menor amplitud en una posición distinta a los dos ecos ya conocidos.

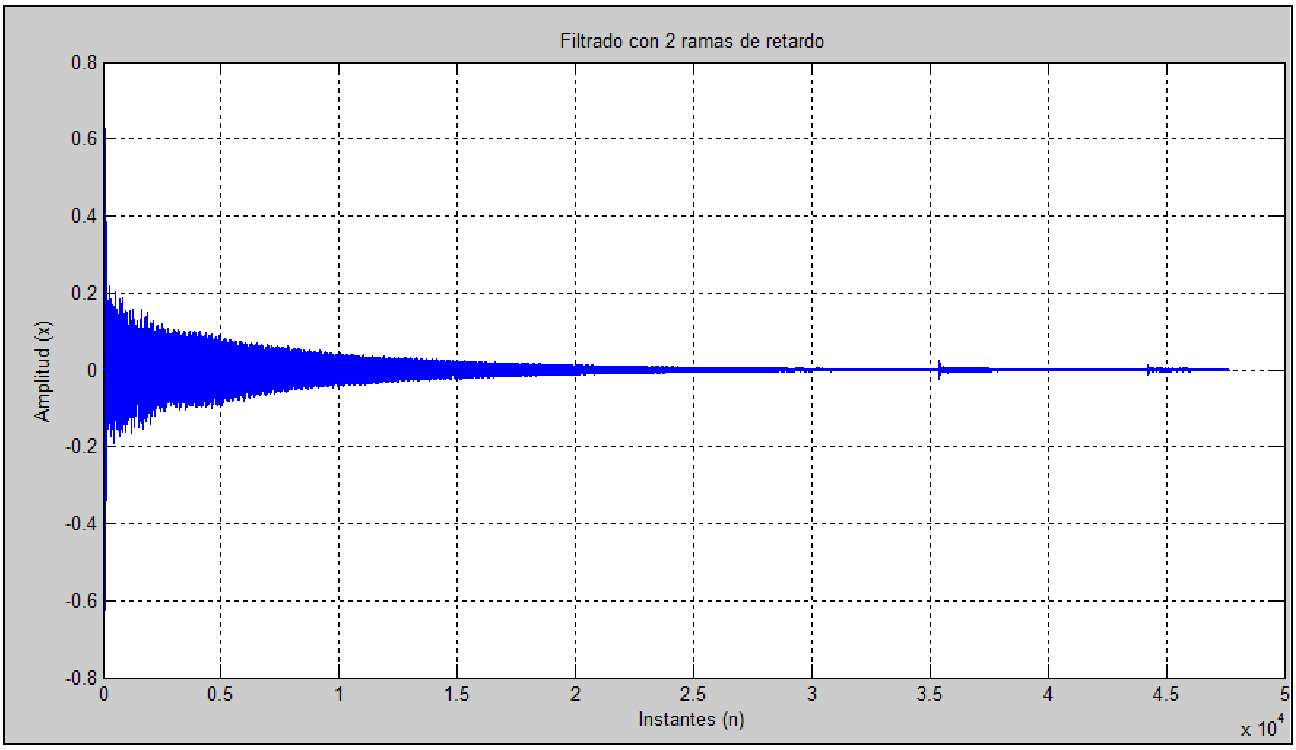


La respuesta impulsional del filtro puede ser calculada de manera analítica, permitiendo identificar los nuevos ecos generados por el filtro:

