



Instituto Politécnico Nacional

Escuela Superior de Cómputo

Proyecto 3

MatLab

INSTITUTO POLITÉCNICO NACIONAL



Materia:

Teoría de Señales y Comunicaciones

Grupo:

3CV14

Profesor:

Alfonso Fernández Vázquez

Alumno:

Castro Cruces Jorge Eduardo

Boleta:

2015080213

Fecha:

Viernes, 10 de diciembre de 2021

Contenido

A.	Project 7.1 Estimation of IIR Filter Order	5
•	Questions:.....	5
1)	Q7.1 Using MATLAB determine the lowest order of a digital IIR lowpass filter of all four types. The specifications are as follows: sampling rate of 40 kHz, passband edge frequency of 4 kHz, stopband edge frequency of 8 kHz, passband ripple of 0.5 dB, and a minimum stopband attenuation of 40 dB. Comment on your results.....	5
2)	Q7.2 Using MATLAB determine the lowest order of a digital IIR highpass filter of all four types. The specifications are as follows: sampling rate of 3,500 Hz, passband edge frequency of 1,050 Hz, stopband edge frequency of 600 Hz, passband ripple of 1 dB and a minimum stopband attenuation of 50 dB. Comment on your results.....	6
3)	Q7.3 Using MATLAB determine the lowest order of a digital IIR bandpass filter of all four types. The specifications are as follows: sampling rate of 7 kHz, passband edge frequencies at 1.4 kHz and 2.1 kHz, stopband edge frequencies at 1.05 kHz and 2.45 kHz, passband ripple of 0.4 dB, and a minimum stopband attenuation of 50 dB. Comment on your results.....	7
4)	Q7.4 Using MATLAB determine the lowest order of a digital IIR bandstop filter of all four types. The specifications are as follows: sampling rate of 12 kHz, passband edge frequencies at 2.1 kHz and 4.5 kHz, stopband edge frequencies at 2.7 kHz and 3.9 kHz, passband ripple of 0.6 dB, and a minimum stopband attenuation of 45 dB. Comment on your results.....	8
B.	Project 7.2 IIR Filter Design	10
•	Questions:.....	11
5)	Q7.5 Design the Butterworth bandstop filter by running Program P7 1. Write down the exact expression for the transfer function generated. What are the filter specifications? Does your design meet the specifications? Using MATLAB, compute and plot the filter's unwrapped phase response and the group delay response.....	11
6)	Q7.6 Modify Program P7 1 to design a Type 1 Chebyshev lowpass filter meeting the specifications given in Question Q7.1. Write down the exact expression for the transfer function generated. Does your design meet the specifications? Using MATLAB, compute and plot the filter's unwrapped phase response and the group delay response.....	13
7)	Q7.7 Modify Program P7 1 to design a Type 2 Chebyshev highpass filter meeting the specifications given in Question Q7.2. Write down the exact expression for the transfer function generated. Does your design meet the specifications? Using MATLAB, compute and plot the filter's unwrapped phase response and the group delay response.....	17
8)	Q7.8 Modify Program P7 1 to design an elliptic bandpass filter meeting the specifications given in Question Q7.3. Write down the exact expression for the transfer function generated. Does your design meet the specifications? Using MATLAB, compute and plot the filter's unwrapped phase response and the group delay response.....	20
C.	Project 7.3 Gibb's Phenomenon	25
•	Questions:.....	25
9)	Q7.9 Using the function sinc write a MATLAB program to generate the impulse response coefficients of four zero-phase lowpass filters with cutoffs at $\omega_c = 0.4\pi$ and of lengths 81, 61, 41, and 21, respectively, and then compute and plot their magnitude responses. Use the colon ":" operator to extract the impulse response coefficients of the shorter length filters from that of the length-81 filter. Examine the oscillatory behavior of the frequency responses of each filter on both sides of the cutoff frequency. What is the relation between the number of ripples and the length of the filter? What is the relation between the heights of the largest ripples and the length of the filter? How would you modify the above program to generate the impulse response coefficients of a zero-phase lowpass filter of even lengths?.....	25
10)	Q7.10 Using the function sinc write a MATLAB program to generate the impulse response	

coefficients of a zero-phase length-45 highpass filter with a cutoff at $\omega_c = 0.4\pi$ and then compute and plot its magnitude response. Examine the oscillatory behavior of the frequency responses of each filter on both sides of the cutoff frequency. How would you modify the above program to generate the impulse response coefficients of a zero-phase highpass filter of even length?.....	29
11) Q7.11 Write a MATLAB program to generate the impulse response coefficients of four zero-phase differentiators of lengths 81, 61, 41, and 21, respectively, and then compute and plot their magnitude responses. The following code fragments show how to generate a differentiator of length $2M+1$	30
12) Q7.12 Write a MATLAB program to generate the impulse response coefficients of four discrete-time Hilbert transformers of lengths 81, 61, 41, and 21, respectively, and then compute and plot their magnitude responses. The following code fragments show how to generate a Hilbert transformer of length $2M+1$	33
D. Project 7.4 Estimation of Order of FIR Filter.....	37
• Questions:.....	37
13) Q7.13 Using the function kaiord, estimate the order of a linear-phase lowpass FIR filter with the following specifications: passband edge = 2 kHz, stopband edge = 2.5 kHz passband ripple $\delta p = 0.005$, stopband ripple $\delta s = 0.005$, and sampling rate of 10 kHz. What are the purposes of the commands ceil and nargin in the function kaiord?	37
14) Q7.14 Repeat Question Q7.13 for the following cases: (a) sampling rate of 20 kHz, (b) $\delta p = 0.002$ and $\delta s = 0.002$, and (c) stopband edge = 2.3 kHz. Compare the filter length obtained in each case with that obtained in Question Q7.13. Comment on the effect of the sampling rate, ripples, and the transition bandwidth on the filter order.	37
15) Q7.15 Repeat Question Q7.13 using the function kaiserord. Compare the value of the filter order obtained with that obtained in Question Q7.13.	37
16) Q7.16 Repeat Question Q7.13 using the function firpmord. Compare the value of the filter order obtained with those obtained in Questions Q7.13 and Q7.15.....	38
17) Q7.17 Using the function kaiord, estimate the order of a linear-phase bandpass FIR filter with the following specifications: passband edges = 1.8 and 3.6 kHz, stopband edges 1.2 and 4.2 kHz, passband ripple $\delta p = 0.1$, stopband ripple $\delta s = 0.02$, and sampling rate of 12 kHz.	38
18) Q7.18 Repeat Question Q7.17 using the function kaiserord. Compare the value of the filter order obtained with that obtained in Question Q7.17.	38
19) Q7.19 Repeat Question Q7.17 using the function firpmord. Compare	39
E. Project 7.5 FIR Filter Design.....	40
• Questions:.....	40
20) Q7.20 Using the function fir1, design a linear-phase FIR lowpass filter meeting the specifications given in Question Q7.13 and plot its gain and phase responses. Use the order estimated using Kaiser's formula in Question Q7.13. Show the filter coefficients in a tabular form. Does your design meet the specifications? If it does not, adjust the filter order until the design meets the specifications. What is the order of the filter meeting the specifications?	40
21) Q7.21 Repeat Question Q7.20 using each of the following windows: Hanning, Blackman, and Dolph–Chebyshev windows.	45
22) Q7.22 Repeat Question Q7.20 using the function firpm.	59
23) Q7.23 Design an FIR lowpass filter using a Kaiser window. The filter specifications are: $\omega_p = 0.3\pi$, $\omega_s = 0.4\pi$, and $A_s = 50$ dB. Note that the function kaiser requires the values of the parameter β and the order N which must first be calculated using Eqs. (7.36) and (7.37), respectively. Does your design meet the specifications?	65
24) Q7.24 Repeat Question Q7.23 using the functions kaiserord and fir1.	68
25) Q7.25 Using fir2 design an FIR filter of order 95 with three different constant magnitude levels: 0.4 in the frequency range 0 to 0.25, 1.0 in the frequency range 0.3 to 0.45, and 0.8 in the frequency	

range 0.5 to 1.0. Plot the magnitude response of the filter designed. Does your design meet the specifications? 71

26) Q7.26 Using firpm, design the FIR bandpass filter with specifications given in Question Q7.17 and of order obtained using kaiserord. The vector wgts needed in the function firpm is given by 73

27) Q7.27 Using firpm, design an FIR bandpass filter with the following specifications: passband edges = 1.8 and 3.0 kHz, stopband edges = 1.5 and 4.2 kHz, passband ripple $\delta_p = 0.1$, stopband ripple $\delta_p = 0.02$, and sampling rate of 12 kHz. Estimate the filter order using kaiserord. Is your design an optimal FIR filter? Does your design meet the specifications? If it does not, does increasing the filter order help in meeting the specifications? Are the specifications met by a filter with a lower order than that obtained using kaiserord? In the case of unequal transition bands, the filter designed using firpm may exhibit unsatisfactory behavior in the larger transition bandwidth in its gain response. One way to improve the behavior is to reduce the transition band by moving the stopband edge until the design meets the specifications with smooth roll-off in the transition bands. Try this approach and determine the new specifications, with passband edges remaining fixed, that provide smooth roll-off in the transition bands.

A. Project 7.1 Estimation of IIR Filter Order

El primer paso en el proceso de diseño del filtro es elegir el tipo de aproximación del filtro que se utilizará y luego estimar el orden de la función de transferencia a partir de las especificaciones del filtro. El comando MATLAB para estimar el orden de un filtro de Butterworth es

$$[N, Wn] = buttord (Wp, Ws, Rp, Rs),$$

donde los parámetros de entrada son la frecuencia de borde de banda de paso normalizada W_p , la frecuencia de borde de banda de parada normalizada W_s , la ondulación de banda de paso R_p en dB y la atenuación de banda de parada mínima R_s en dB. Tanto W_p como W_s deben ser un número entre 0 y 1 con el muestreo se supone que la frecuencia es de 2 Hz. Los datos de salida son los de orden más bajo N que cumplen las especificaciones y la frecuencia de corte normalizada W_n . Si $R_p = 3$ dB, entonces $W_n = W_p$. `buttord` también se puede utilizar para estimar el orden de un paso alto, un paso de banda y una parada de banda Butterworth filtrar. Para un diseño de filtro de paso alto, $W_p > W_s$. Para los diseños de filtro de paso de banda y supresión de banda, W_p y W_s son vectores de dos elementos que especifican ambas frecuencias de borde, siendo la frecuencia de borde inferior el primer elemento del vector. En los últimos casos, W_n también es un vector de dos elementos.

Para estimar el orden de un filtro Chebyshev de tipo 1, el comando MATLAB es

$$[N, Wn] = cheb1ord (Wp, Ws, Rp, Rs)$$

y para diseñar un filtro Chebyshev de tipo 2, el comando MATLAB para estimar el orden es

$$[N, Wn] = cheb2ord (Wp, Ws, Rp, Rs).$$

Finalmente, en el caso de un diseño de filtro elíptico, el comando es

$$[N, Wn] = elipord (Wp, Ws, Rp, Rs).$$

Como antes, W_p y W_s son las frecuencias de borde de banda de paso y banda de supresión con valores entre 0 y 1. Asimismo, R_p y R_s son la ondulación de banda de paso y la atenuación mínima de banda de supresión en dB. N contiene el orden más bajo estimado y W_n es la frecuencia de corte. Cabe señalar que para los diseños de filtro de paso de banda y supresión de banda, el orden real de la función de transferencia obtenida utilizando el comando de diseño de filtro apropiado es $2N$

- **Questions:**

- 1) **Q7.1 Using MATLAB determine the lowest order of a digital IIR lowpass filter of all four types. The specifications are as follows: sampling rate of 40 kHz, passband edge frequency of 4 kHz, stopband edge frequency of 8 kHz, passband ripple of 0.5 dB, and a minimum stopband attenuation of 40 dB. Comment on your results.**

Del planteamiento del problema de Q7.1, tenemos $F_T = 40$ kHz, $F_p = 4$ kHz, $F_s = 8$ kHz, $R_p = 0.5$ dB, y $R_s = 40$ dB.

La frecuencia angular de borde de banda de paso normalizada W_p es –

$$\begin{aligned}\omega_p &= \frac{2\pi F_p}{F_T} = \frac{2\pi(4 \times 10^3)}{40 \times 10^3} = \frac{\pi}{5} = 0.2\pi. \\ W_p &= \frac{\omega_p}{\pi} = \frac{0.2\pi}{\pi} = 0.2\end{aligned}$$

La frecuencia angular del borde de la banda de supresión normalizada W_s es –

$$\begin{aligned}\omega_s &= \frac{2\pi F_s}{F_T} = \frac{2\pi(8 \times 10^3)}{40 \times 10^3} = \frac{2\pi}{5} = 0.4\pi. \\ W_s &= \frac{\omega_s}{\pi} = \frac{0.4\pi}{\pi} = 0.4\end{aligned}$$

La ondulación deseada de la banda de paso Rp es - 0.5 dB
 La ondulación deseada de la banda de detención Rs es – 40 dB

- a) Usando este valor y buttord obtenemos el pedido más bajo para que un filtro de paso bajo Butterworth sea – la llamada correcta es $[N, Wn] = \text{buttord}(0.2, 0.4, 0.5, 40)$. Esto da $N=8$.
 La frecuencia de borde de banda de paso normalizada correspondiente Wn es - 0.2469, o $\omega_n = 0.2469\pi$.
- b) Usando estos valores y cheb1ord obtenemos el orden más bajo para un filtro de paso bajo de Chebyshev Tipo 1: la llamada correcta es $[N, Wn] = \text{cheb1ord}(0.2, 0.4, 0.5, 40)$. Esto da $N=5$.
 La frecuencia de borde de banda de paso normalizada correspondiente Wn es - 0.2000, o $\omega_n = 0.2000\pi$.
- c) Usando estos valores y cheb2ord obtenemos el orden más bajo para que sea un filtro de paso bajo Chebyshev Tipo 2 - la llamada correcta es $[N, Wn] = \text{cheb2ord}(0.2, 0.4, 0.5, 40)$. Esto da $N = 5$.
 La frecuencia de borde de banda de paso normalizada correspondiente Wn es - 0.4000, o $\omega_n = 0.4000\pi$.
- d) Usando estos valores y ellipord obtenemos el orden más bajo para un filtro de paso bajo elíptico - la llamada correcta es $[N, Wn] = \text{ellipord}(0.2, 0.4, 0.5, 40)$. Esto da $N = 4$.
 La frecuencia de borde de banda de paso normalizada correspondiente $Wn = 0.2000$ o $\omega_n = 0.2000\pi$.

