# PROCESAMIENTO DIGITAL DE SEÑALES

## PROYECTO INTEGRADOR

**GRUPO 6**

**INTEGRANTES:**

* Guerra, Juan Carlos
* Pary, Nelson Guillermo
* Rodríguez, Ignacio

CARRERA: Ingeniería en Computación

AÑO DE CURSADO: 2016

**ENUNCIADO**

Se desea construir un banco de filtros digitales para cubrir un rango de 50 a 150 Hz, destinado a un instrumento utilizado para seleccionar las notas musicales básicas de la primera octava. Para ello se divide el entorno 50 a 150 Hz en 7 bandas angostas muy selectivas centradas en las frecuencias correspondientes a cada nota:

Do 1: 65,406 Hz (Do# 1: 69,296)

Re 1: 73,416 Hz (Re# 1: 77,782)  
Mi 1: 82,407 Hz  
Fa 1: 87,307 Hz (Fa# 1: 92,499)  
Sol 1: 97,999 Hz (Sol#1: 103,826)  
La 1: 110 Hz (La# 1: 116,541)  
Si 1: 123,471 Hz

El filtro digital a diseñar deberá tener una transferencia unitaria a la frecuencia máxima, y dicha frecuencia deberá ser la especificada para cada nota. Las bandas deben estar limitadas por las frecuencias de media potencia (0.707 del valor de máxima transferencia, o -3 dB) cuyo valor debe ser ~±2 % respecto de la frecuencia central. Especificar la atenuación en la banda de transición.

Para el diseño utilizar las técnicas de diseño: **DIRECTA**.

Como prototipo pasabajos utilizar un filtro: **BUTTERWORTH**.

Desarrollar analíticamente la solución de cada punto, luego usar Matlab para encontrar los resultados numéricos y graficar.

1. Especificar el orden del filtro que cumpla las especificaciones.
2. Obtener los coeficientes de los filtros correspondientes a cada banda.
3. Proponer una estructura de realización. Especificar el hardware.
4. Encontrar la respuesta temporal de cada filtro y del filtro total.
5. Encontrar la respuesta en frecuencia Amplitud |H(ω)| y Fase Φ(ω) vs. Frecuencia ω, con ωmáx = π/T. Expresar los resultados en una gráfica de ~10cm en ordenadas (Amplitud y Fase) por ~15cm en abscisas (Frecuencia).
6. Encontrar la FFT de todas las señales para N=32, y representar todas en un mismo gráfico de magnitud en función de la frecuencia.
7. Verificación del diseño: superponer en un mismo gráfico (lineal) las respuestas de amplitud de los siete filtros.
8. Implementar una simulación computacional en Matlab/Simulink visualizando la salida del sistema, utilizando una señal de excitación sinusoidal en el rango de frecuencias permitidas por el diseño

Escribir conclusiones, observaciones y sugerencias sobre los resultados obtenidos.

**CASO DE ESTUDIO**

Se desea construir un banco de filtros digitales para cubrir un rango de frecuencias de 50 a 150Hz, destinado a un instrumento utilizado para seleccionar las 7 notas musicales básicas de la primera octava. Para ello se divide el espectro 50 a 150 Hz en 7 bandas angostas muy selectivas centradas en las frecuencias correspondientes a cada nota:

* Do 1: 65,406 Hz
* Re 1: 73,416 Hz
* Mi 1: 82,407 Hz
* Fa 1: 87,307 Hz
* Sol 1: 97,999 Hz
* La 1: 110 Hz
* Si 1: 123,471 Hz

El filtro digital a diseñar deberá tener una transferencia unitaria a la frecuencia central, y dicha frecuencia deberá ser la especificada para cada nota. Las bandas estarán limitadas por las frecuencias de media potencia (0,707 del valor de máxima transferencia, o -3dB) cuyos valores serán ±1 Hz con respecto a la frecuencia central de cada nota. Luego definimos las frecuencias de las bandas de atenuación (paso la banda de transición) a ± 2,9 Hz pero respecto de las frecuencias de corte en vez de la central y a estas frecuencias lograremos una atenuación de 4dB.

Para el desarrollo de dichos filtros vamos a utilizar el método directo, el cual consiste en realizar nuestro filtro digital a partir de otro, también digital, llamado prototipo, mediante transformaciones espectrales. El filtro prototipo es un filtro pasabajos Butterworth.

Calcularemos los filtros a utilizar analíticamente de manera cualitativa, luego usaremos Matlab y su herramienta fdatools para obtener los filtros cuantitativamente.

**DESARROLLO DEL TEMA**

Explicaremos cómo calcular un filtro digital **Butterworth** usando el **método directo**.

En la siguiente tabla se presentan las frecuencias de corte, atenuación y central para cada nota:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Frecuencias[Hz]** | | | | |
| **Nota** | **fai** | **fci** | **f0** | **Fcs** | **fas** |
| Do | 61,506 | 64,406 | 65,406 | 66,406 | 69,306 |
| Re | 69,516 | 72,416 | 73,416 | 74,416 | 77,316 |
| Mi | 78,507 | 81,407 | 82,407 | 83,407 | 86,307 |
| Fa | 83,407 | 86,307 | 87,307 | 88,307 | 91,207 |
| Sol | 94,099 | 96,999 | 97,999 | 98,999 | 101,899 |
| La | 106,1 | 109 | 110 | 111 | 113,9 |
| Si | 119,571 | 122,471 | 123,471 | 124,471 | 127,371 |

Donde:

f0: frecuencia central

fci: frecuencia de corte inferior

fai: frecuencia de atenuación inferior

fcs: frecuencia de corte superior

fas: frecuencia de atenuación superior

En vez de calcular las especificaciones de 7 filtros por separado, puede observarse en la tabla precedente que la región espectral más comprometida es la que corresponde a las notas Mi y Fa dada su cercanía en frecuencia, por lo que procederemos a calcular el filtro para Mi, que usaremos como base para los demás filtros, ya que si cumple las especificaciones para la región más comprometida en las demás regiones las especificaciones se cumplirían por exceso.

Basándonos en el criterio anterior, se toman las especificaciones para la nota MI y, a continuación, se calcula el orden del filtro Butterworth. Se debe aclarar que el orden que se obtendrá a continuación corresponde a un filtro prototipo pasabajos, el orden del filtro pasabandas deseado será el doble del orden calculado para el prototipo pasabajos. La frecuencia de muestreo se calcula usando el criterio de Nyquist:

fs = 2\*fmax=2\*150Hz=300Hz se adoptan 400Hz

DATOS PARA EL DISEÑO DEL FILTRO PROTOTIPO PARA LA NOTA MI:

* Frecuencia de corte, fc: 83,407Hz
* Frecuencia de atenuación, fa: 86,307 Hz
* Aproximación usada: Butterworth
* fs=400Hz
* Atenuación, : 4 dB (adoptada)

El primer paso para calcular nuestro filtro es calcular el orden “n” para que se cumplan las especificaciones, para ello usaremos la siguiente ecuación:

Podemos despejar n, quedando:

Reemplazamos los siguientes parámetros en la ecuación anterior

Obtenemos el orden del filtro:

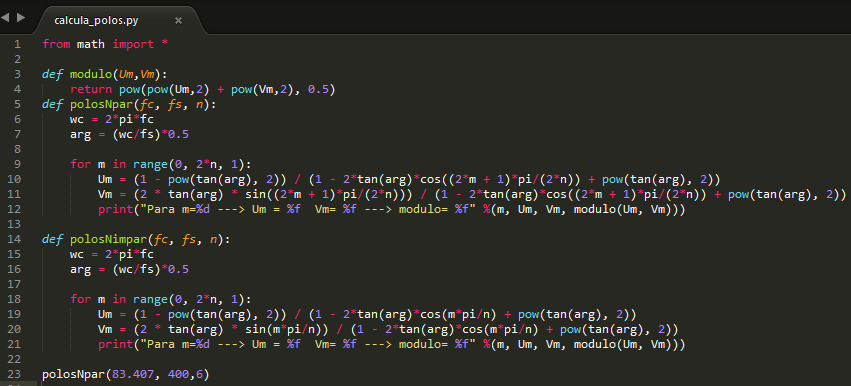
Como “n” debe ser un número entero, lo aproximamos a n= 6. Verificamos:

Como cumplen las especificaciones del problema, procedemos a realizar nuestro filtro Butterworth de orden n= 6.

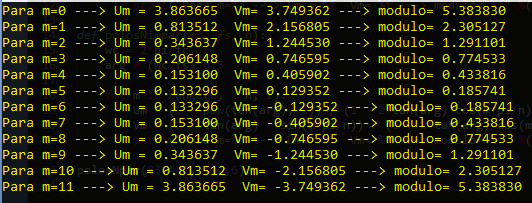
El método directo nos proporciona una serie de expresiones que nos entregan la posición de los ***polos*** del filtro prototipo que estamos diseñando, como n es par, usamos las siguientes expresiones calcular los polos:





Como el orden del filtro es 6 resulta en un excesivo cálculo analítico de la posición de los polos, por lo que decidimos escribir un script en lenguaje de programación Python para automatizar el cálculo de polos.