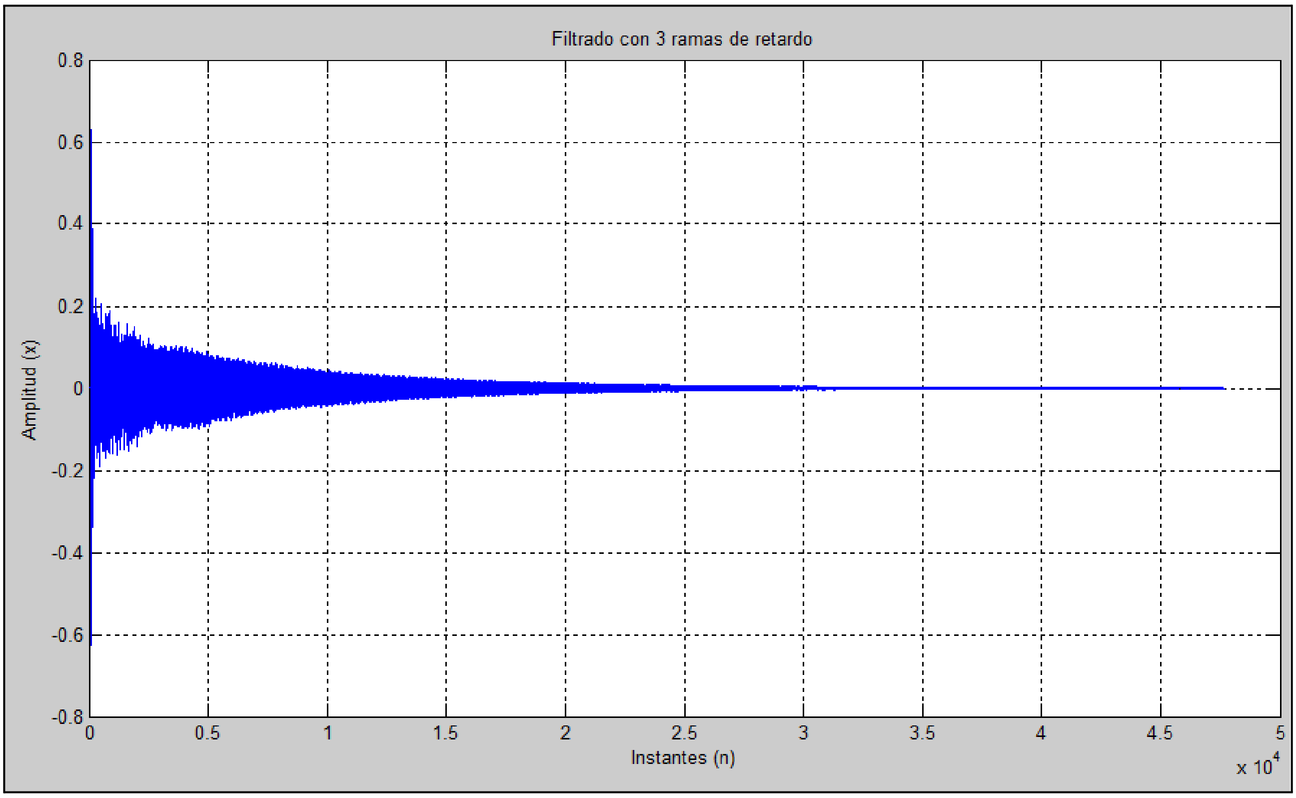


Con los demás filtros, se realiza el mismo cálculo pero colocando los filtros en cascada. Los cálculos para cada filtro están debajo.

Debido a este nuevo eco, se aplica un filtro de **2 ramas** con la intención de eliminarlo o reducirlo lo máximo posible. Este nuevo filtro fue implementado en otra función distinta a la anterior mencionada y se la llamó "*filtro2*" el cual elimina o reduce el eco generado por el *filtro1* y aunque si bien agrega 2 nuevos ecos, éstos son aún de menor amplitud que el generado anteriormente. En conclusión se logra eliminar o reducir los ecos de mayor amplitud y aunque aparecen nuevos, éstos son más despreciables que los ya eliminados. Se muestra una imagen donde pueden verse gráficamente:



Por último, se aplica un filtro de **3 ramas** logrando así reducir la mayor cantidad de eco posible. Éste fue implementado en otra función llamada "*filtro3*" y el resultado fue el siguiente:



Puede apreciarse que los filtros dieron resultado y no hay ecos.

A continuación se adjunta una imagen con el detallado análisis matemático seguido para lograr llegar a las expresiones que componen los filtros. Una explicación previa para dar una introducción a la imagen: A partir de la ecuación en diferencias del sistema ($y[n] = x[n] - 0.45\,x[n - 8820] + 0.2025\,x[n - 17640]$), se comenzaron a aplicar los distintos filtros mencionados anteriormente los cuales están compuestos por la diferencia entre una delta de Kronecker centrada en 0 y el producto de un número arbitrario (alfa) y una delta de Kronecker centrada en otro número arbitrario (n1). Este *alfa* y *n1* son elegidos teniendo en cuenta la amplitud y dónde está centrado el eco de mayor amplitud.

$$h_A[n] = \delta[n] - 0{,}45\,\delta[n-8820] + 0{,}2025\,x[n-17640]$$

⊛ Filtro con 1 rama:

- $h_{F_1}[n] = \delta[n] - \alpha\,\delta[n-n_1]$

Debemos encontrar estos valores en base a qué términos de h_T queremos "contrarrestar" para obtener $h_T[n] = \delta[n]$