De los resultados anteriores observamos que el filtro Elliptic tiene el orden más bajo que cumple con las especificaciones.

- 2) **Q7.2 Using MATLAB determine the lowest order of a digital IIR highpass filter of all four types. The specifications are as follows: sampling rate of 3,500 Hz, passband edge frequency of 1,050 Hz, stopband edge frequency of 600 Hz, passband ripple of 1 dB, and a minimum stopband attenuation of 50 dB. Comment on your results.**

La frecuencia angular de borde de banda de paso normalizada Wp es –

$$\omega_p = \frac{2\pi F_p}{F_T} = \frac{2\pi(1.050 \times 10^3)}{3.5 \times 10^3} = \frac{3\pi}{5} = 0.6\pi.$$

$$W_p = \frac{\omega_p}{\pi} = \frac{0.6\pi}{\pi} = 0.6$$

La frecuencia angular del borde de la banda de supresión normalizada Ws es –

$$\omega_s = \frac{2\pi F_s}{F_T} = \frac{2\pi(0.6 \times 10^3)}{3.5 \times 10^3} = 0.3429\pi.$$

$$W_s = \frac{\omega_s}{\pi} = \frac{0.3429\pi}{\pi} = 0.3429$$

La ondulación deseada de la banda de paso Rp es - 1.0 dB
 La ondulación deseada de la banda de detención Rs es – 50 dB

- a) Usando este valor y buttord obtenemos el pedido más bajo para que un filtro de paso bajo Butterworth sea – la llamada correcta es $[N, Wn] = \text{buttord}(Wp, Ws, Rp, Rs)$. Esto da $N=8$.
 La frecuencia de borde de banda de paso normalizada correspondiente Wn es - 0.5646, o $\omega_n = 0.5646\pi$.
- b) Usando estos valores y cheb1ord obtenemos el orden más bajo para un filtro de paso bajo de Chebyshev Tipo 1: la llamada correcta es $[N, Wn] = \text{cheb1ord}(Wp, Ws, Rp, Rs)$. Esto da $N=5$.

La frecuencia de borde de banda de paso normalizada correspondiente W_n es - 0.6000, o $\omega_n = 0.6000\pi$.

- c) Usando estos valores y cheb2ord obtenemos el orden más bajo para que sea un filtro de paso bajo Chebyshev Tipo 2 - la llamada correcta es $[N, Wn] = \text{cheb2ord}(Wp, Ws, Rp, Rs)$. Esto da $N = 5$. La frecuencia de borde de banda de paso normalizada correspondiente W_n es - 0.3429, o $\omega_n = 0.3429\pi$.
- d) Usando estos valores y ellipord obtenemos el orden más bajo para un filtro de paso bajo elíptico - la llamada correcta es $[N, Wn] = \text{ellipord}(Wp, Ws, Rp, Rs)$. Esto da $N = 4$. La frecuencia de borde de banda de paso normalizada correspondiente $W_n = 0.6000$ o $\omega_n = 0.6000\pi$.

De los resultados anteriores observamos que el filtro Elliptic tiene el orden más bajo que cumple con las especificaciones.

- 3) **Q7.3 Using MATLAB determine the lowest order of a digital IIR bandpass filter of all four types. The specifications are as follows: sampling rate of 7 kHz, passband edge frequencies at 1.4 kHz and 2.1 kHz, stopband edge frequencies at 1.05 kHz and 2.45 kHz, passband ripple of 0.4 dB, and a minimum stopband attenuation of 50 dB. Comment on your results.**

La frecuencia angular de borde de banda de paso normalizada W_p es -

$$\begin{aligned}\omega_{p,1} &= \frac{2\pi F_{p,1}}{F_T} = \frac{2\pi(1.400 \times 10^3)}{7 \times 10^3} = \frac{2\pi}{5} = 0.4\pi. \\ Wp1 &= \frac{\omega_{p,1}}{\pi} = \frac{0.4\pi}{\pi} = 0.4 \\ \omega_{p,2} &= \frac{2\pi F_{p,2}}{F_T} = \frac{2\pi(2.100 \times 10^3)}{7 \times 10^3} = \frac{3\pi}{5} = 0.6\pi. \\ Wp2 &= \frac{\omega_{p,2}}{\pi} = \frac{0.6\pi}{\pi} = 0.6 \\ Wp &= [Wp1 \quad Wp2] = [0.4000 \quad 0.6000]\end{aligned}$$

La frecuencia angular del borde de la banda de supresión normalizada W_s es -

$$\begin{aligned}\omega_{s,1} &= \frac{2\pi F_{s,1}}{F_T} = \frac{2\pi(1.050 \times 10^3)}{7 \times 10^3} = \frac{3\pi}{10} = 0.3\pi. \\ Ws1 &= \frac{\omega_{s,1}}{\pi} = \frac{0.3\pi}{\pi} = 0.3 \\ \omega_{s,2} &= \frac{2\pi F_{s,2}}{F_T} = \frac{2\pi(2.450 \times 10^3)}{7 \times 10^3} = \frac{7\pi}{10} = 0.7\pi. \\ Ws2 &= \frac{\omega_{s,2}}{\pi} = \frac{0.7\pi}{\pi} = 0.7 \\ Ws &= [Ws1 \quad Ws2] = [0.3000 \quad 0.7000]\end{aligned}$$

La ondulación deseada de la banda de paso R_p es - 0.4 dB

La ondulación deseada de la banda de detención R_s es - 50 dB

- e) Usando este valor y buttord obtenemos el pedido más bajo para que un filtro de paso bajo Butterworth sea - la llamada correcta es $[N, Wn] = \text{buttord}(Wp, Ws, Rp, Rs) = \text{buttord}([0.4000 \quad 0.6000], [0.3000 \quad 0.7000], 0.4, 50)$. Esto da Order = $2N = 18$. La frecuencia de borde de banda de paso normalizada correspondiente $W_n = [0.3835 \quad 0.6165]$, o $\omega_{n,1} =$

$$0.3835\pi \text{ y } \omega_{n,2} = 0.6165\pi.$$

- f) Usando estos valores y cheb1ord obtenemos el orden más bajo para un filtro de paso bajo de Chebyshev Tipo 1: la llamada correcta es $[N, Wn] = \text{cheb1ord}(Wp, Ws, Rp, Rs) = \text{cheb1ord}([0.40000 0.6000], [0.3000 0.7000], 0.4, 50)$. Esto da Order = $2N = 12$. La frecuencia de borde de banda de paso normalizada correspondiente $Wn = [0.4000 0.6000]$, o $\omega_{n,1} = 0.4000\pi$ y $\omega_{n,2} = 0.6000\pi$.
- g) Usando estos valores y cheb2ord obtenemos el orden más bajo para que sea un filtro de paso bajo Chebyshev Tipo 2 - la llamada correcta es $[N, Wn] = \text{cheb2ord}(Wp, Ws, Rp, Rs) = \text{cheb2ord}([0.4000 0.6000], [0.3000 0.7000], 0.4, 50)$. Esto da Order = $2N = 12$. La frecuencia de borde de banda de paso normalizada correspondiente $Wn = [0.3000 0.7000]$, o $\omega_{n,1} = 0.3000\pi$ y $\omega_{n,2} = 0.7000\pi$.
- h) Usando estos valores y ellipord obtenemos el orden más bajo para un filtro de paso bajo elíptico - la llamada correcta es $[N, Wn] = \text{ellipord}(Wp, Ws, Rp, Rs) = \text{ellipord}([0.4000 0.6000], [0.3000 0.7000], 0.4, 50)$. Esto da Order = $2N = 8$. La frecuencia de borde de banda de paso normalizada correspondiente $Wn = [0.4000 0.6000]$, o $\omega_{n,1} = 0.4000\pi$ y $\omega_{n,2} = 0.6000\pi$.

De los resultados anteriores observamos que el filtro Elliptic tiene el orden más bajo que cumple con las especificaciones.

- 4) **Q7.4 Using MATLAB determine the lowest order of a digital IIR bandstop filter of all four types. The specifications are as follows: sampling rate of 12 kHz, passband edge frequencies at 2.1 kHz and 4.5 kHz, stopband edge frequencies at 2.7 kHz and 3.9 kHz, passband ripple of 0.6 dB, and a minimum stopband attenuation of 45 dB. Comment on your results.**

La frecuencia angular de borde de banda de paso normalizada Wp es –

$$\begin{aligned}\omega_{p,1} &= \frac{2\pi F_{p,1}}{F_T} = \frac{2\pi(2.100 \times 10^3)}{12 \times 10^3} = 0.35\pi. \\ Wp1 &= \frac{\omega_{p,1}}{\pi} = \frac{0.35\pi}{\pi} = 0.35 \\ \omega_{p,2} &= \frac{2\pi F_{p,2}}{F_T} = \frac{2\pi(4.500 \times 10^3)}{12 \times 10^3} = 0.75\pi. \\ Wp2 &= \frac{\omega_{p,2}}{\pi} = \frac{0.75\pi}{\pi} = 0.75 \\ Wp &= [Wp1 \ Wp2] = [0.3500 \ 0.7500]\end{aligned}$$

La frecuencia angular del borde de la banda de supresión normalizada Ws es –

$$\begin{aligned}\omega_{s,1} &= \frac{2\pi F_{s,1}}{F_T} = \frac{2\pi(2.700 \times 10^3)}{12 \times 10^3} = 0.45\pi. \\ Ws1 &= \frac{\omega_{s,1}}{\pi} = \frac{0.45\pi}{\pi} = 0.45 \\ \omega_{s,2} &= \frac{2\pi F_{s,2}}{F_T} = \frac{2\pi(3.900 \times 10^3)}{12 \times 10^3} = 0.65\pi. \\ Ws2 &= \frac{\omega_{s,2}}{\pi} = \frac{0.65\pi}{\pi} = 0.65 \\ Ws &= [Ws1 \ Ws2] = [0.4500 \ 0.6500]\end{aligned}$$

La ondulación deseada de la banda de paso Rp es – 0.6 dB

La ondulación deseada de la banda de detención Rs es – 45 dB

- a) Usando este valor y buttord obtenemos el pedido más bajo para que un filtro de paso bajo Butterworth sea – la llamada correcta es $[N, Wn] = \text{buttord}(Wp, Ws, Rp, Rs) = \text{buttord}([0.3500 0.7500], [0.4500 0.6500], 0.6, 45)$. Esto da Order = $2N = 18$.
La frecuencia de borde de banda de paso normalizada correspondiente $Wn = [0.3783 0.7123]$, o $\omega_{n,1} = 0.3783\pi$ y $\omega_{n,2} = 0.7123\pi$.
- b) Usando estos valores y cheb1ord obtenemos el orden más bajo para un filtro de paso bajo de Chebyshev Tipo 1: la llamada correcta es $[N, Wn] = \text{cheb1ord}(Wp, Ws, Rp, Rs) = \text{cheb1ord}([0.3500 0.7500], [0.4500 0.6500], 0.6, 45)$, Esto da Order = $2N = 10$.
La frecuencia de borde de banda de paso normalizada correspondiente $Wn = [0.3500 0.7500]$, o $\omega_{n,1} = 0.3500\pi$ y $\omega_{n,2} = 0.7500\pi$.
- c) Usando estos valores y cheb2ord obtenemos el orden más bajo para que sea un filtro de paso bajo Chebyshev Tipo 2 - la llamada correcta es $[N, Wn] = \text{cheb2ord}(Wp, Ws, Rp, Rs) = \text{cheb2ord}([0.3500 0.7500], [0.4500 0.6500], 0.6, 45)$, Esto da Order = $2N = 10$.
La frecuencia de borde de banda de paso normalizada correspondiente $Wn = [0.4500 0.6500]$, o $\omega_{n,1} = 0.4500\pi$ y $\omega_{n,2} = 0.6500\pi$.
- d) Usando estos valores y ellipord obtenemos el orden más bajo para un filtro de paso bajo elíptico - la llamada correcta es $[N, Wn] = \text{ellipord}(Wp, Ws, Rp, Rs) = \text{ellipord}([0.3500 0.7500], [0.4500 0.6500], 0.6, 45)$, Esto da Order = $2N = 8$.
La frecuencia de borde de banda de paso normalizada correspondiente $Wn = [0.3500 0.7500]$, o $\omega_{n,1} = 0.3500\pi$ y $\omega_{n,2} = 0.7500\pi$.

De los resultados anteriores observamos que el filtro Elliptic tiene el orden más bajo que cumple con las especificaciones.