De esta manera **m** va de 0 a 2n-1, es decir de 0 a 11. Ejecutando el script se obtiene:



Se puede ver que hay polos cuyo modulo es mayor a 1 (polos para m=0, m=1, m=2, m=9, m=10, m=11) es decir, se encuentran fuera del círculo unitario por lo que llevarían a un sistema inestable, por lo tanto los polos de nuestra función son los correspondientes a m=3, m=4, m=5, m=6, m=7 y m=8:

Escribimos la función recordando que en el numerador tenemos n ceros en z=-1

Trabajando algebraicamente la expresión anterior con queda de la siguiente forma:

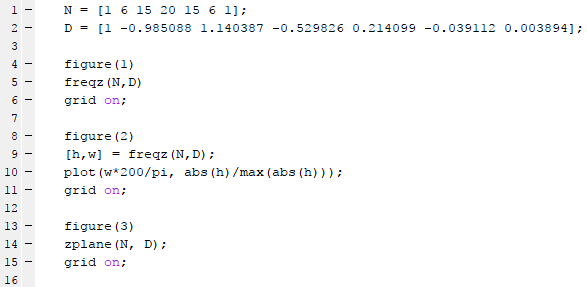
Ahora obtendremos , multiplicando y dividiendo por :

Esta es la función transferencia del filtro digital prototipo pasabajos Butterworth de orden n=6. Cuyos coeficientes fueron calculados de forma analítica:

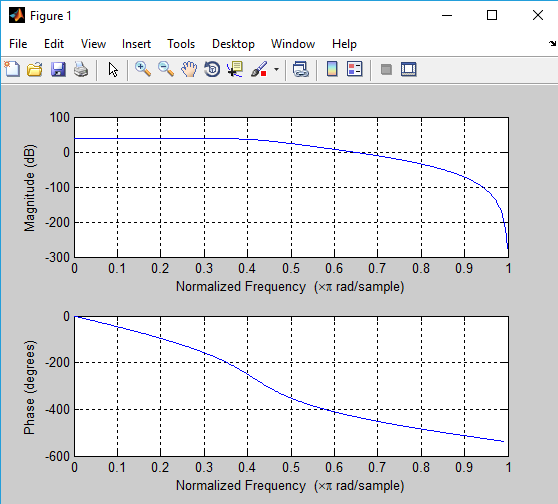
NUMERADOR:

DENOMINADOR:

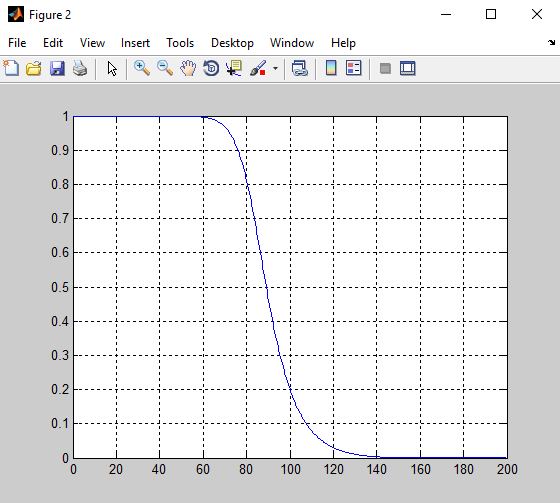
Para obtener las graficas del filtro prototipo cargamos los coeficientes obtenidos en Matlab y usamos el comando freqz (N,D) como se muestra en el siguiente script:



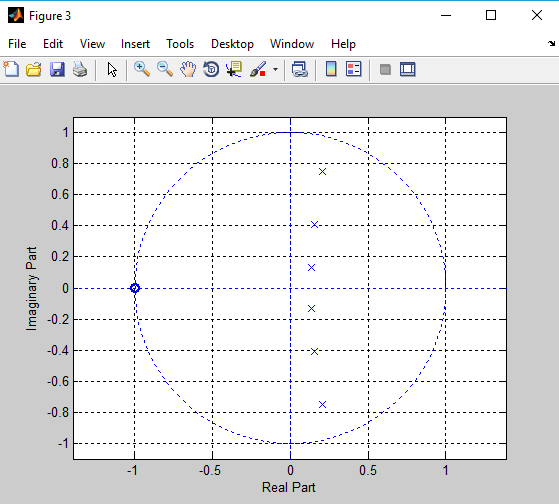
En la Figura 1, tanto la de magnitud como la de fase de la respuesta en frecuencia serán graficadas de 0 a π, ya que para valores más grandes, la gráfica se repite.



En la Figura 2, observamos la respuesta en amplitud del filtro pasabajo en función de la frecuencia.



En la Figura 3, se grafican los Polos y Ceros.



Ya obtenida la función transferencia del filtro prototipo digital pasabajos debemos hacer una transformación espectral para lograr así el filtro pasabanda que queríamos para la nota MI. Esta transformación se hace sustituyendo de por una función unitaria:

Donde los parámetros K y α están definidos por las siguientes expresiones:



**β:** es la frecuencia de corte del filtro pasabajo prototipo.

**ω0:** es la frecuencia central del pasabanda.

**ω1,2:** son las frecuencias de corte inferior y superior respectivamente del pasabanda.

Reemplazamos los siguientes parámetros en las ecs anteriores

rad/s

rad/s

rad/s

Obtenemos los siguientes resultados para α y K

Quedando esta función unitaria se reemplaza en todo de del filtro prototipo, obteniendo así la función del filtro digital pasabanda para la nota MI y va a tener la siguiente forma.

Este filtro pasabanda tiene orden 12. Todos estos procedimientos para encontrar la función transferencia del filtro pasabanda para la nota MI, se deben realizar para las 6 notas musicales restantes.

**DISEÑO EN MATLAB**

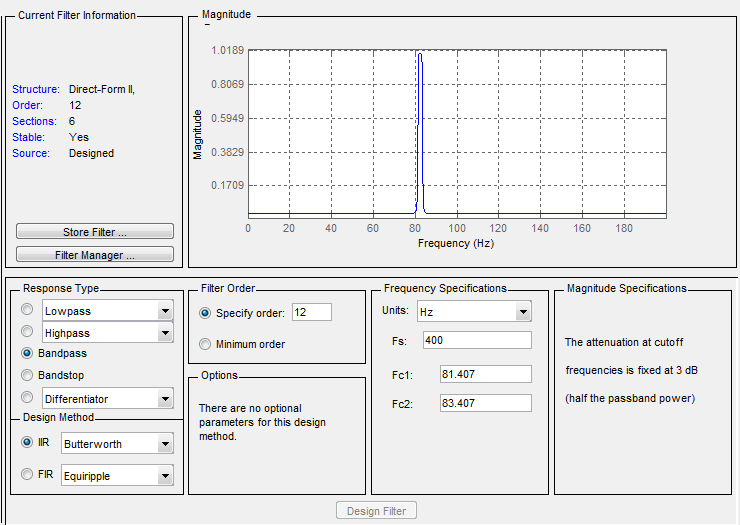
A continuación se procederá a usar la herramienta de matlab *fdatool* para obtener los 7 filtros correspondientes a cada nota utilizando el orden del filtro prototipo multiplicado por dos al ser un filtro pasabanda. Vale aclarar que con dicha herramienta, obtendremos primero el orden del filtro “MI”, ya que es el más exigente. Luego usaremos el mismo orden para el resto de los filtros.

###### FILTRO MI

Ingresamos al fdatool con los siguientes valores correspondientes a la nota “MI”, con las especificaciones que también se indicarán abajo:

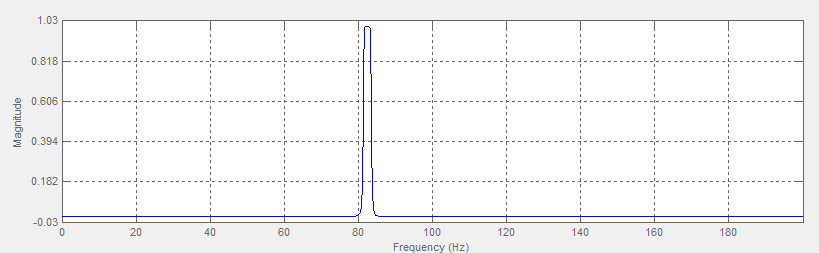
fc1= 81,407[Hz] fc2= 83,407[Hz]

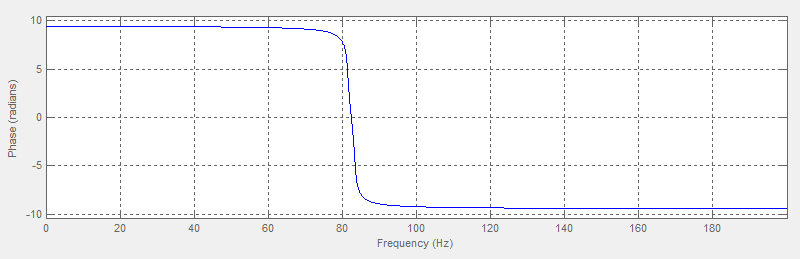
fsample=400[Hz] orden = 12



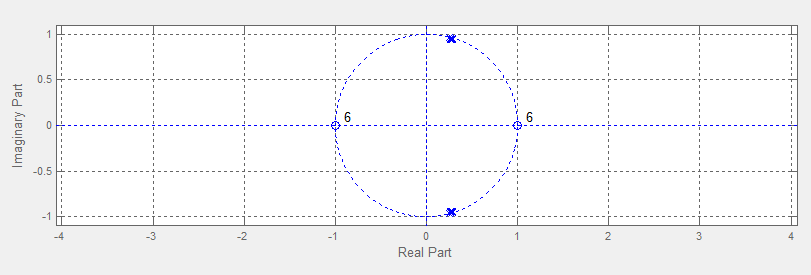
Como se puede observar, la herramienta nos da 6 secciónes para un orden de 12 que se obtuvo de multiplicar por dos el valor calculado en el filtro prototipo.