- $h_{T_1}[n] = \{h_A * h_F\}[n] \cong \delta[n]$

$$h_{T_1}[n] = \delta[n] - 0{,}45\,\delta[n-8820] + 0{,}2025\,\delta[n-17640] \longrightarrow$$
$$-\alpha\left(\delta[n-n_1] - 0{,}45\,\delta[n-8820-n_1] + 0{,}2025\,\delta[n-17640-n_1]\right)$$

$$\boxed{\begin{aligned}\alpha &= -0{,}45\\ n_1 &= 8820\end{aligned}}$$

elimina el Término con mayor amplitud

$-0{,}45\,\delta[n-8820]$

$$h_{T_1}[n] = \delta[n] + \cancel{0{,}2025\,\delta[n-17640]} - \cancel{0{,}2025\,\delta[n-1760]} \longrightarrow$$
$$+ 0{,}091125\,\delta[n-26460]$$

$$h_{T_1}[n] = \delta[n] + 0{,}091125\,\delta[n-26460] \rightarrow$$ Respuesta impulsional total del sistema.

- $h_{F_1}[n] = \delta[n] - \alpha\,\delta[n-n_1] \rightarrow \boxed{h_{F_1}[n] = \delta[n] + 0{,}45\,\delta[n-8820]}$

Respuesta impulsional del filtro con 1 rama de retardo

⊛ Filtro con 2 ramas de retardo:

- $h_{F_2}[n] = \delta[n] - \alpha\,\delta[n-n_1] - \beta\,[n-n_2]$ — $h_{T_2} = \{h_A * h_{F_2}\}[n] \cong \delta[n]$

$$\boxed{\begin{aligned}\alpha &= -0{,}45\\ n_1 &= 8820\\ \beta &= 0{,}091125\\ n_2 &= 26460\end{aligned}}$$

- $h_{T_2}[n] = \delta[n] + 0{,}091125\,\delta[n-26460] - \beta\left(\delta[n-n_2] - 0{,}45\,\delta[n-8820-n_2]\right.$
$\left. + 0{,}2025\,\delta[n-17640-n_2]\right)$

$$h_{T_2}[n] = \delta[n] = 0{,}041006\,25\,\delta[n-35280] - 0{,}01845\,\delta[n-44100]$$

- $\boxed{h_{F_2}[n] = \delta[n] + 0{,}45\,\delta[n-8820] - 0{,}091125\,\delta[n-26460]}$

Respuesta impulsional del filtro con 2 ramas de retardo

4 Filtro con 3 ramas de retardo:

$h_{T3}[n] = \{ h_4 * h_{F3} \}$

- $h_{F3}[n] = \delta[n] - \alpha\,\delta[n-n_1] - \beta\,\delta[n-n_2] - \varphi\,\delta[n-n_3]$

$\alpha = -0{,}45$, $n_1 = 8820$, $\beta = 0{,}091125$, $n_2 = 26460$

- $h_{T3}[n] = \delta[n] + 0{,}04100625\,\delta[n-35280] - 0{,}01845\,\delta[n-44100]$
  $- \varphi\left(\delta[n-n_3] - 0{,}45\,\delta[n-8820-n_3] + 0{,}2025\,\delta[n-17640-n_3]\right)$

$h_{T3}[n] = \delta[n] - 0{,}0083025\,\delta[n-52{,}920] \rightarrow \varphi = 0{,}04100625$, $n_3 = 35280$

- $h_{F3}[n] = \delta[n] + 0{,}45\,\delta[n-8820] - 0{,}091125\,\delta[n-26460] - 0{,}04100625\,\delta[n-35280]$

Respuesta impulsional del filtro con 3 ramas de retardo

Conclusión

En este segundo ejercicio se abordó de manera práctica la utilización y aplicación de diferentes **filtros** para completar el desafío de lograr la máxima reducción de ecos posibles de la señal de audio “*hcanald.m*”.

Se implementaron tres filtros distintos: el primer filtro aplicado consta de una única rama, el segundo filtro de dos ramas y el tercer filtro de 3 ramas. A medida que fueron aplicados se pudo observar que, si bien tanto el primer filtro como el segundo lograron reducir significativamente la presencia de ecos, cada uno de ellos agregaba nuevos aunque más despreciables (de menor magnitud). A la hora de aplicar el tercer filtro se pudo notar que los ecos quedaron completamente reducidos llegando a la conclusión de que, si bien es algo laborioso, se pueden obtener muy buenos resultados a medida que agregamos ramas al filtro logrando así corregir las imperfecciones del sonido.