B. Project 7.2 IIR Filter Design

Una vez que se ha seleccionado el tipo de filtro y se ha estimado su orden, el siguiente paso es determinar la función de transferencia del filtro. Con este fin, MATLAB proporciona funciones para los cuatro tipos de filtros. Para diseñar filtros digitales de paso bajo o de banda de Butterworth, el comando es

[num, den] = butter (N, Wn)

donde los parámetros de entrada N y Wn se determinan mediante el uso de la función buttord, y la salida son los vectores num y den que contienen, respectivamente, los coeficientes de los polinomios numerador y denominador de la función de transferencia en potencias ascendentes de z^{-1} . Si Wn es un escalar, la mantequilla devuelve una función de transferencia de paso bajo de orden N, y si Wn es un vector de dos elementos, devuelve una función de transferencia de paso de banda de orden 2N. Para diseñar un filtro de paso alto digital Butterworth de orden N, el comando es

[num, den] = butter (N, Wn, "alto")

mientras que el comando

[num, den] = butter (N, Wn, "detener")

devuelve la función de transferencia de un filtro de parada de banda de Butterworth de orden 2N siempre que Wn sea un vector de dos elementos. Para diseñar un filtro digital Chebyshev Tipo 1, los comandos son

[num, den] = cheby1 (N, Rp, Wn)
[num, den] = cheby1 (N, Rp, Wn, "tipo de filtro")

Para diseñar un filtro digital Chebyshev Tipo 2, los comandos son

[num, den] = cheby2 (N, Rs, Wn)
[num, den] = cheby2 (N, Rs, Wn, "tipo de filtro")

Finalmente, para diseñar un filtro digital elíptico, los comandos son

[num, den] = ellip (N, Rp, Rs, Wn)
[num, den] = ellip (N, Rp, Rs, Wn, "tipo de filtro")

Se devuelve una función de transferencia de paso bajo de orden N en cada caso si Wn es un escalar, y se devuelve una función de transferencia de paso de banda de orden 2N si Wn es un vector de dos elementos. En cada uno de los comandos anteriores, el tipo de filtro es alto para diseñar un filtro de paso alto con Wn como escalar y filtertype es stop para diseñar un filtro de detención de banda con Wn como un vector de dos elementos. El programa P7_1 ilustra el diseño de un filtro de parada de banda Butterworth:

```
1. % Program P7_1
2. % Design of a Butterworth Bandstop Digital Filter
3. Ws = [0.4 0.6]; Wp = [0.3 0.7]; Rp = 0.4; Rs = 50;
4. % Estimate the Filter Order
5. [N1, Wn1] = buttord(Wp, Ws, Rp, Rs);
6. % Design the Filter
7. [num,den] = butter(N1,Wn1,'stop');
8. % Display the transfer function
9. disp('Numerator coefficients are ');disp(num);
10. disp('Denominator coefficients are ');disp(den);
11. % Compute the gain response
12. [g,w] = gain(num,den);
13. % Plot the gain response
14. plot(w/pi,g);grid
15. axis([0 1 -60 5]);
```

```

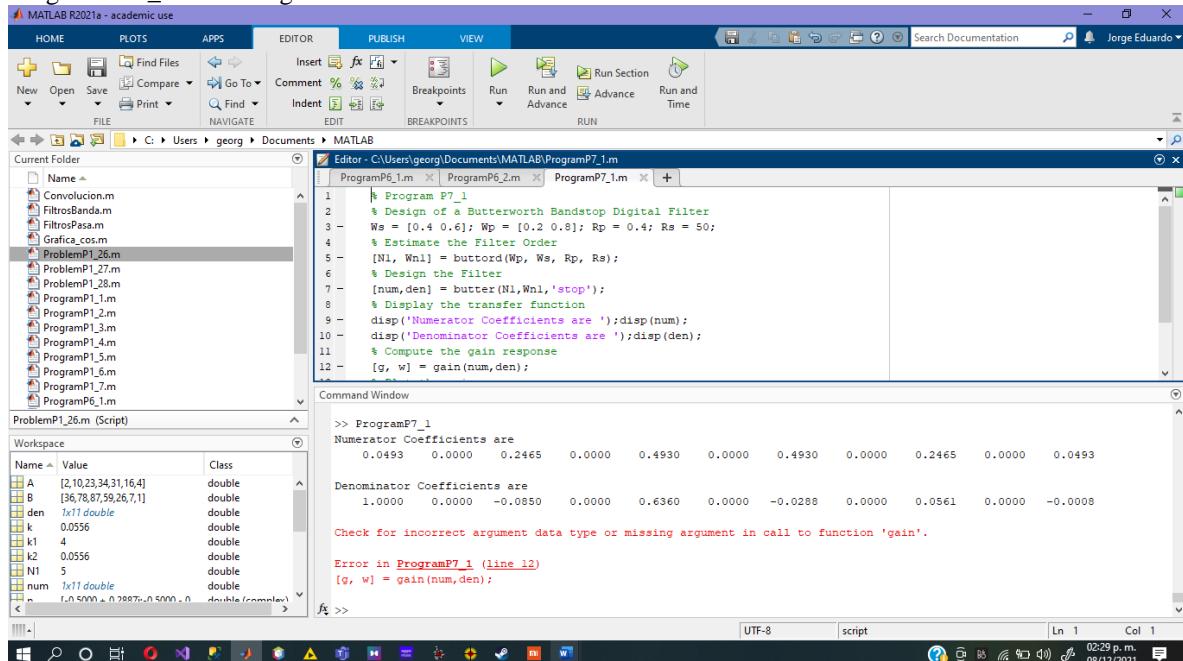
16. xlabel ('\omega /\pi'); ylabel ('Gain, dB');
17. title ('Gain Response of a Butterworth Bandstop Filter');

```

- **Questions:**

- 5) **Q7.5 Design the Butterworth bandstop filter by running Program P7_1. Write down the exact expression for the transfer function generated. What are the filter specifications? Does your design meet the specifications? Using MATLAB, compute and plot the filter's unwrapped phase response and the group delay response.**

Los coeficientes de la función de transferencia de parada de banda de Butterworth generados al ejecutar el Programa P7_1 son los siguientes:

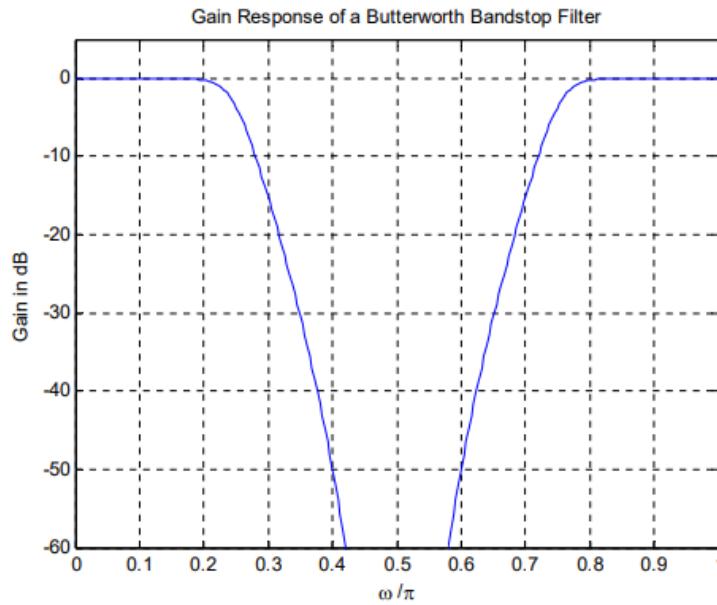


La expresión exacta de la función de transferencia es:

$$H(z) = \frac{0.0493 + 0.2465z^{-2} + 0.4930z^{-4} + 0.4930z^{-6} + 0.2465z^{-8} + 0.0493z^{-10}}{1 - 0.0850z^{-2} + 0.6360z^{-4} - 0.0288z^{-6} + 0.0561z^{-8} - 0.0008z^{-10}}$$

Las especificaciones del filtro son: $\omega_{p1} = 0.2\pi$, $\omega_{s1} = 0.4\pi$, $\omega_{s2} = 0.6\pi$, $\omega_{p2} = 0.8\pi$, $R_p = 0.4dB$ y $R_s = 50dB$.

La respuesta de ganancia del filtro como se diseñó se da a continuación:



De la gráfica concluimos que el diseño CUMPLE con las especificaciones.

El gráfico de la respuesta de fase sin envolver y la respuesta de retardo de grupo de este filtro se muestra a continuación:

Aquí está el programa para encontrar y trazar la respuesta de fase sin envolver y el retardo de grupo:

```

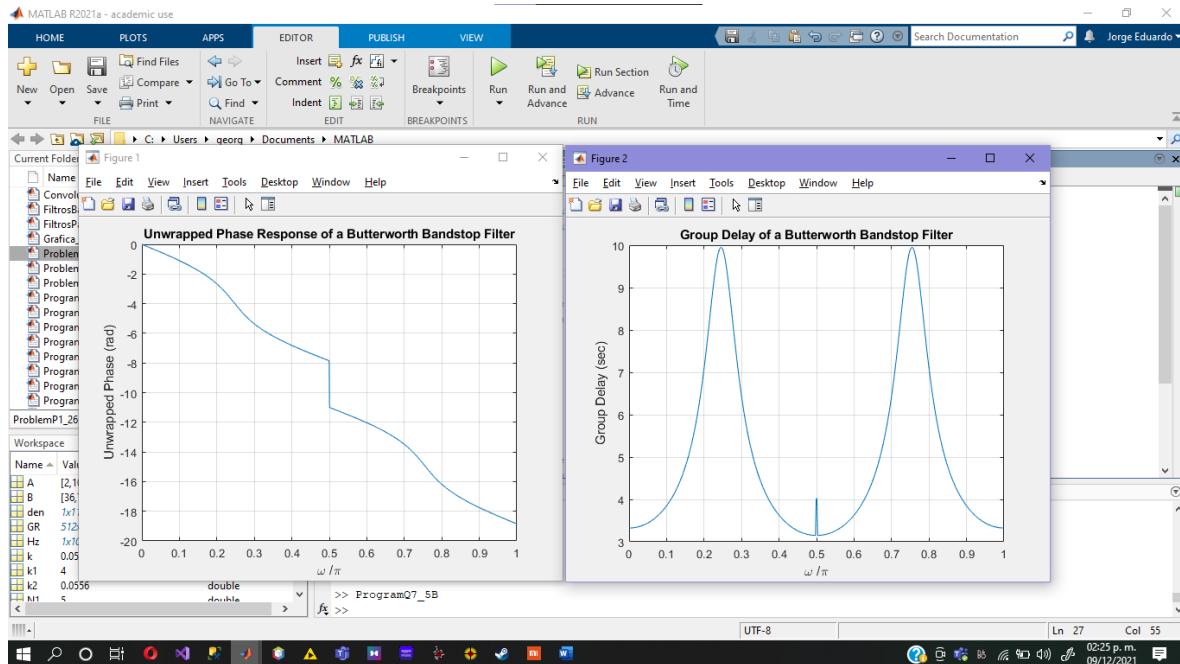
1. % Program Q7_5B
2. % Design of a Butterworth Bandstop Digital Filter for Q7.5.
3. % Plot the unwrapped phase and the group delay.
4. Ws = [0.4 0.6]; Wp = [0.2 0.8]; Rp = 0.4; Rs = 50;
5. % Estimate the Filter Order
6. [N1, Wn1] = buttord(Wp, Ws, Rp, Rs);
7. % Design the Filter
8. [num,den] = butter(N1,Wn1,'stop');
9. % Find the frequency response; find and plot unwrapped phase
10. wp = 0:pi/1023:pi;
11. wg = 0:pi/511:pi;
12. Hz = freqz(num,den,wp);
13. Phase = unwrap(angle(Hz));
14. figure(1);
15. plot(wp/pi,Phase);
16. grid;
17. % axis([0 1 a b]);
18. xlabel('\omega / \pi'); ylabel('Unwrapped Phase (rad)');
19. title('Unwrapped Phase Response of a Butterworth
Bandstop Filter');
20. % Find and plot the group delay
21. GR = grpdelay(num,den,wg);
22. figure(2);
23. plot(wg/pi,GR);

```

```

24.     grid;
25. %axis([0 1 a b]);
26. xlabel('omega /pi'); ylabel('Group Delay (sec)');
27. title('Group Delay of a Butterworth Bandstop Filter');

```



- 6) Q7.6 Modify Program P7_1 to design a Type 1 Chebyshev lowpass filter meeting the specifications given in Question Q7.1. Write down the exact expression for the transfer function generated. Does your design meet the specifications? Using MATLAB, compute and plot the filter's unwrapped phase response and the group delay response.

Se realizaron las siguientes modificaciones al código inicial:

```

1. % Program Q7_6
2. % Design of a Chebyshev Type 1 Lowpass Digital Filter
3. % meeting the design specification given in Q7.1.
4. % - Print out the numerator and denominator coefficients
5. % for the transfer function.
6. % - Compute and plot the gain function.
7. % - Compute and plot the unwrapped phase response.
8. % - Compute and plot the group delay.
9. %%%%%%
10. % Design spec as given in Q7.1.
11. FT = 40*10^3; % sampling freq
12. Fp = 4*10^3; % analog passband edge freq
13. Fs = 8*10^3; % analog stopband edge freq
14. Rp = 0.5; % max passband ripple, dB
15. Rs = 40; % min stopband attenuation, dB
16. % Convert spec to normalized digital frequencies

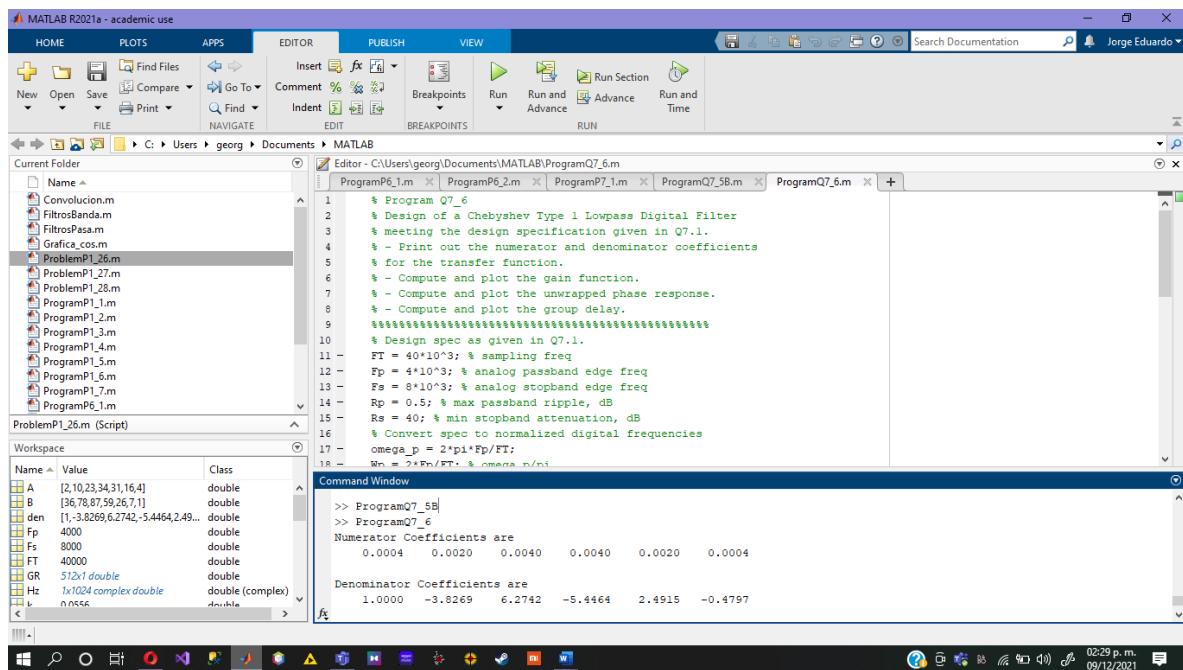
```

```

17.     omega_p = 2*pi*Fp/FT;
18.     Wp = 2*Fp/FT; % omega_p/pi
19.     omega_s = 2*pi*Fs/FT;
20.     Ws = 2*Fs/FT; % omega_s/pi
21.     % Estimate the Filter Order
22.     [N, Wn] = cheblord(Wp, Ws, Rp, Rs);
23.     % Design the Filter
24.     [num,den] = cheby1(N,Rp,Wn);
25.     % Display the transfer function
26.     disp('Numerator Coefficients are ');disp(num);
27.     disp('Denominator Coefficients are ');disp(den);
28.     % Compute the gain response
29.     [g, w] = gain(num,den);
30.     % Plot the gain response
31.     figure(1);
32.     plot(w/pi,g);grid;
33.     axis([0 1 -60 5]);
34.     xlabel('\omega /\pi'); ylabel('Gain in dB');
35.     title('Gain Response of a Type 1 Chebyshev Lowpass
    Filter');
36.     % Find and plot the phase
37.     figure(2);
38.     w2 = 0:pi/511:pi;
39.     Hz = freqz(num,den,w2);
40.     Phase = unwrap(angle(Hz));
41.     plot(w2/pi,Phase);grid;
42.     xlabel('\omega /\pi'); ylabel('Unwrapped Phase (rad)');
43.     title('Unwrapped Phase Response of a Type 1 Chebyshev
    Lowpass Filter');
44.     % Find and plot the group delay
45.     figure(3);
46.     GR = grpdelay(num,den,w2);
47.     plot(w2/pi,GR);grid;
48.     xlabel('\omega /\pi'); ylabel('Group Delay (sec)');
49.     title('Group Delay of a Type 1 Chebyshev Lowpass
    Filter');