Magnitud y Fase

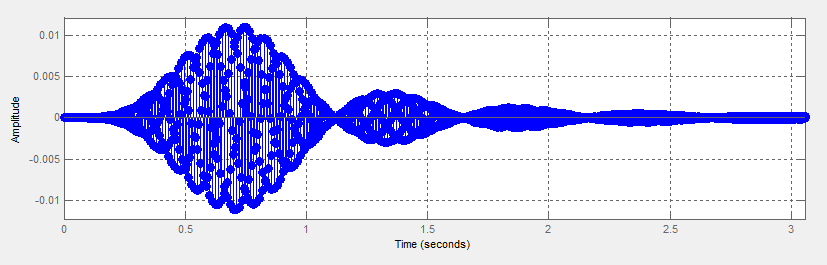




Polos y Ceros



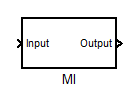
Respuesta al impulso

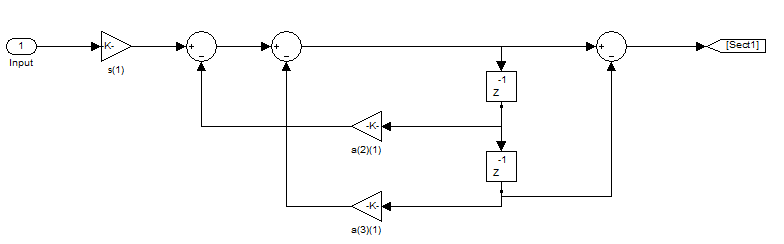


**REALIZACION FILTRO MI:**

Mostraremos la realización del filtro MI. Hay que aclarar que estamos mostrando UNA sección del filtro, cada uno de los filtros se forma por 6 secciones idénticas, lo único que cambia es la ganancia de los multiplicadores que detallaremos en una tabla más abajo.

Cuando presentemos las tablas se mostraran los valores de los coeficientes correspondientes a los denominadores de las funciones transferencia de cada sección. No incluimos los valores del numerador ya que es exactamente el mismo para todas las secciones, como se puede ver en el circuito de abajo este numerador tiene la siguiente forma: .





Coeficientes Filtro MI

|  |  |  |  |
| --- | --- | --- | --- |
|  | Sección 1 | Sección 2 | Sección 3 |
| s | 0.015643856081688782 | 0.015643856081688782 | 0.015535734583275557 |
| a2 | -0.57251016928138632 | -0.51434089051356646 | -0.56085593977015735 |
| a3 | 0.99193778114531972 | 0.99186837273413786 | 0.9780989715895605 |

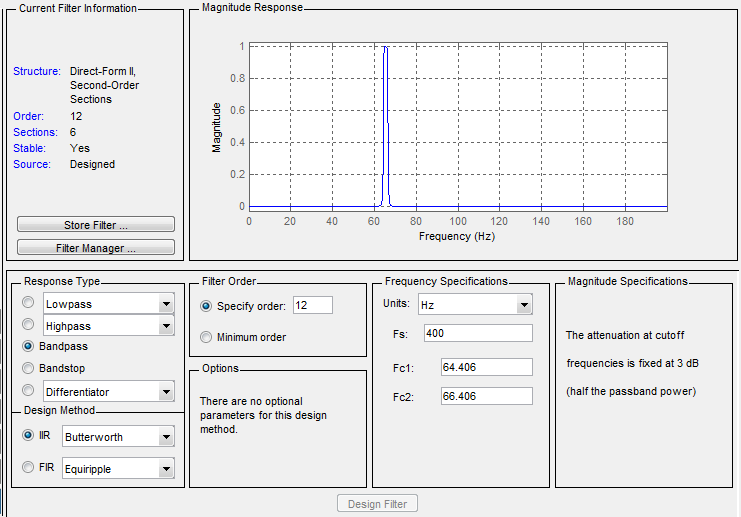
|  |  |  |  |
| --- | --- | --- | --- |
|  | Sección 4 | Sección 5 | Sección 6 |
| s | 0.015535734583275557 | 0.015474322344596165 | 0.015474322344596165 |
| a2 | -0.51854153016054361 | -0.54530978396371821 | -0.52987711957735661 |
| a3 | 0.97796205117839285 | 0.97014011648529264 | 0.970072188608057 |

###### FILTRO DO

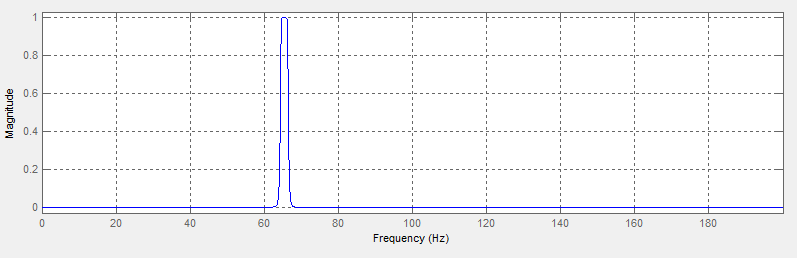
Para el filtro Do usaremos las mismas caracteristicas de atenuacion pero ajustando las frecuencias . Se puede observar que tanto en este como en los demas filtros se sigue obteniendo 6 secciones para un orden de 12.

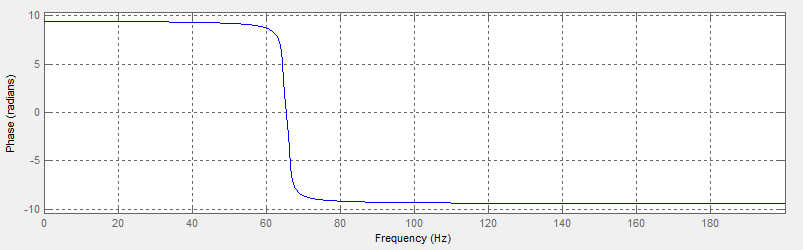
fc1= 64,406[Hz] fc2= 66,406[Hz]

fsample=400[Hz] orden = 12

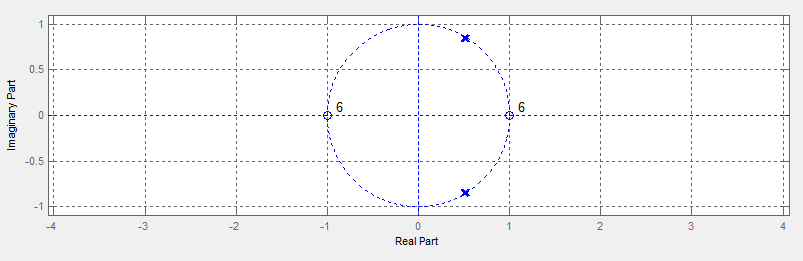


Magnitud y Fase

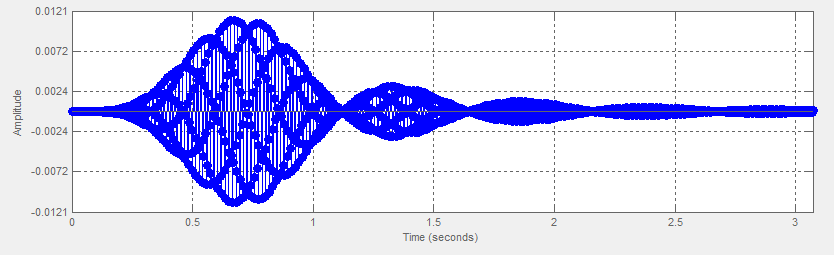




Polos y Ceros



Respuesta al impulso



Hay que tener en cuenta que cada uno de los 7 filtros son idénticos ya que todos fueron diseñados a partir del filtro MI entonces mostraremos una tabla con los valores de los multiplicadores correspondientes a cada filtro porque eso es lo que si cambia entre filtro y filtro.

Coeficientes Filtro DO

|  |  |  |  |
| --- | --- | --- | --- |
|  | Sección 1 | Sección 2 | Sección 3 |
| s | 0.015643856081688716 | 0.015643856081688716 | 0.015535734583275491 |
| a2 | -1.0038982753588384 | -1.0557118085016397 | -1.0416252848327254 |
| a3 | 0.99182916124117371 | 0.99197699693252472 | 0.97817634036191103 |

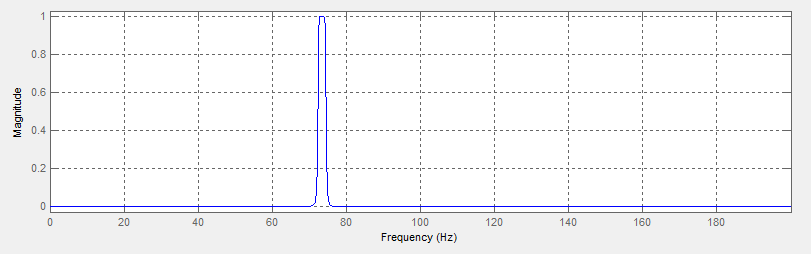
|  |  |  |  |
| --- | --- | --- | --- |
|  | Sección 4 | Sección 5 | Sección 6 |
| s | 0.015535734583275491 | 0.015474322344596102 | 0.015474322344596102 |
| a2 | -1.0038600588372544 | -1.0256475911629499 | -1.0118586233368561 |
| a3 | 0.97788469935522748 | 0.97017850017126328 | 0.97003380912814663 |