```

Los coeficientes de la función de transferencia de paso bajo de Chebyshev Tipo 1 para los parámetros dados en la Pregunta 7.1 y generados al ejecutar el Programa P7_1 modificado son los siguientes:

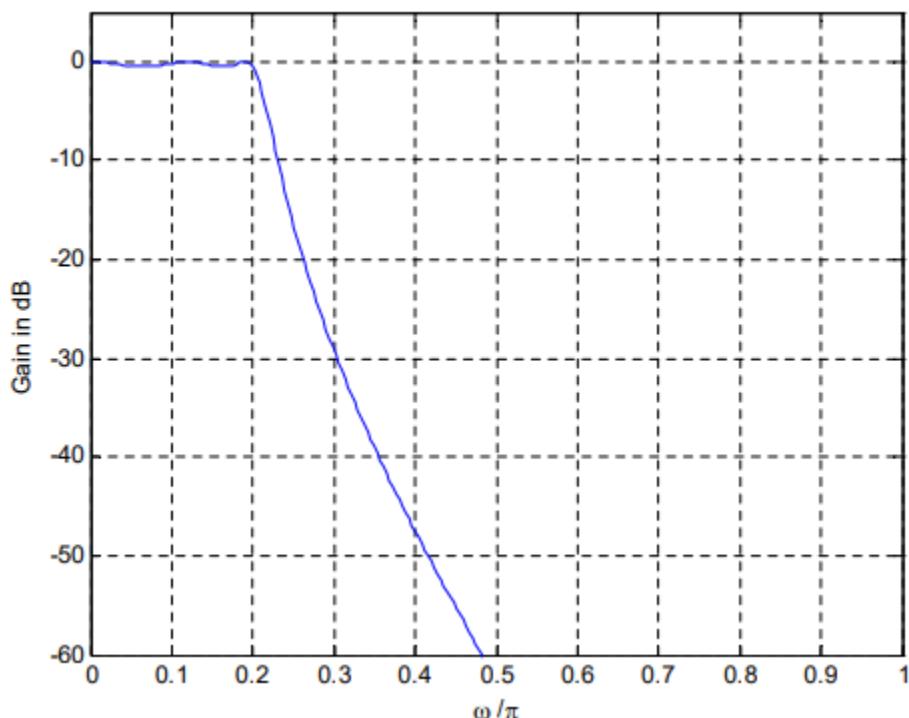


La expresión exacta de la función de transferencia es:

$$H(z) = \frac{0.0004 + 0.0020z^{-1} + 0.0040z^{-2} + 0.0040z^{-3} + 0.0020z^{-4} + 0.0004z^{-5}}{1 - 3.8269z^{-1} + 6.2742z^{-2} - 5.4464z^{-3} + 2.4915z^{-4} - 0.4797z^{-5}}$$

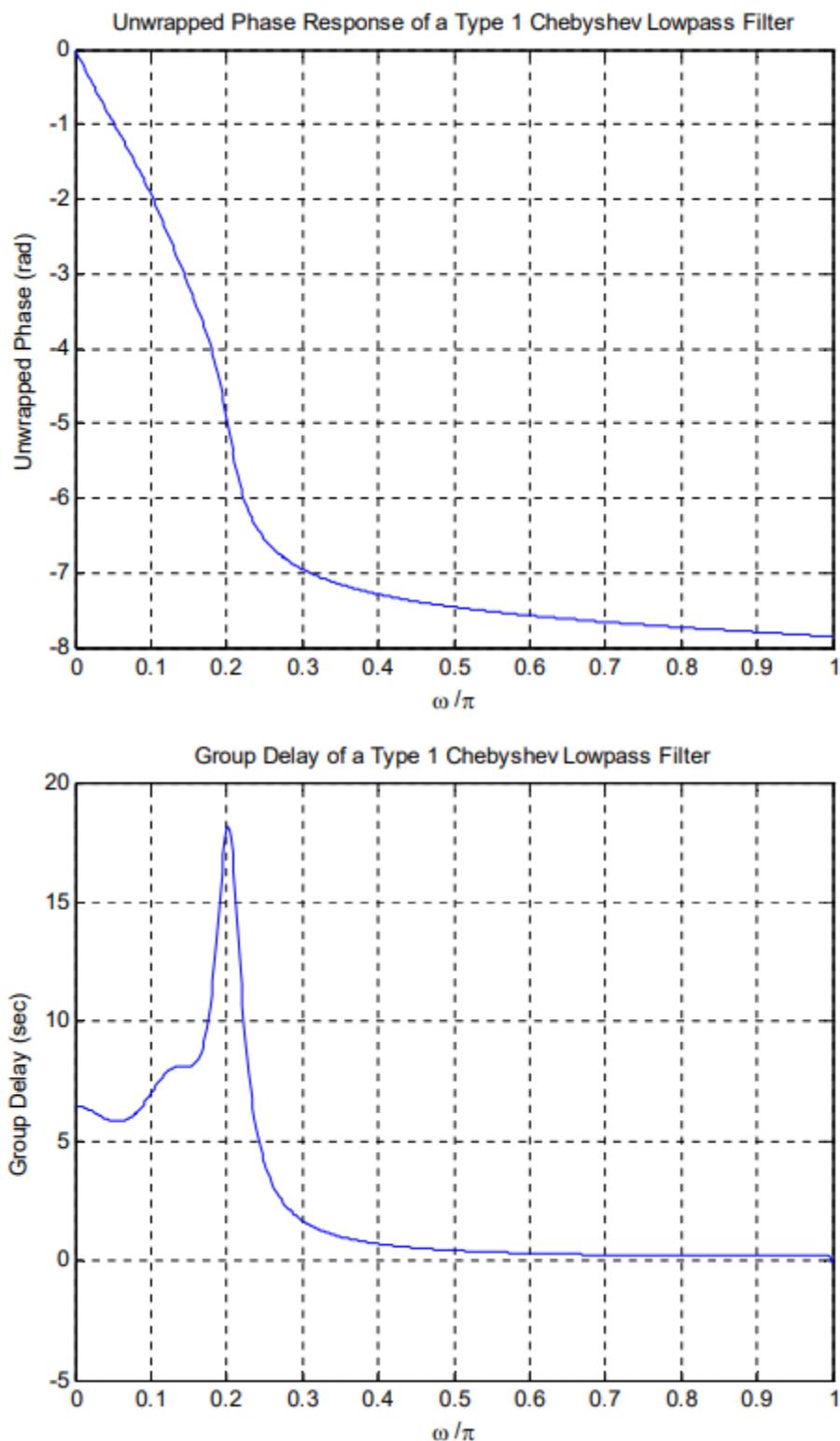
La respuesta de ganancia del filtro como se diseñó se da a continuación:

Gain Response of a Type 1 Chebyshev Lowpass Filter



De la gráfica concluimos que el diseño CUMPLE con las especificaciones.

El gráfico de la respuesta de fase sin envolver y la respuesta de retardo de grupo de este filtro se muestra a continuación:



- 7) **Q7.7 Modify Program P7_1 to design a Type 2 Chebyshev highpass filter meeting the specifications given in Question Q7.2. Write down the exact expression for the transfer function generated. Does your design meet the specifications? Using MATLAB, compute and plot the filter's unwrapped phase response and the group delay response.**

Se realizó la siguiente modificación al código original:

```

1. % Program Q7_7
2. % Design of a Chebyshev Type 2 Highpass Digital Filter
3. % meeting the design specification given in Q7.2.
4. % - Print out the numerator and denominator coefficients
5. % for the transfer function.
6. % - Compute and plot the gain function.
7. % - Compute and plot the unwrapped phase response.
8. % - Compute and plot the group delay.
9. %%%%%%%%%%%%%%
10. % Design spec as given in Q7.2.
11. FT = 3.50*10^3; % sampling freq
12. Fp = 1.05*10^3; % analog passband edge freq
13. Fs = 0.60*10^3; % analog stopband edge freq
14. Rp = 1.0; % max passband ripple, dB
15. Rs = 50; % min stopband attenuation, dB
16. % Convert spec to normalized digital frequencies
17. omega_p = 2*pi*Fp/FT;
18. Wp = 2*Fp/FT; % omega_p/pi
19. omega_s = 2*pi*Fs/FT;
20. Ws = 2*Fs/FT; % omega_s/pi
21. % Estimate the Filter Order
22. [N, Wn] = cheb2ord(Wp, Ws, Rp, Rs);
23. % Design the Filter
24. [num,den] = cheby2(N,Rs,Wn,'high');
25. % Display the transfer function
26. disp('Numerator Coefficients are ');disp(num);
27. disp('Denominator Coefficients are ');disp(den);
28. % Compute the gain response
29. [g, w] = gain(num,den);
30. % Plot the gain response
31. figure(1);
32. axis([0 1 -60 5]);
33. % Add lines to the plot to help determine if the spec
   was met.
34. hold on;
35. tmpY = -60:65/511:5;
36. tmpX = ones(1,length(tmpY))*Wp;
37. plot(tmpX,tmpY,'r-'); % vertical line at passband edge
   freq
38. tmpX = ones(1,length(tmpY))*Ws;

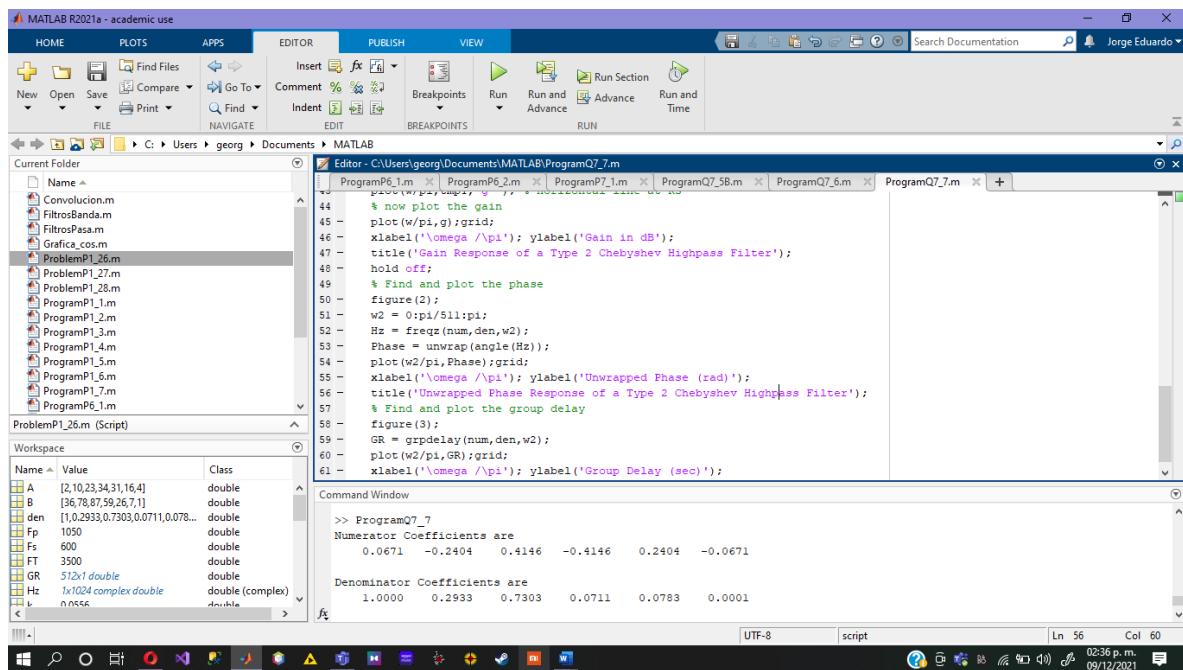
```

```

39. plot(tmpX,tmpY,'g-'); % vertical line at stopband edge
   freq
40. tmpY = ones(1,length(w))*(-Rp);
41. plot(w/pi,tmpY,'r-'); % horizontal line at Rp
42. tmpY = ones(1,length(w))*(-Rs);
43. plot(w/pi,tmpY,'g-'); % horizontal line at Rs
44. % now plot the gain
45. plot(w/pi,g);grid;
46. xlabel('omega /pi'); ylabel('Gain in dB');
47. title('Gain Response of a Type 2 Chebyshev Highpass
   Filter');
48. hold off;
49. % Find and plot the phase
50. figure(2);
51. w2 = 0:pi/511:pi;
52. Hz = freqz(num,den,w2);
53. Phase = unwrap(angle(Hz));
54. plot(w2/pi,Phase);grid;
55. xlabel('omega /pi'); ylabel('Unwrapped Phase (rad)');
56. title('Unwrapped Phase Response of a Type 2 Chebyshev
   Highpass Filter');
57. % Find and plot the group delay
58. figure(3);
59. GR = grpdelay(num,den,w2);
60. plot(w2/pi,GR);grid;
61. xlabel('omega /pi'); ylabel('Group Delay (sec)');
62. title('Group Delay of a Type 2 Chebyshev Highpass
   Filter');