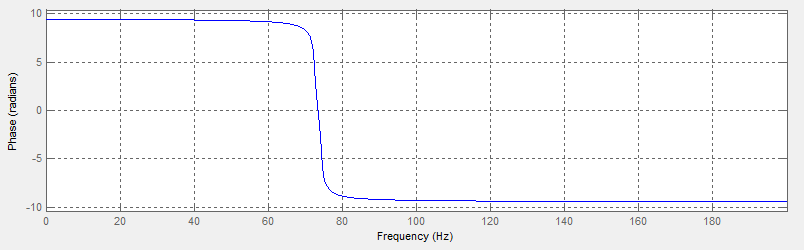
###### FILTRO RE

fc1= 72,416[Hz] fc2=74,416[Hz]

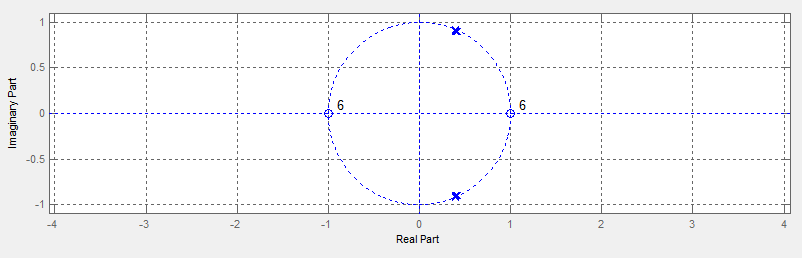
fsample=400[Hz] orden = 12

Magnitud y Fase

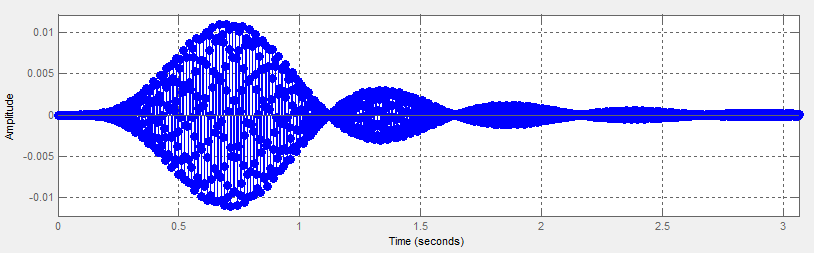




Polos y Ceros



Respuesta al impulso



Coeficientes Filtro RE

|  |  |  |  |
| --- | --- | --- | --- |
|  | Sección 1 | Sección 2 | Sección 3 |
| s | 0.015643856081688595 | 0.015643856081688595 | 0.015535734583275368 |
| a2 | -0.83537780721957589 | -0.78008381502027802 | -0.82231904693126312 |
| a3 | 0.99195736907053333 | 0.99184878656630682 | 0.97813761531579158 |

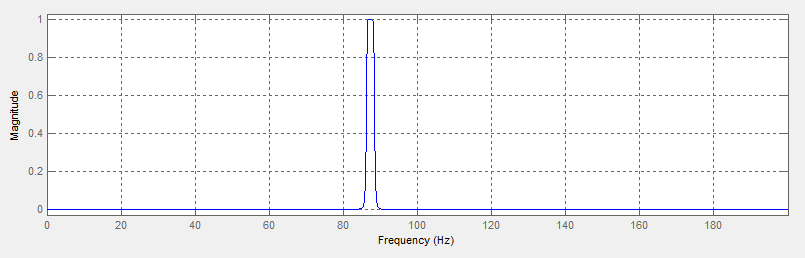
|  |  |  |  |
| --- | --- | --- | --- |
|  | Sección 4 | Sección 5 | Sección 6 |
| s | 0.015535734583275368 | 0.015474322344595979 | 0.015474322344595979 |
| a2 | -0.7820637910859557 | -0.80640639934753611 | -0.79171798370565172 |
| a3 | 0.97792341438825503 | 0.97015928790813177 | 0.97005301890639706 |

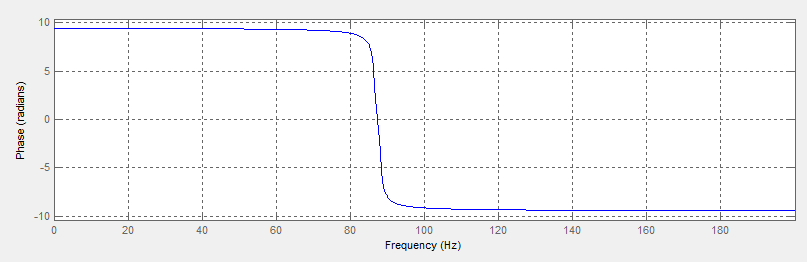
###### FILTRO FA

fc1= 86,307[Hz] fc2=88,307[Hz]

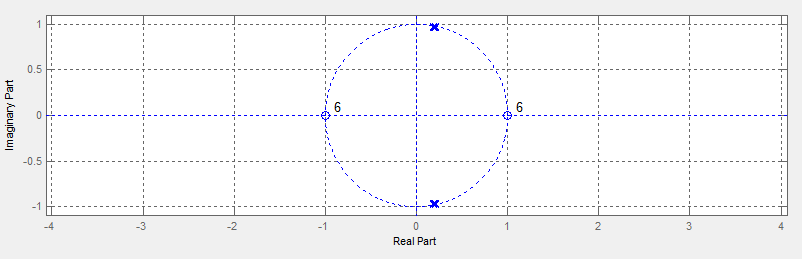
fsample=400[Hz] orden = 12

Magnitud y Fase

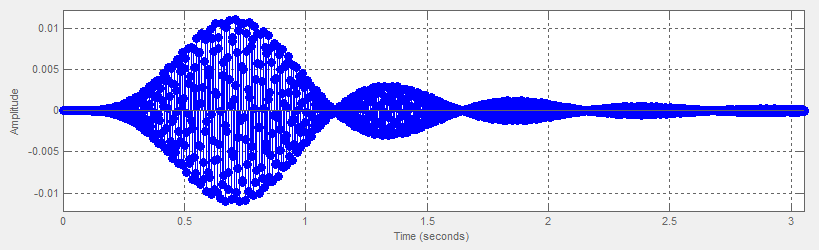




Polos y Ceros



Respuesta al impulso



Coeficientes Filtro FA

|  |  |  |  |
| --- | --- | --- | --- |
|  | Sección 1 | Sección 2 | Sección 3 |
| s | 0.015643856081688799 | 0.015643856081688799 | 0.015535734583275573 |
| a2 | -0.42410816737576007 | -0.36485208625618137 | -0.41332137356227322 |
| a3 | 0.99192780287930682 | 0.99187835040230987 | 0.97807928689220003 |

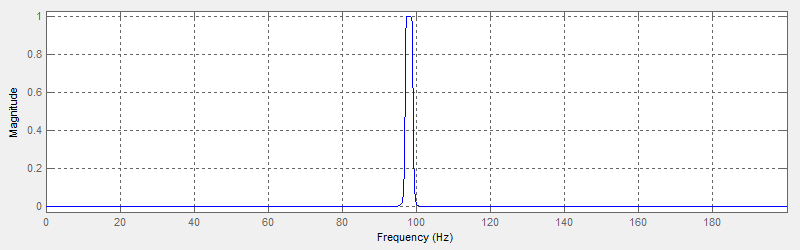
|  |  |  |  |
| --- | --- | --- | --- |
|  | Sección 4 | Sección 5 | Sección 6 |
| s | 0.015535734583275573 | 0.015474322344596182 | 0.015474322344596182 |
| a2 | -0.37022821584224752 | -0.39810365110236468 | -0.38238942977989177 |
| a3 | 0.97798173351627704 | 0.97013035085814869 | 0.97008195364971961 |

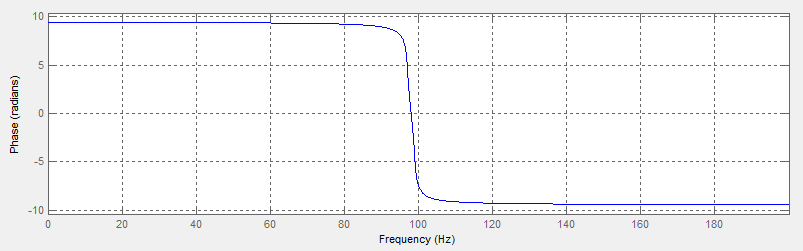
###### FILTRO SOL

fc1= 96,999 [Hz] fc2=98,999 [Hz]

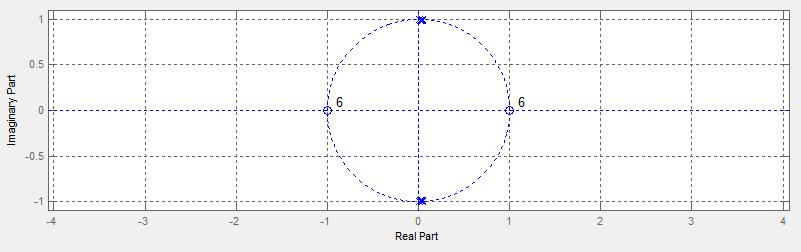
fsample=400[Hz] orden = 12

Magnitud y Fase

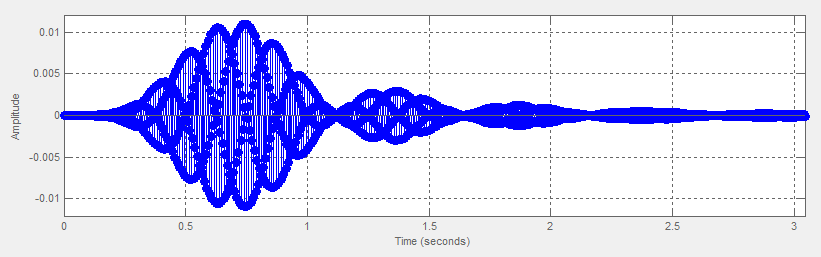




Polos y Ceros



Respuesta al impulso



Coeficientes Filtro SOL

|  |  |  |  |
| --- | --- | --- | --- |
|  | Sección 1 | Sección 2 | Sección 3 |
| s | 0.015643856081688806 | 0.015643856081688806 | 0.015535734583275581 |
| a2 | -0.092798883172934166 | -0.03238477859242761 | -0.084124165828006137 |
| a3 | 0.9919069238064363 | 0.99189922887373028 | 0.97803809876327696 |