```

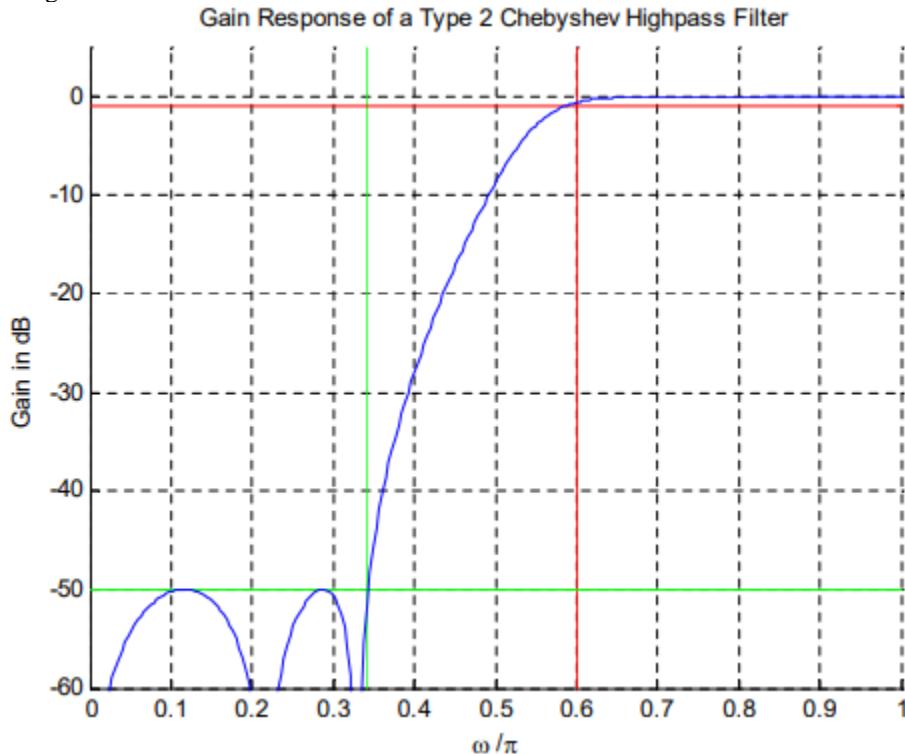
Los coeficientes de la función de transferencia de paso alto de Chebyshev Tipo 1 para los parámetros dados en la Pregunta 7.2 y generados al ejecutar el Programa P7_1 modificado son los siguientes:



La expresión exacta de la función de transferencia es:

$$H(z) = \frac{0.0671 - 0.2404z^{-1} + 0.4146z^{-2} - 0.4146z^{-3} + 0.2404z^{-4} - 0.0671z^{-5}}{1 + 0.2933z^{-1} + 0.7303z^{-2} + 0.0711z^{-3} + 0.0783z^{-4} + 0.0001z^{-5}}$$

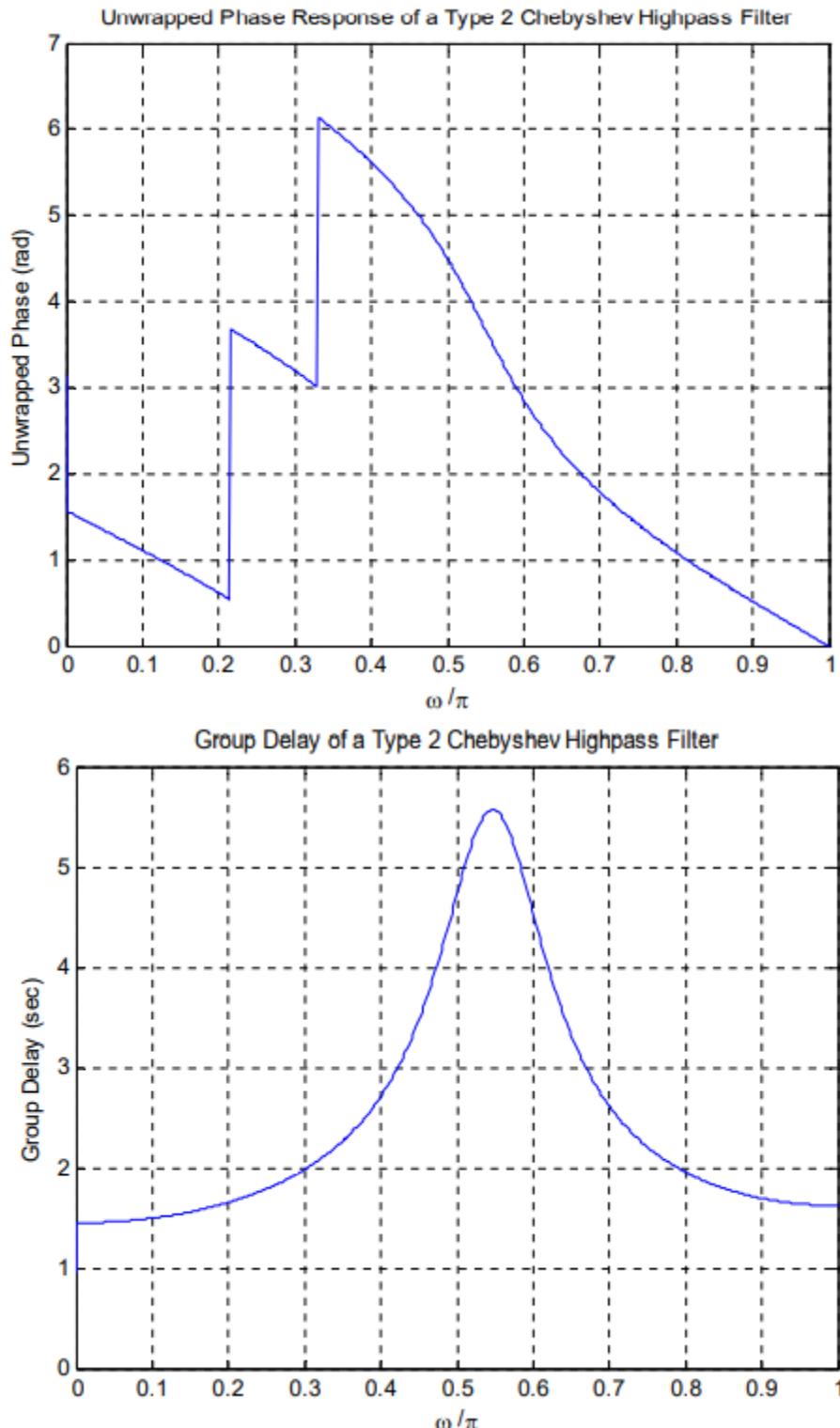
La respuesta de ganancia del filtro como se diseñó se da a continuación:



De la gráfica concluimos que el diseño CUMPLE con las especificaciones.

El gráfico de la respuesta de fase sin envolver y la respuesta de retardo de grupo de este filtro se muestra a

continuación:



- 8) Q7.8 Modify Program P7.1 to design an elliptic bandpass filter meeting the specifications given in Question Q7.3. Write down the exact expression for the transfer function generated. Does your design meet the specifications? Using MATLAB, compute and plot the filter's unwrapped

phase response and the group delay response.

Se realizó la siguiente modificación al código original:

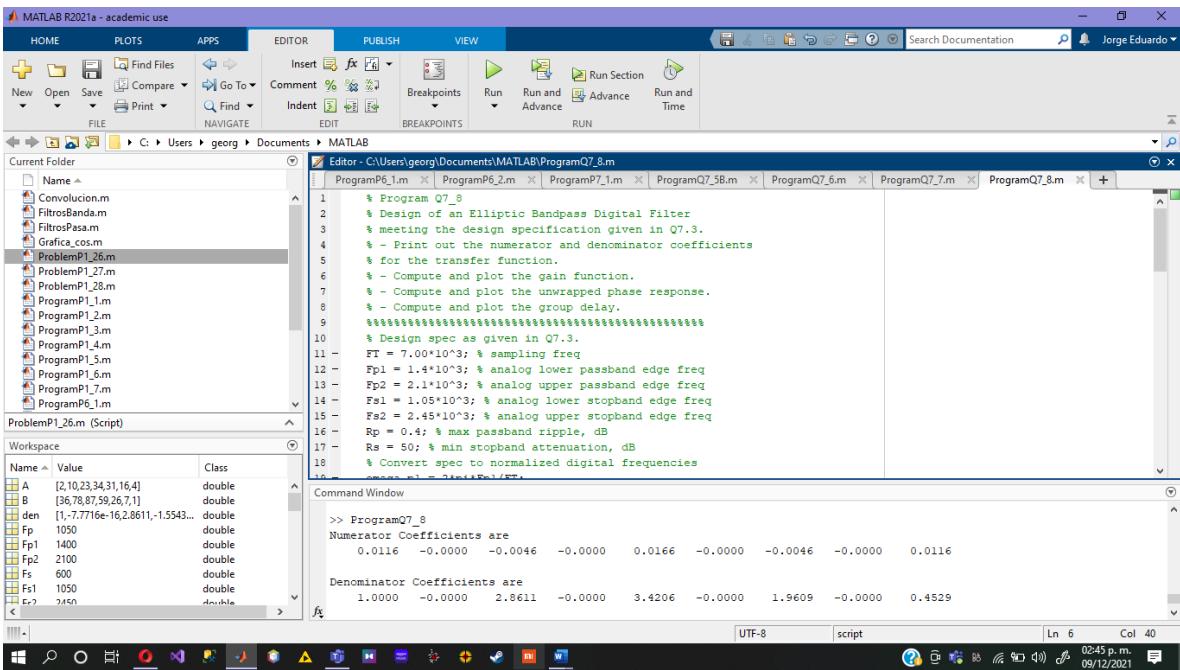
```
1. % Program Q7_8
2. % Design of an Elliptic Bandpass Digital Filter
3. % meeting the design specification given in Q7.3.
4. % - Print out the numerator and denominator coefficients
5. % for the transfer function.
6. % - Compute and plot the gain function.
7. % - Compute and plot the unwrapped phase response.
8. % - Compute and plot the group delay.
9. %%%%%%%%%%%%%%%%
10. % Design spec as given in Q7.3.
11. FT = 7.00*10^3; % sampling freq
12. Fp1 = 1.4*10^3; % analog lower passband edge freq
13. Fp2 = 2.1*10^3; % analog upper passband edge freq
14. Fs1 = 1.05*10^3; % analog lower stopband edge freq
15. Fs2 = 2.45*10^3; % analog upper stopband edge freq
16. Rp = 0.4; % max passband ripple, dB
17. Rs = 50; % min stopband attenuation, dB
18. % Convert spec to normalized digital frequencies
19. omega_p1 = 2*pi*Fp1/FT;
20. Wp1 = 2*Fp1/FT; % omega_p1/pi
21. omega_p2 = 2*pi*Fp2/FT;
22. Wp2 = 2*Fp2/FT; % omega_p2/pi
23. Wp = [Wp1 Wp2];
24. omega_s1 = 2*pi*Fs1/FT;
25. Ws1 = 2*Fs1/FT; % omega_s1/pi
26. omega_s2 = 2*pi*Fs2/FT;
27. Ws2 = 2*Fs2/FT; % omega_s2/pi
28. Ws = [Ws1 Ws2];
29. % Estimate the Filter Order
30. [N, Wn] = ellipord(Wp, Ws, Rp, Rs);
31. % Design the Filter
32. [num,den] = ellip(N,Rp,Rs,Wn);
33. % Display the transfer function
34. disp('Numerator Coefficients are ');disp(num);
35. disp('Denominator Coefficients are ');disp(den);
36. % Compute the gain response
37. [g, w] = gain(num,den);
38. % Plot the gain response
39. figure(1);
40. axis([0 1 -60 5]);
41. % Add lines to the plot to help determine if the spec
   was met.
```

```

42.      hold on;
43.      tmpY = -60:65/511:5;
44.      tmpX = ones(1,length(tmpY))*Wp1;
45.      plot(tmpX,tmpY,'r-'); % vertical line at passband edge
    freq
46.      tmpX = ones(1,length(tmpY))*Wp2;
47.      plot(tmpX,tmpY,'r-'); % vertical line at passband edge
    freq
48.      tmpX = ones(1,length(tmpY))*Ws1;
49.      plot(tmpX,tmpY,'g-'); % vertical line at stopband edge
    freq
50.      tmpX = ones(1,length(tmpY))*Ws2;
51.      plot(tmpX,tmpY,'g-'); % vertical line at stopband edge
    freq
52.      tmpY = ones(1,length(w))*(-Rp);
53.      plot(w/pi,tmpY,'r-'); % horizontal line at Rp
54.      tmpY = ones(1,length(w))*(-Rs);
55.      plot(w/pi,tmpY,'g-'); % horizontal line at Rs
56.      % now plot the gain
57.      plot(w/pi,g);grid;
58.      xlabel('\omega /\pi'); ylabel('Gain in dB');
59.      title('Gain Response of an Elliptic Bandpass Filter');
60.      hold off;
61.      % Find and plot the phase
62.      figure(2);
63.      w2 = 0:pi/511:pi;
64.      Hz = freqz(num,den,w2);
65.      Phase = unwrap(angle(Hz));
66.      plot(w2/pi,Phase);grid;
67.      xlabel('\omega /\pi'); ylabel('Unwrapped Phase (rad)');
68.      title('Unwrapped Phase Response of an Elliptic Bandpass
    Filter');
69.      % Find and plot the group delay
70.      figure(3);
71.      GR = grpdelay(num,den,w2);
72.      plot(w2/pi,GR);grid;
73.      xlabel('\omega /\pi'); ylabel('Group Delay (sec)');
74.      title('Group Delay of an Elliptic Bandpass Filter');