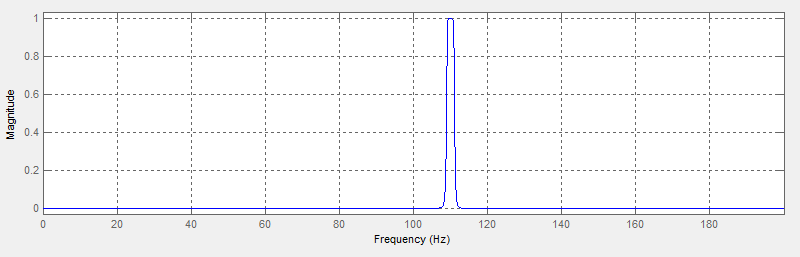
|  |  |  |  |
| --- | --- | --- | --- |
|  | Sección 4 | Sección 5 | Sección 6 |
| s | 0.015535734583275581 | 0.015474322344596189 | 0.015474322344596189 |
| a2 | -0.040200990382160129 | -0.069927273688859315 | -0.053912908877486204 |
| a3 | 0.97802291927149687 | 0.97010991729246976 | 0.97010238662640225 |

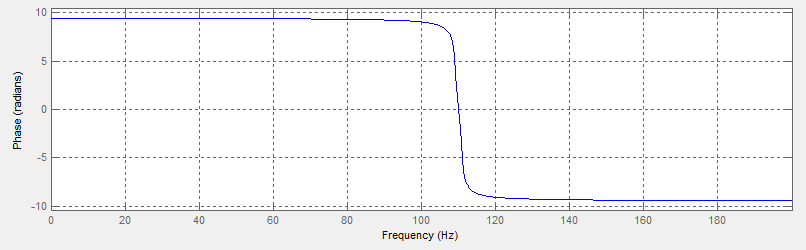
###### FILTRO LA

fc1= 109[Hz] fc2=111 [Hz]

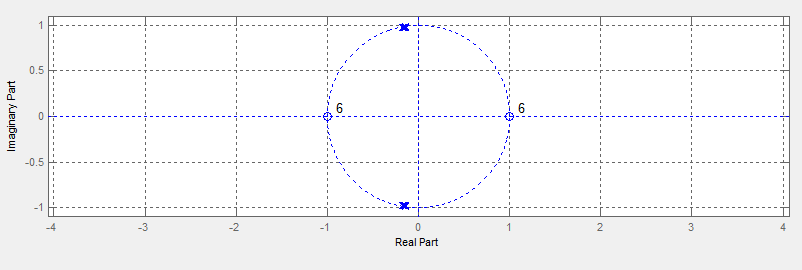
fsample=400[Hz] orden = 12

Magnitud y Fase

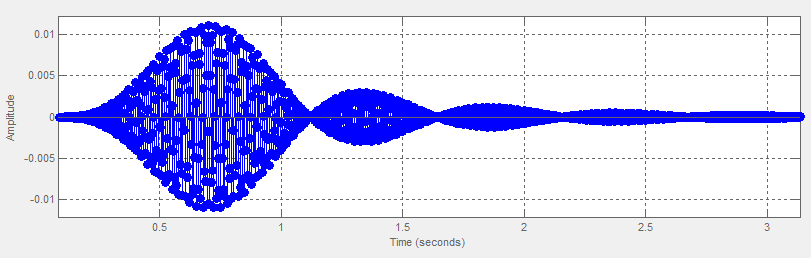




Polos y Ceros



Respuesta al impulso



Coeficientes filtro LA

|  |  |  |  |
| --- | --- | --- | --- |
|  | Sección 1 | Sección 2 | Sección 3 |
| s | 0.015643856081688581 | 0.015643856081688581 | 0.015535734583275356 |
| a2 | 0.3414220984166767 | 0.28171647985766884 | 0.33114022551010275 |
| a3 | 0.99192245758077136 | 0.99188369546316335 | 0.97806874208776051 |

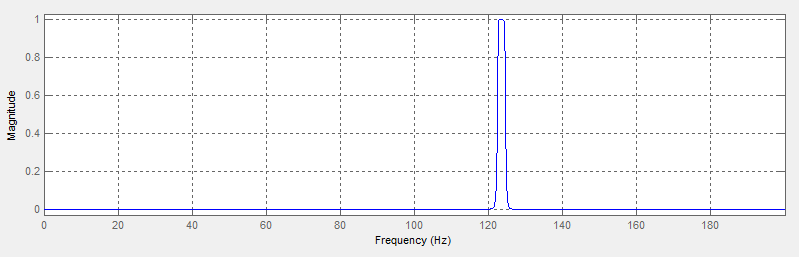
|  |  |  |  |
| --- | --- | --- | --- |
|  | Sección 4 | Sección 5 | Sección 6 |
| s | 0.015535734583275356 | 0.015474322344595965 | 0.015474322344595965 |
| a2 | 0.28772488829163234 | 0.31614086675989106 | 0.30031014776843012 |
| a3 | 0.97799227738265704 | 0.97012511955926017 | 0.9700871847158421 |

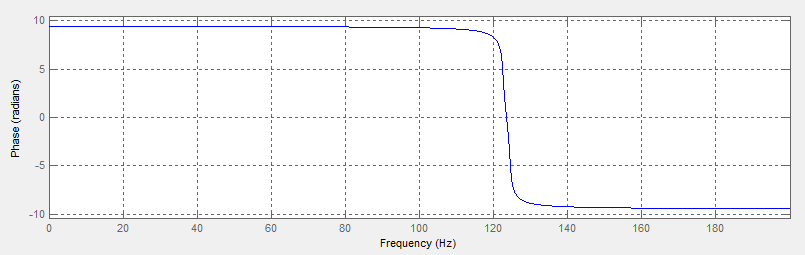
**FILTRO SI**

fc1= 122,471[Hz] fc2=124,471 [Hz]

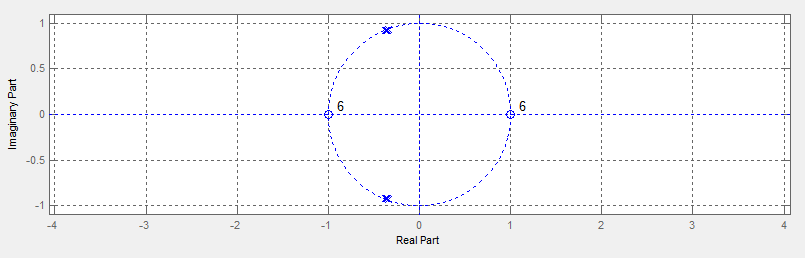
fsample=400[Hz] orden = 12

Magnitud y Fase

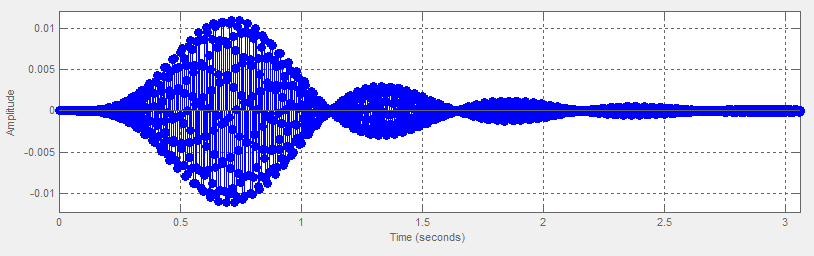
****



Polos y Ceros



Respuesta al Impulso



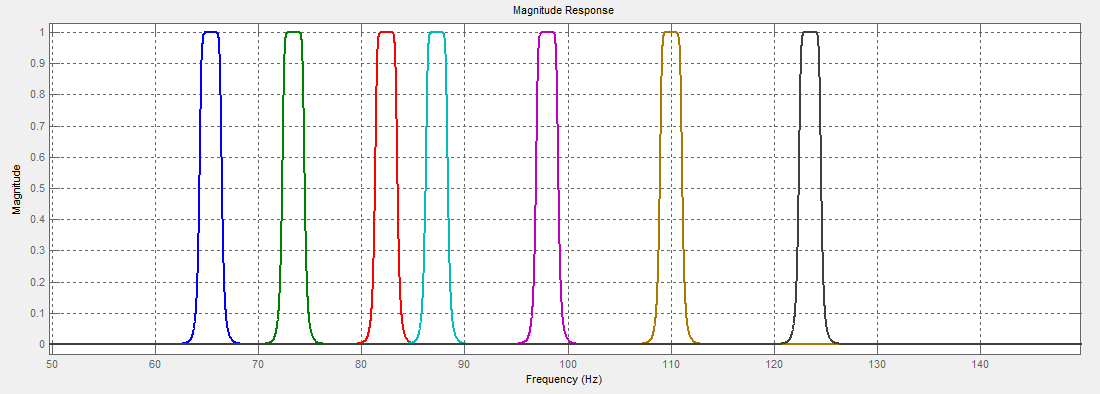
Coeficientes Filtro SI

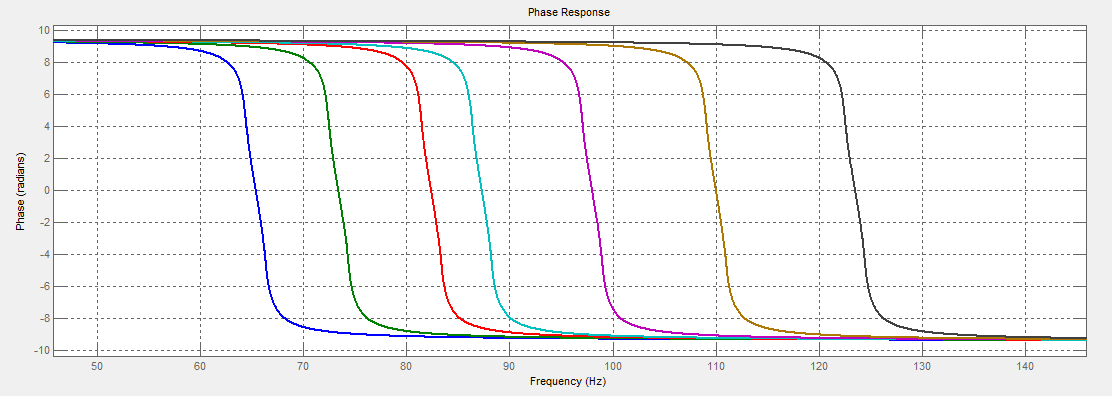
|  |  |  |  |
| --- | --- | --- | --- |
|  | Sección 1 | Sección 2 | Sección 3 |
| s | 0.015643856081688549 | 0.015643856081688549 | 0.015535734583275325 |
| a2 | 0.74598592339027081 | 0.68956975279007149 | 0.73338469311749588 |
| a3 | 0.99195035377795737 | 0.99185580114058536 | 0.97812377505060311 |

|  |  |  |  |
| --- | --- | --- | --- |
|  | Sección 4 | Sección 5 | Sección 6 |
| s | 0.015535734583275325 | 0.015474322344595936 | 0.015474322344595936 |
| a2 | 0.69232598807378132 | 0.71756398273395594 | 0.70258521657247819 |
| a3 | 0.97793725181837665 | 0.97015242161528381 | 0.97005988449572 |

###### COMPARACIÓN

A continuación, mostramos los 7 filtros en un mismo gráfico, para comprobar que no existen solapamientos entre ellos y, en especial, entre los filtros correspondientes a las notas “Mi” y “Fa”, que son las de frecuencias más cercanas. No debe existir solapamiento por encima de una magnitud de 0.707 para que el sistema de filtrado sea correcto.



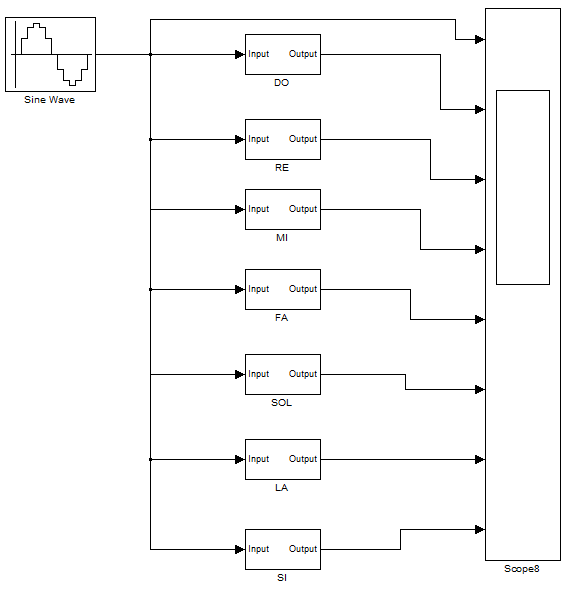


**OBSERVACIÓN**

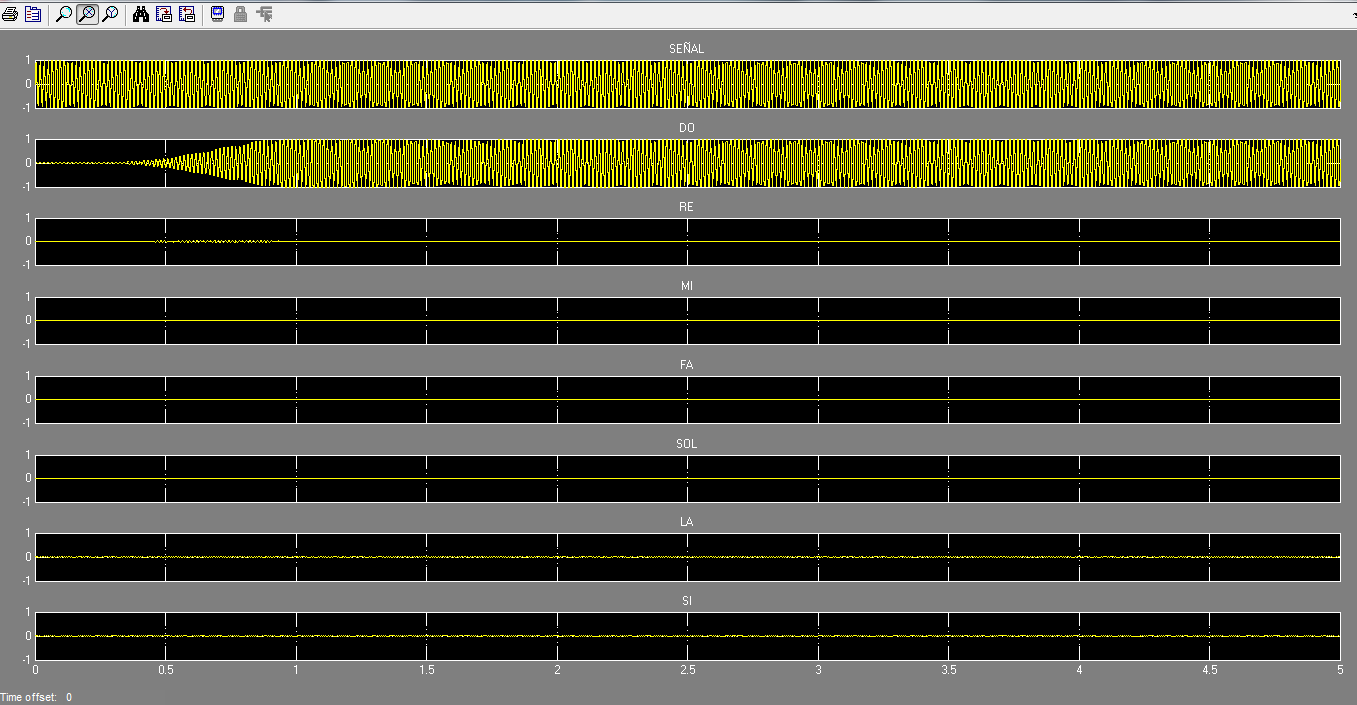
Como se puede ver colocamos la respuesta al impulso de cada filtro ya que esto demuestra la condicion de estabilidad de cada filtro porque en todo caso la respuesta se atenua (sistemas estables).

#### SIMULACIONES

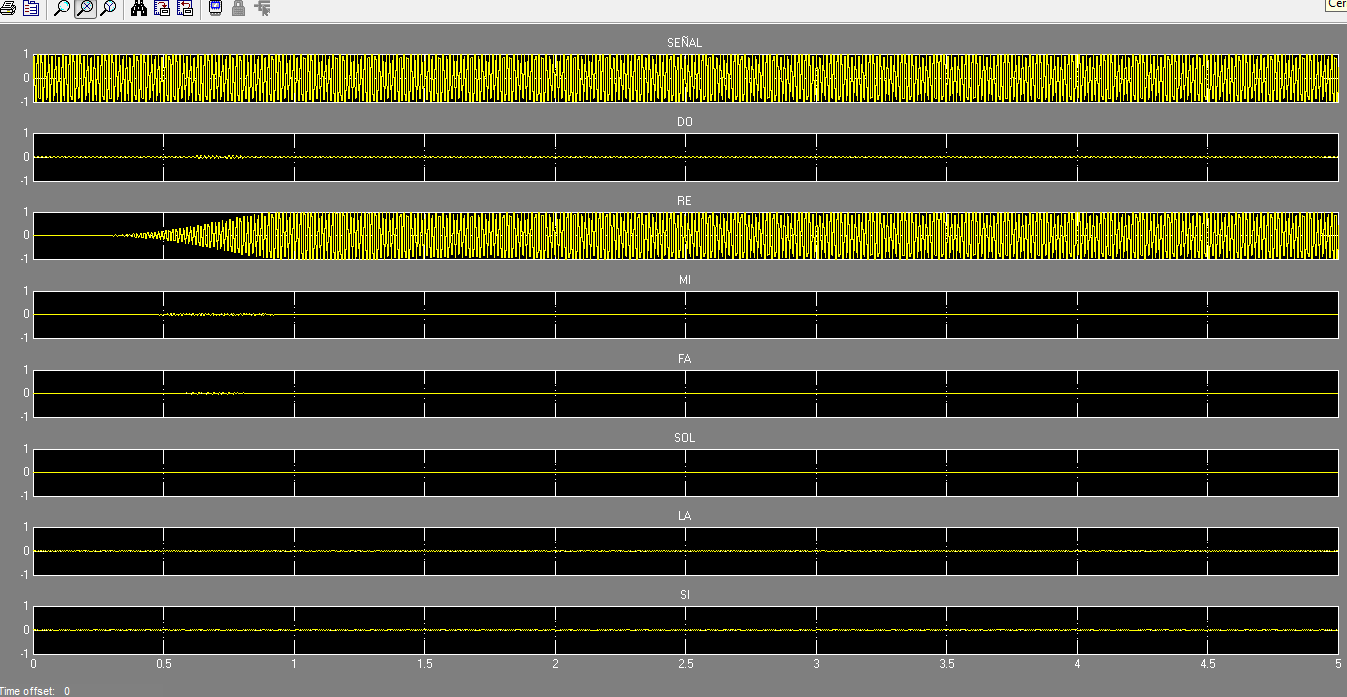
A continuación se utilizara la herramienta Simulink provista por Matlab para realizar la simulación del banco de filtros a partir del siguiente circuito formado por un generador senoidal discretizado de frecuencia variable (colocaremos las frecuencias de cada nota musical), los 7 filtros diseñados anteriormente y un osciloscopio de 8 canales donde veremos la salida de cada uno de los filtros y la entrada que viene del generador:

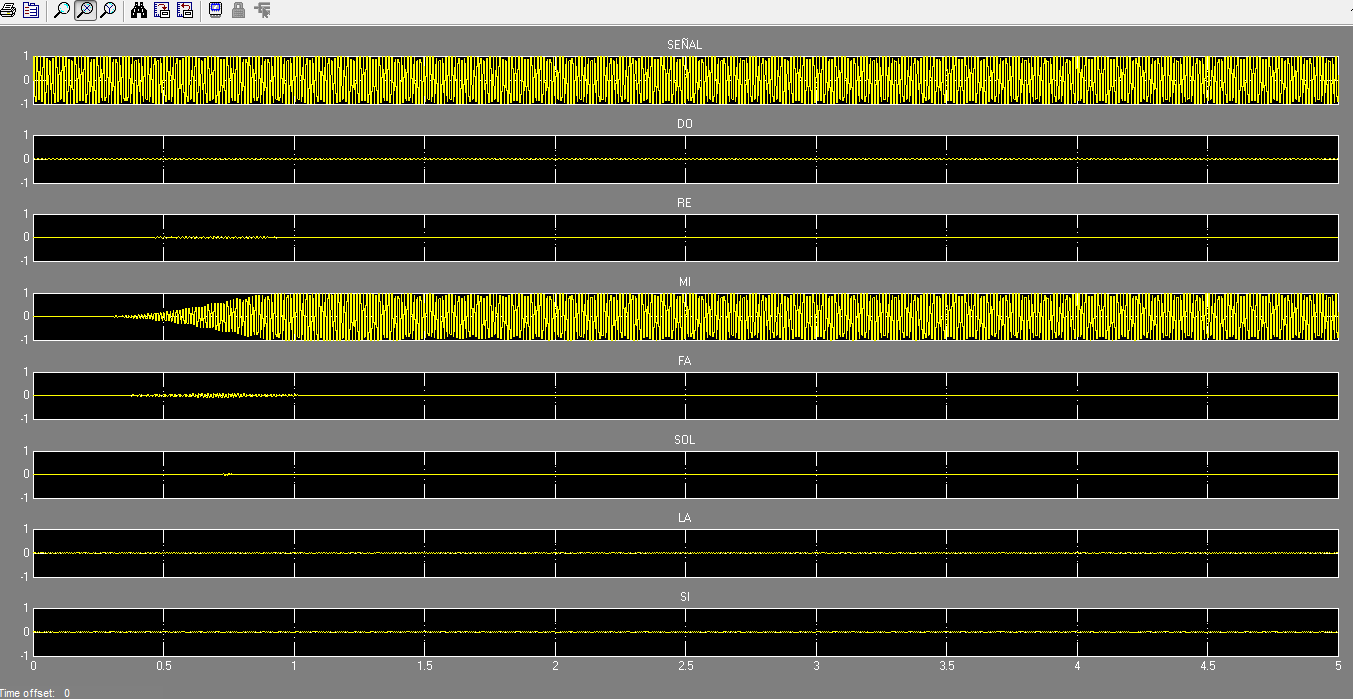


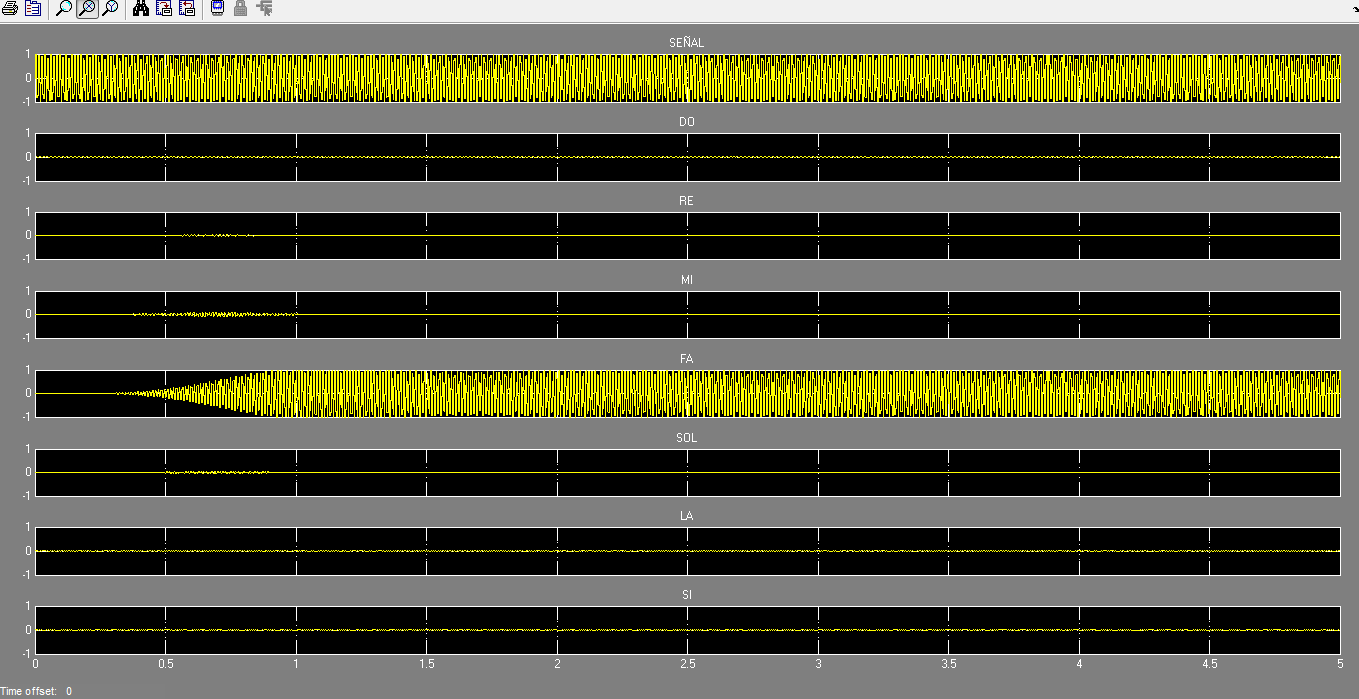
* Nota DO (65,406 Hz):



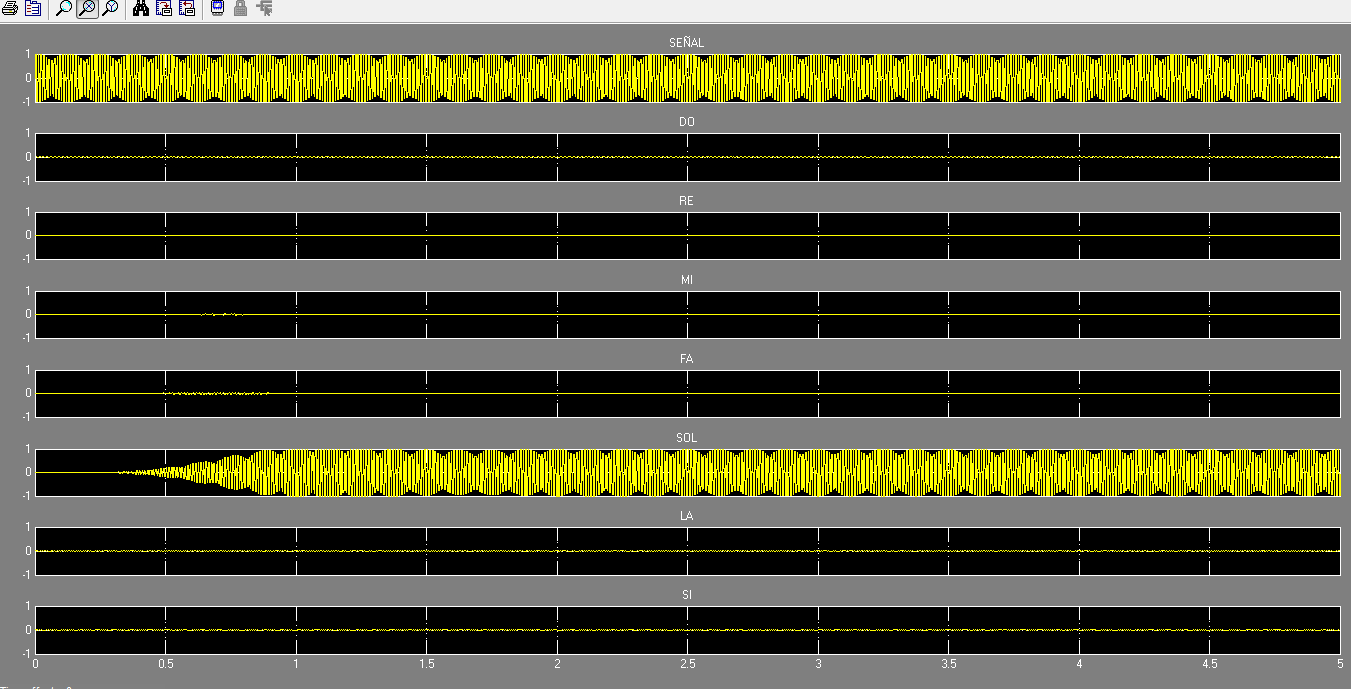
* Nota RE (73,416 Hz):