```

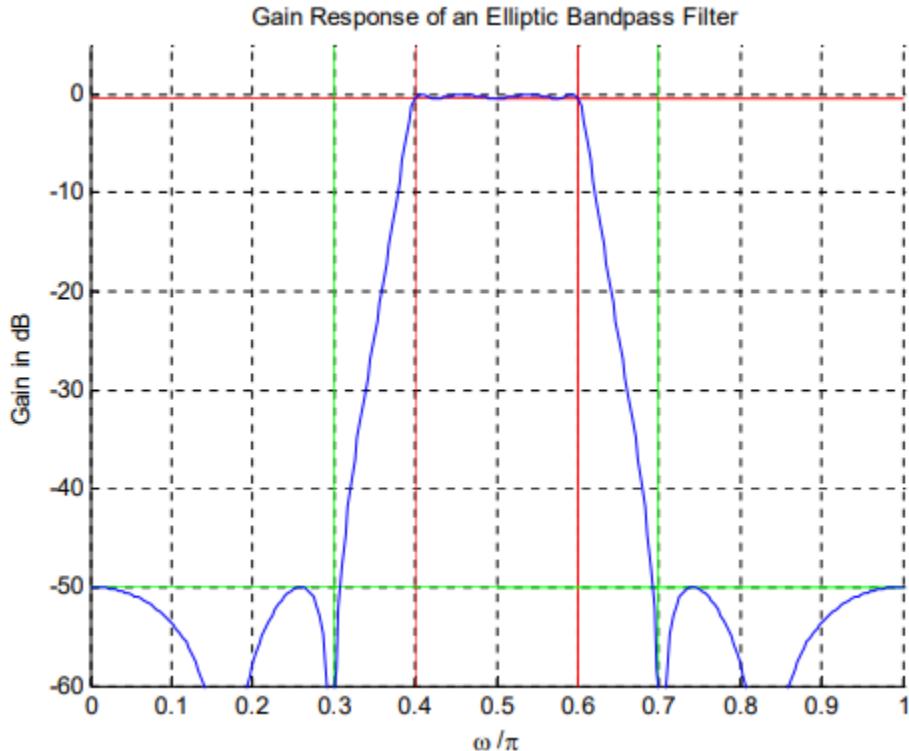
Los coeficientes de la función de transferencia de paso alto de Chebyshev Tipo 1 para los parámetros dados en la Pregunta 7.3 y generados al ejecutar el Programa P7_1 modificado son los siguientes:



La expresión exacta de la función de transferencia es:

$$H(z) = \frac{0.0116 - 0.0046z^{-2} + 0.0166z^{-4} - 0.0046z^{-6} + 0.0116z^{-8}}{1 + 2.8611z^{-2} + 3.4205z^{-4} + 1.9609z^{-6} + 0.4529z^{-8}}$$

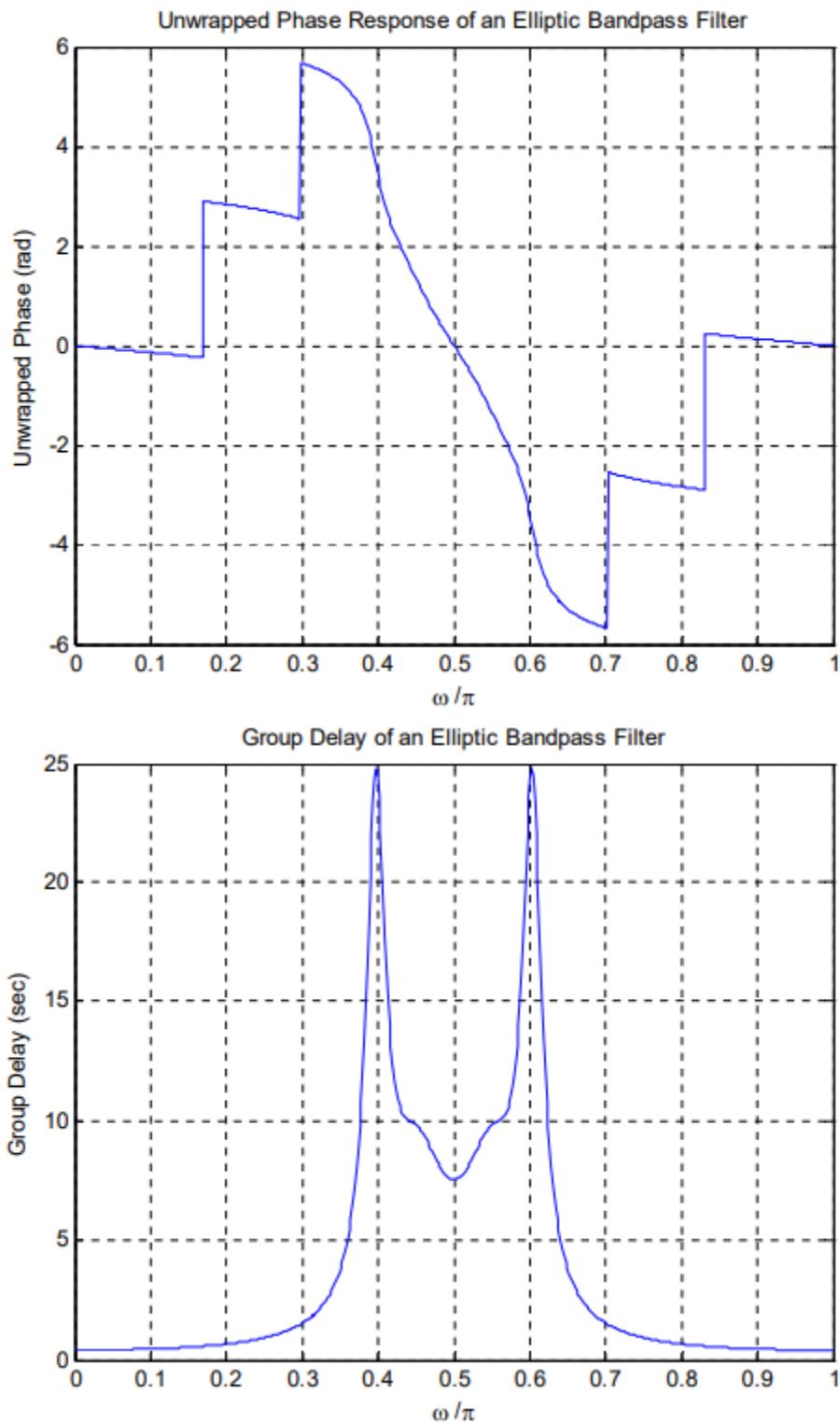
La respuesta de ganancia del filtro como se diseñó se da a continuación:



De la gráfica concluimos que el diseño CUMPLE con las especificaciones.

El gráfico de la respuesta de fase sin envolver y la respuesta de retardo de grupo de este filtro se muestra a

continuación:



C. Project 7.3 Gibb's Phenomenon

The occurrence of Gibb's phenomenon can be illustrated by considering the design of an FIR filter obtained by truncating the impulse response of the ideal filters given by Eqs. (7.16), (7.18)–(7.20), (7.22), (7.24) and (7.26), and then computing their frequency responses. The truncated impulse response coefficients of a lowpass filter can be generated in MATLAB using the function sinc, which can also be used with simple modifications to generate the truncated impulse response coefficients of a highpass, bandpass, or bandstop filter

- **Questions:**

- 9) **Q7.9** Using the function sinc write a MATLAB program to generate the impulse response coefficients of four zero-phase lowpass filters with cutoffs at $\omega_c = 0.4\pi$ and of lengths 81, 61, 41, and 21, respectively, and then compute and plot their magnitude responses. Use the colon ":" operator to extract the impulse response coefficients of the shorter length filters from that of the length-81 filter. Examine the oscillatory behavior of the frequency responses of each filter on both sides of the cutoff frequency. What is the relation between the number of ripples and the length of the filter? What is the relation between the heights of the largest ripples and the length of the filter? How would you modify the above program to generate the impulse response coefficients of a zero-phase lowpass filter of even lengths?

El programa MATLAB que genera la respuesta de impulso, truncado a 81 muestras, de un filtro de paso bajo ideal de fase cero con un corte en $\omega_c = 0.4\pi$ y traza su respuesta de magnitud se da a continuación: NOTA: este programa hace las cuatro longitudes.

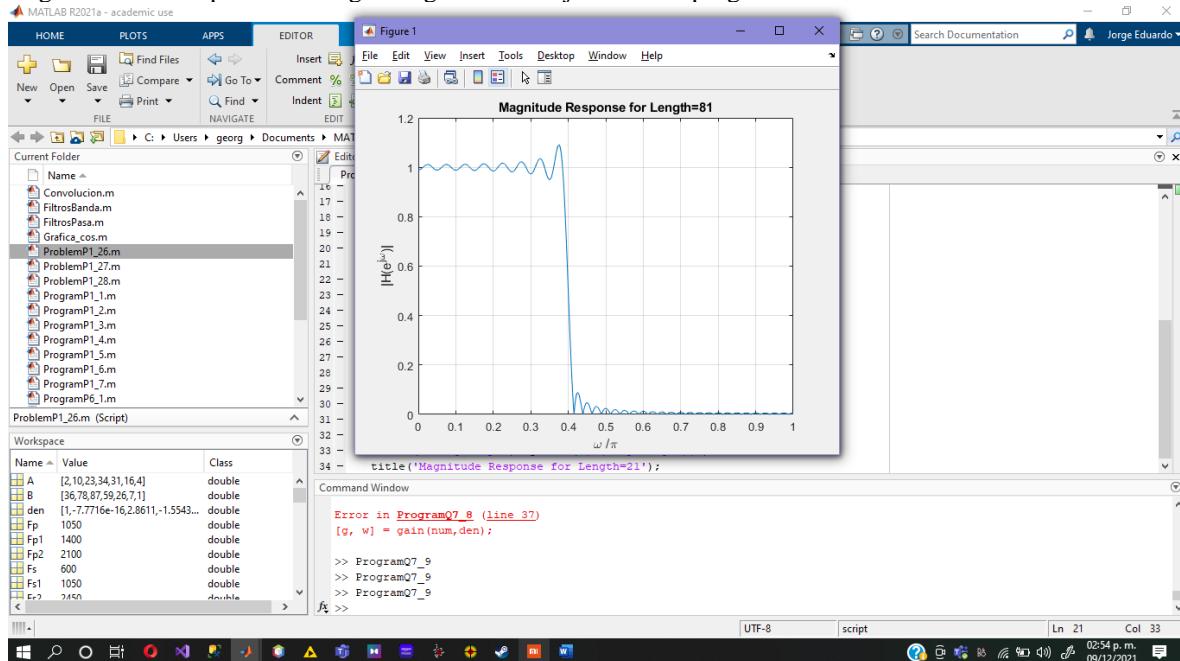
```
1. % Program Q7_9
2. % Investigate Gibbs phenomena for a FIR lowpass filter as
3. % asked for in Q7.9.
4. %%%%%%%%%%%%%%%%
5. n = -40:40; % this gives us a length of 81
6. hn_81 = 0.4 * sinc(0.4*n); % the length-81 impulse response
7. omega = 0:pi/1023:pi; % radian frequency vector
8. W = omega/pi; % Matlab normalized freq vector
9. Hz_81 = abs(freqz(hn_81,1,omega)); % 1024 samles of |H(e^jw)|
10. figure(1);
11. plot(W,Hz_81); grid;
12. xlabel('omega /pi'); ylabel('|H(e^{j\omega})|');
13. title('Magnitude Response for Length=81');
14. % Reduce length to 61 and repeat
15. hn_61 = hn_81(11:71);
16. Hz_61 = abs(freqz(hn_61,1,omega));
17. figure(2);
18. plot(W,Hz_61); grid;
19. xlabel('omega /pi'); ylabel('|H(e^{j\omega})|');
20. title('Magnitude Response for Length=61');
21. % Reduce length to 41 and repeat
22. hn_41 = hn_61(11:51);
23. Hz_41 = abs(freqz(hn_41,1,omega));
24. figure(3);
25. plot(W,Hz_41); grid;
26. xlabel('omega /pi'); ylabel('|H(e^{j\omega})|');
27. title('Magnitude Response for Length=41');
```

```

28. % Reduce length to 21 and repeat
29. hn_21 = hn_41(11:31);
30. Hz_21 = abs(freqz(hn_21, 1, omega));
31. figure(4);
32. plot(W, Hz_21); grid;
33. xlabel('omega /pi'); ylabel('|H(e^{j\omega})|');
34. title('Magnitude Response for Length=21');