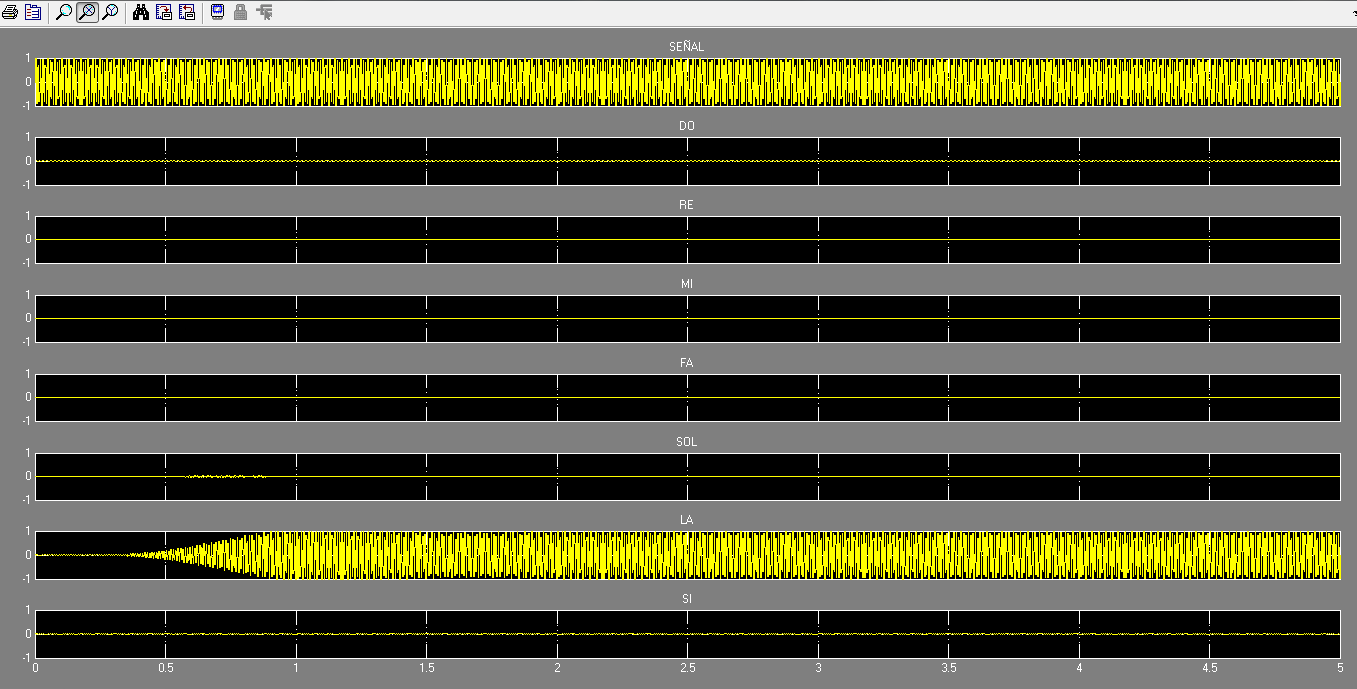
* Nota MI (82,407 Hz): 
* Nota FA (87,307 Hz):



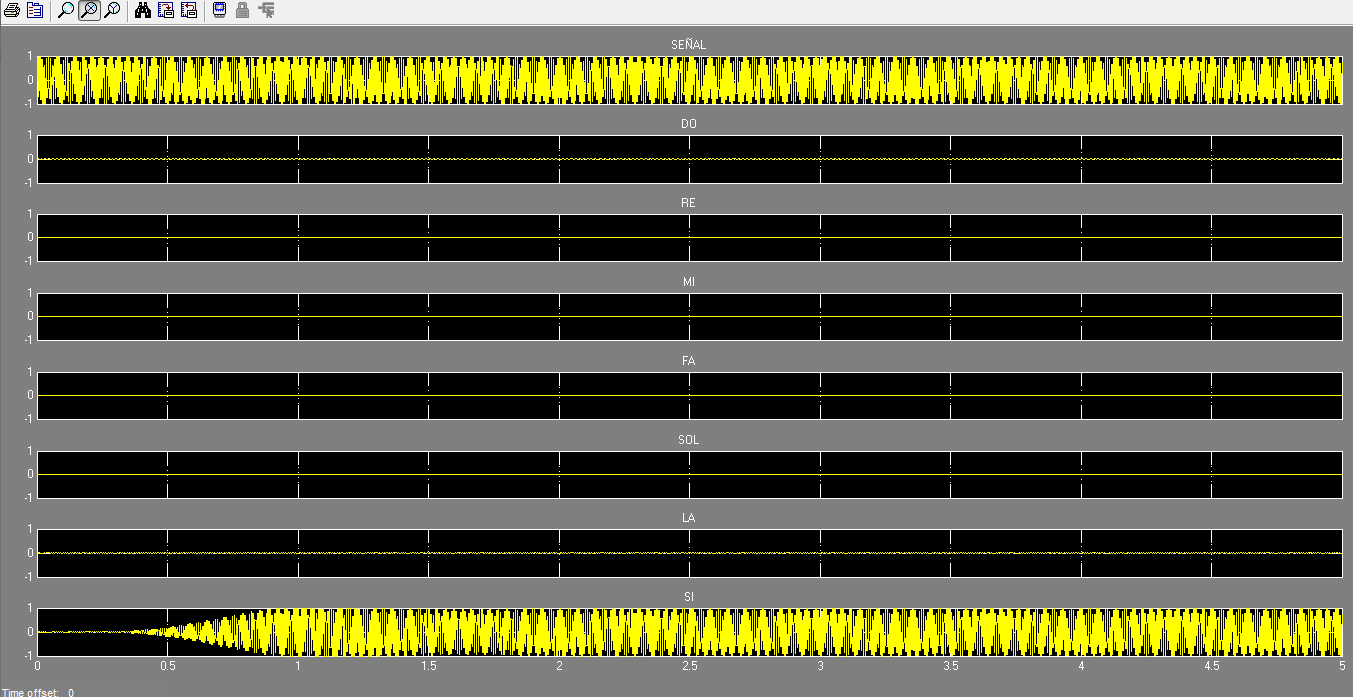
* Nota SOL (97,999 Hz):



* Nota LA (110 Hz):



* Nota SI (123,471 Hz):



**CONCLUSIONES:**

* Como se puede apreciar en las simulaciones solo tenemos señal a la salida de los filtros cuya frecuencia central coincide con la del generador mostrando el buen desempeño del banco de filtros. También notamos que de acuerdo a nuestros cálculos del filtro prototipo, nos da como resultado un banco de filtros pasabandas muy selectivo, esto se debe a que no implementamos los filtros con el mínimo orden calculado por la herramienta fdatool, si no con el orden calculado analíticamente. Notar que en algunos filtros aparece una pequeña perturbación que pudimos medir usando el Simulink y encontramos que se trata de una señal cuya amplitud es igual al 5% de la entrada, esto corresponde a un fenómeno transitorio ya que hay que recordar que la excitación de entrada es una senoidal pero que esta discretizada así que tiene contenido armónico entonces puede tener frecuencias que pasarían por alguno de los filtros en los que no deberíamos ver nada, aunque el efecto es casi indetectable así que no los consideramos un problema.
* En el principio del informe, a la hora de diseñar el filtro, detallábamos el problema de la proximidad en frecuencia de las notas MI y FA y dijimos que si diseñábamos todo en base al filtro MI el banco se comportaría como se desea. Esto se verifica viendo las simulaciones que como se puede observar casi ni se solapan debido al diseño de un filtro muy selectivo.
* Con este proyecto afianzamos los conocimientos aprendidos en la materia sobre el diseño de filtros digitales y lo más importante pudimos utilizar la herramienta de Matlab, fdatool. Esta herramienta permite diseñar casi cualquier filtro siempre que el orden no sea muy alto, en nuestro caso necesitábamos filtros muy selectivos y aun así el fdatool nos dio un orden mínimo de 2 y con esto ya cumplía todas las especificaciones que necesitábamos.

#### Bibliografía y herramientas utilizadas:

Rabiner, Lawrence R, and Bernard Gold. “*Theory And Application Of Digital Signal Processing”*. Englewood Cliffs, N.J.: Prentice-Hall, 1975. Print.

Ingle, V., Proakis J.G. *“Digital Signal Processing Using Matlab”*.  Brooks/Cole Thomson Learning. 2000.

Stanley, W., *“Digital Signal Processing”*, Reston Pub., Reston, Virginia, 1975.

Apuntes de clases.

Matlab versión 2015.