```

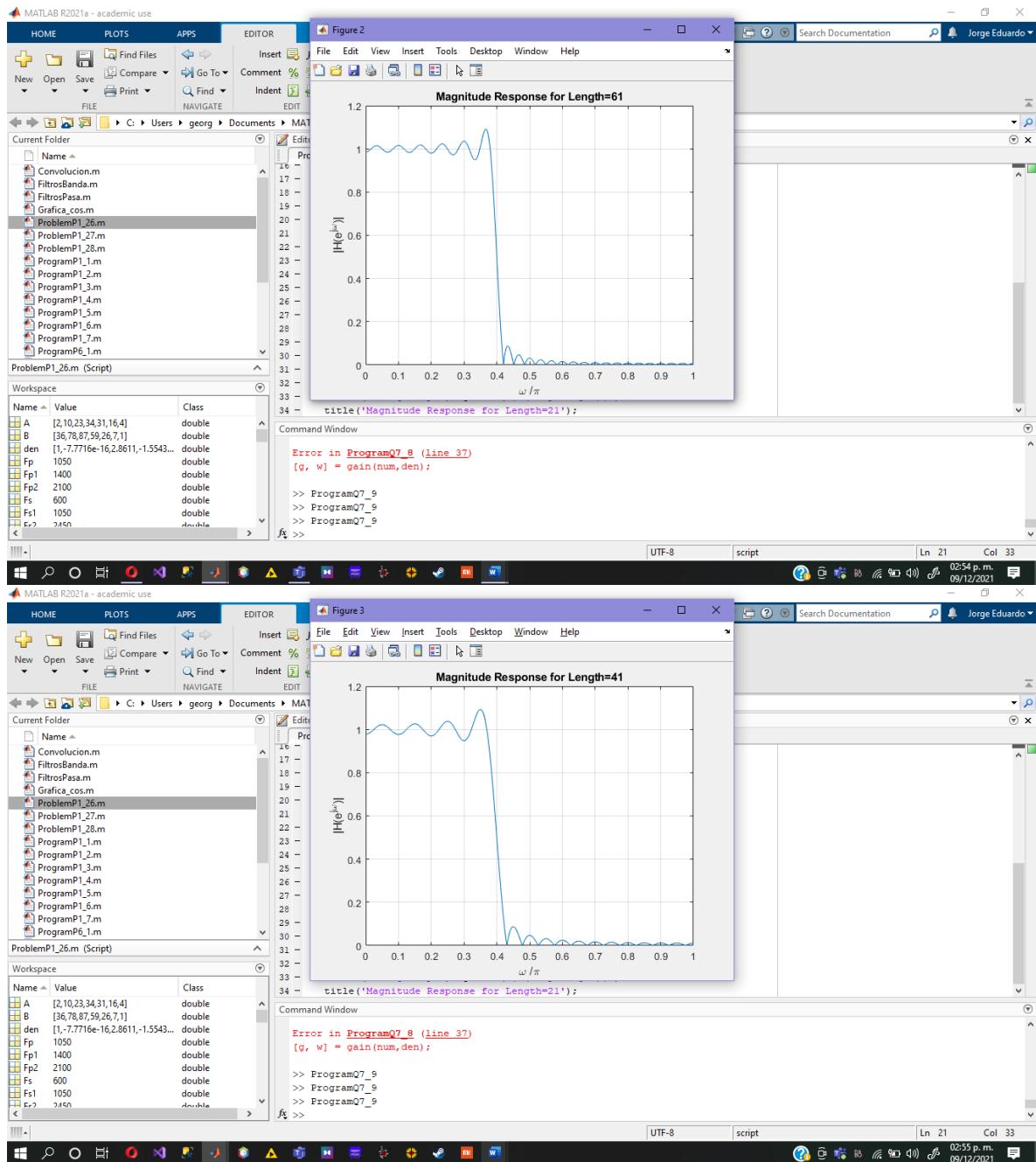
El gráfico de la respuesta de magnitud generada al ejecutar este programa es como se muestra a continuación:

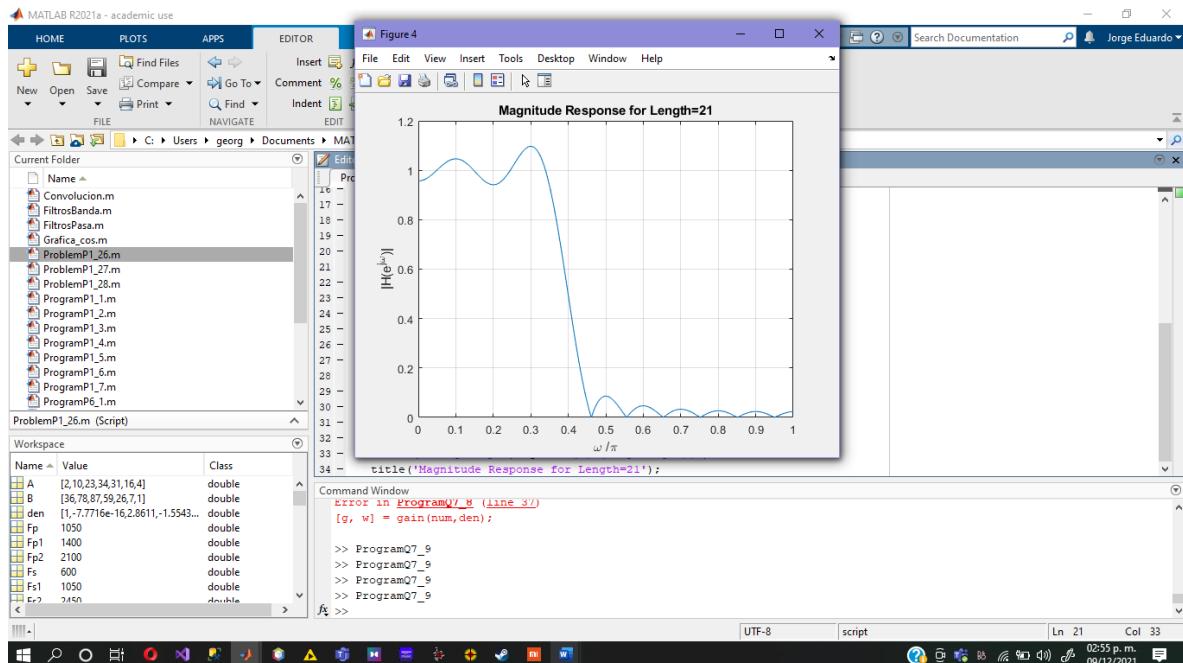


El programa se modificó como se indica a continuación para extraer los coeficientes de un filtro de longitud más corta utilizando el operador de dos puntos:

El código que se muestra arriba ya hace esto.

Las gráficas de respuesta de magnitud generadas al ejecutar el programa modificado para las siguientes longitudes, 61, 41 y 21, se dan a continuación:





A partir de estos gráficos observamos el comportamiento oscilatorio de las respuestas de magnitud en cada caso debido al fenómeno de Gibb. La relación entre el número de ondulaciones y la longitud del filtro es - El número de ondulaciones disminuye en proporción directa a la longitud.

La relación entre las alturas de las ondas más grandes y la longitud del filtro es: la altura máxima de la ondulación no se ve afectada por la longitud: permanece igual sin importar la longitud.

El programa modificado para generar coeficientes de respuesta de impulso para un filtro de longitud uniforme se da a continuación:

```

1. % Program Q7_9B
2. % Investigate Gibbs phenomena for a FIR lowpass filter as
3. % asked for in Q7.9, with an even filter length.
4. % The "trick" to make the length even is to offset the array
  "n" by 0.5
5. %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
6. n = -39.5:39.5; % this gives us a length of 80
7. hn_80 = 0.4 * sinc(0.4*n); % the length-80 impulse response
8. omega = 0:pi/1023:pi; % radian frequency vector
9. W = omega/pi; % Matlab normalized freq vector
10. Hz_80 = abs(freqz(hn_80,1,omega)); % 1024 samples of
    |H(e^jw)|
11. figure(1);
12. plot(W,Hz_80); grid;
13. xlabel('omega /pi'); ylabel('|H(e^{j\omega})|');
14. title('Magnitude Response for Length=80');
15. % Reduce length to 60 and repeat
16. hn_60 = hn_80(11:70);
17. Hz_60 = abs(freqz(hn_60,1,omega));

```

```

18. figure(2);
19. plot(W,Hz_60); grid;
20. xlabel('omega /pi'); ylabel('|H(e^{j\omega})|');
21. title('Magnitude Response for Length=60');
22. % Reduce length to 40 and repeat
23. hn_40 = hn_60(11:50);
24. Hz_40 = abs(freqz(hn_40,1,omega));
25. figure(3);
26. plot(W,Hz_40); grid;
27. xlabel('omega /pi'); ylabel('|H(e^{j\omega})|');
28. title('Magnitude Response for Length=40');
29. % Reduce length to 20 and repeat
30. hn_20 = hn_40(11:30);
31. Hz_20 = abs(freqz(hn_20,1,omega));
32. figure(4);
33. plot(W,Hz_20); grid;
34. xlabel('omega /pi'); ylabel('|H(e^{j\omega})|');
35. title('Magnitude Response for Length=20');

```

- 10) Q7.10 Using the function sinc write a MATLAB program to generate the impulse response coefficients of a zero-phase length-45 highpass filter with a cutoff at $\omega_c = 0.4\pi$ and then compute and plot its magnitude response. Examine the oscillatory behavior of the frequency responses of each filter on both sides of the cutoff frequency. How would you modify the above program to generate the impulse response coefficients of a zero-phase highpass filter of even length?

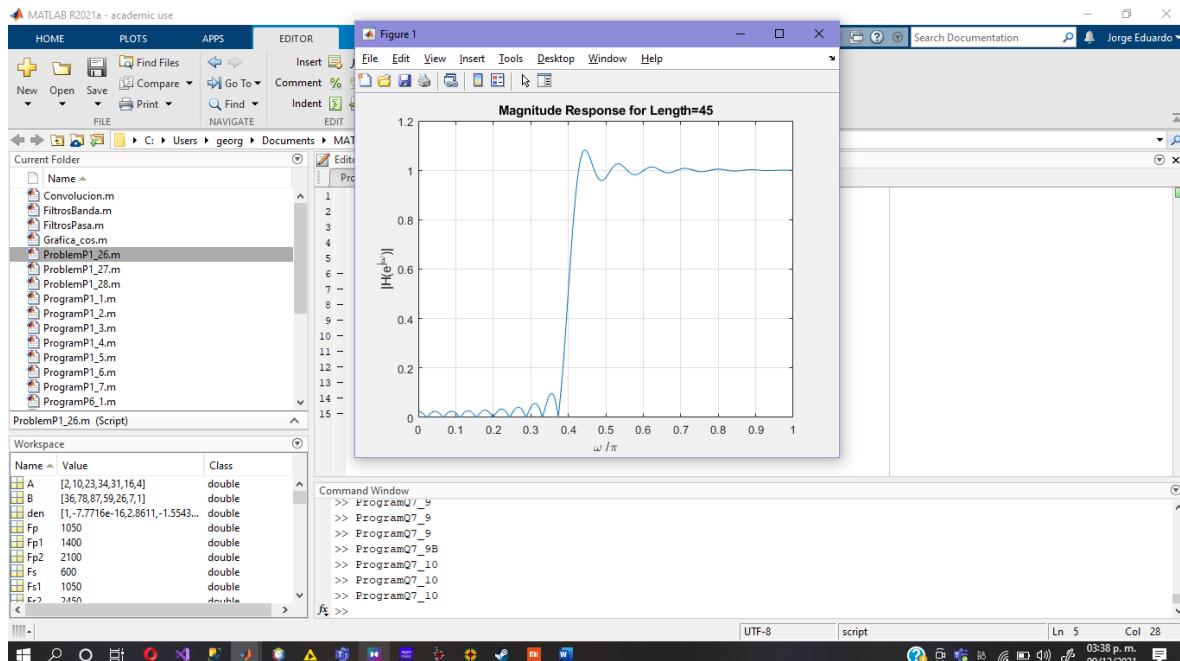
El programa MATLAB que genera la respuesta de impulso, truncada a 45 muestras, de un filtro de paso alto ideal de fase cero con un corte en $\omega_c = 0.4\pi$ y traza su respuesta de magnitud se da a continuación:

```

1. % Program Q7_10
2. % Investigate Gibbs phenomena for a FIR highpass filter.
3. % The desired impulse response is given by Eq. (7.18) in the
4. % Lab book.
5. %%%%%%%%%%%%%%
6. n = -22:22; % this gives us a length of 45
7. hn = -0.4 * sinc(0.4*n);
8. hn(23) = 0.6; % 0.6 = 1 - w_c/pi; see (7.18)
9. omega = 0:pi/1023:pi; % radian frequency vector
10. W = omega/pi; % Matlab normalized freq vector
11. Hz = abs(freqz(hn,1,omega)); % 1024 samles of |H(e^{jw})|
12. figure(1);
13. plot(W,Hz); grid;
14. xlabel('omega /pi'); ylabel('|H(e^{j\omega})|');
15. title('Magnitude Response for Length=45');

```

El gráfico de la respuesta de magnitud generada al ejecutar este programa es como se muestra a continuación:



A partir de estos gráficos, observamos el comportamiento oscilatorio de las respuestas de magnitud debido al fenómeno de Gibb.

El programa modificado para generar coeficientes de respuesta de impulso para un filtro de longitud uniforme se da a continuación:

¡Esta es una pregunta con trampa! La respuesta es: no puedes hacer la longitud ni siquiera en este caso. Este es el motivo: es un filtro FIR de fase cero. Eso significa que también es un filtro FIR de fase lineal, porque la fase cero es un subconjunto especial de la fase lineal. Ahora, la respuesta de impulso de longitud finita deseada tiene una simetría uniforme alrededor del punto medio de $h[n]$. Lo que hace este es un filtro FIR de fase lineal TIPO II. Los filtros TIPO II NO PUEDEN SER DE PASO ALTO, porque deben tener un cero en $z = -1$, que es lo mismo que $\omega = \pm \pi$.

- 11) **Q7.11 Write a MATLAB program to generate the impulse response coefficients of four zero-phase differentiators of lengths 81, 61, 41, and 21, respectively, and then compute and plot their magnitude responses. The following code fragments show how to generate a differentiator of length $2M+1$.**

```
1. n = 1:M;
2. b = cos(pi*n)./n;
3. num = [-fliplr(b) 0 b];
```

Examine the oscillatory behavior of the frequency response of the differentiator for each case. What is the relation between the number of ripples and the length of the differentiator? What is the relation between the heights of the largest ripples and the length of the filter?

El programa MATLAB que genera las muestras de respuesta al impulso de un diferenciador de fase cero de longitud $2M + 1$ y grafica su respuesta de magnitud se muestra a continuación:

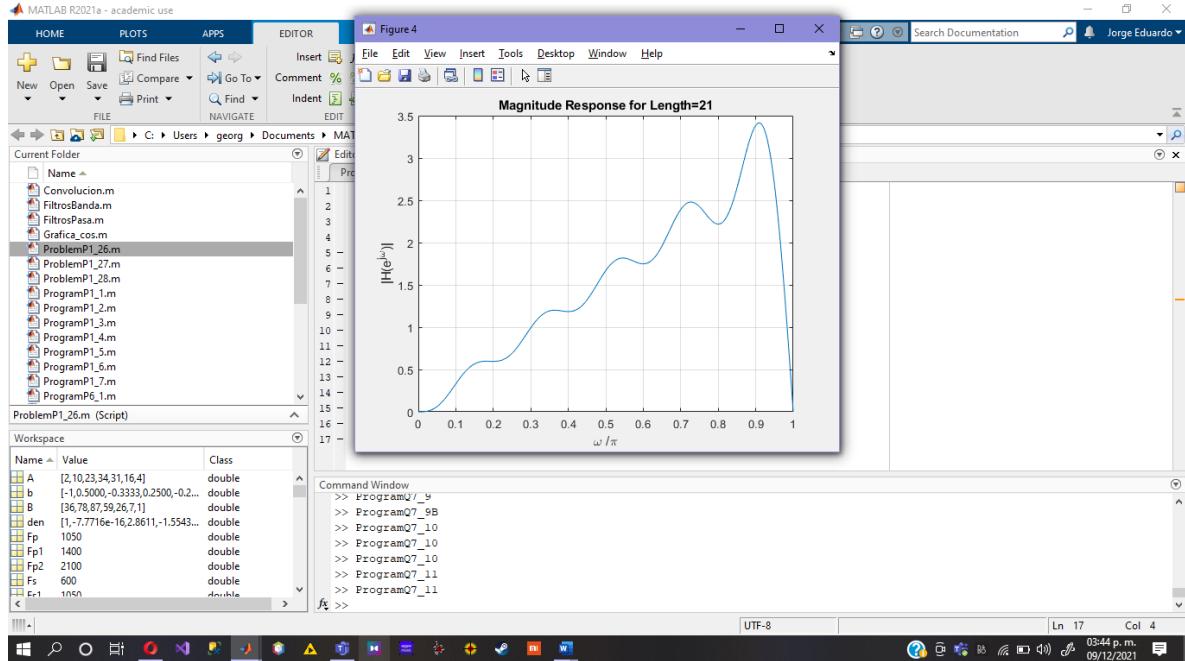
```
1. % Program Q7_11
2. % Investigate Gibbs phenomena for FIR differentiator
3. %
4. %%%%%%%%%%%%%%
5. M = [40 30 20 10]; % vector of filter half-lengths
```

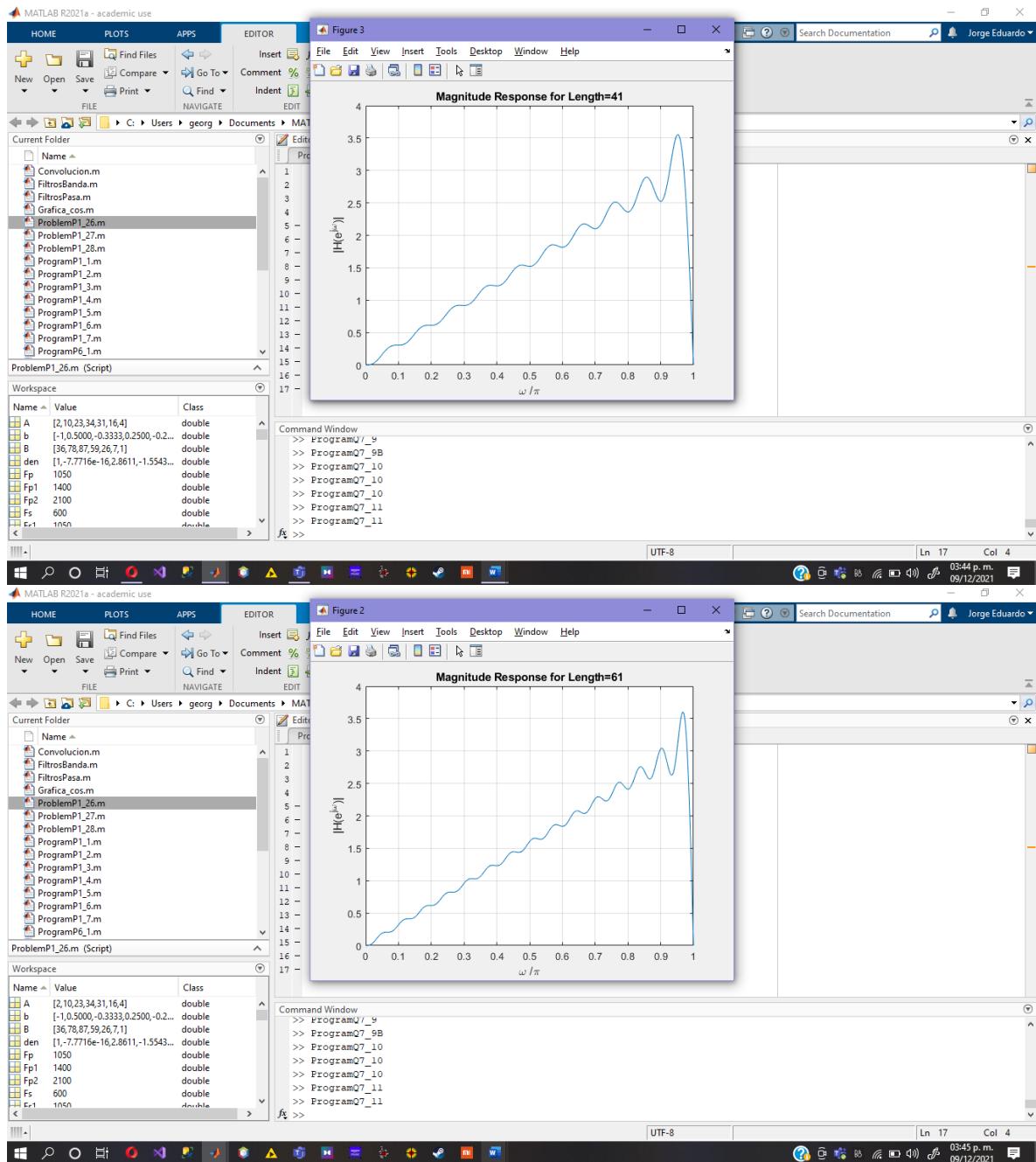
```

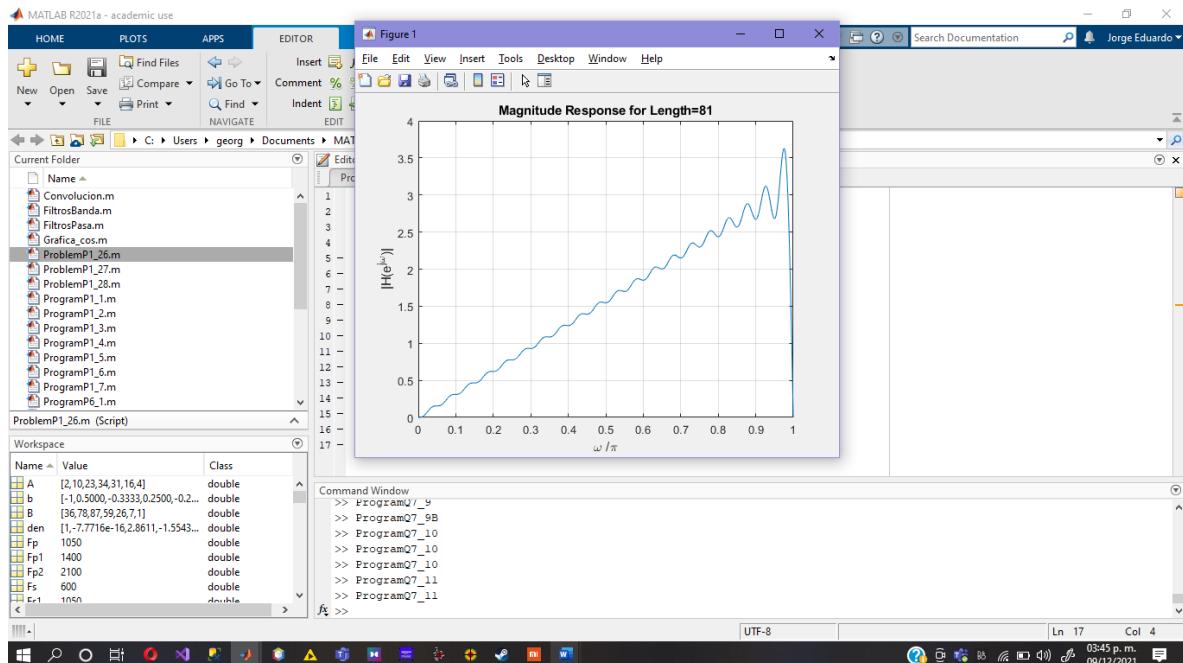
6. omega = 0:pi/1023:pi; % radian frequency vector
7. W = omega/pi; % Matlab normalized freq vector
8. for TRIAL=1:4, % loop on lengths
9. n = 1:M(TRIAL);
10. b = cos(pi*n)./n;
11. hn = [-fliplr(b) 0 b];
12. Hz = abs(freqz(hn,1,omega)); % 1024 samples of |H(e^{j\omega})|
13. figure(TRIAL);
14. plot(W,Hz);grid;
15. xlabel('omega /pi'); ylabel('|H(e^{j\omega})|');
16. title(['Magnitude Response for
Length=',int2str(2*M(TRIAL)+1)]);
17. end

```

El programa se ejecutó para los siguientes valores diferentes de longitud 81, 61, 41 y 21. A partir de los gráficos generados se observa el comportamiento oscilatorio de las respuestas de magnitud en cada caso debido al fenómeno de Gibb.







La relación entre el número de ondulaciones y la longitud del filtro es: El número de ondulaciones disminuye en proporción directa a medida que disminuye la longitud.

La relación entre las alturas de las ondas más grandes y la longitud del filtro es: la ondulación máxima no cambia con la longitud.

- 12) Q7.12 Write a MATLAB program to generate the impulse response coefficients of four discrete-time Hilbert transformers of lengths 81, 61, 41, and 21, respectively, and then compute and plot their magnitude responses. The following code fragments show how to generate a Hilbert transformer of length $2M+1$.

```

1. n = 1:M;
2. c = sin(pi*n/2);
3. b = 2*(c.*c)./(pi*n);
4. num = [-fliplr(b) 0 b];

```

Examine the oscillatory behavior of the frequency responses of the Hilbert transformer for each case. What is the relation between the number of ripples and the length of the Hilbert transformer? What is the relation between the heights of the largest ripples and the length of the filter?

El programa MATLAB que genera las muestras de respuesta al impulso de un transformador Hilbert de fase cero de longitud $2M + 1$ y traza su respuesta de magnitud se muestra a continuación.

```

1. % Program Q7_12
2. % Investigate Gibbs phenomena for FIR Hilbert xformer
3. %
4. %%%%%%%%%%%%%%
5. M = [40 30 20 10]; % vector of filter half-lengths
6. omega = 0:pi/1023:pi; % radian frequency vector
7. W = omega/pi; % Matlab normalized freq vector
8. for TRIAL=1:4, % loop on lengths

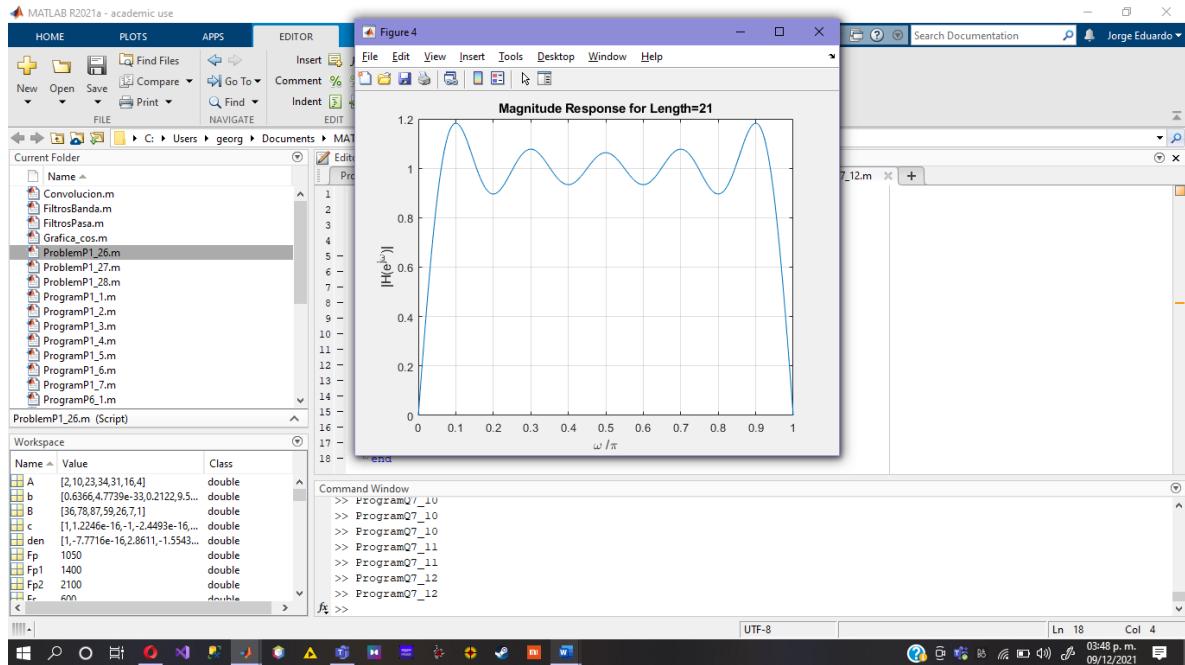
```

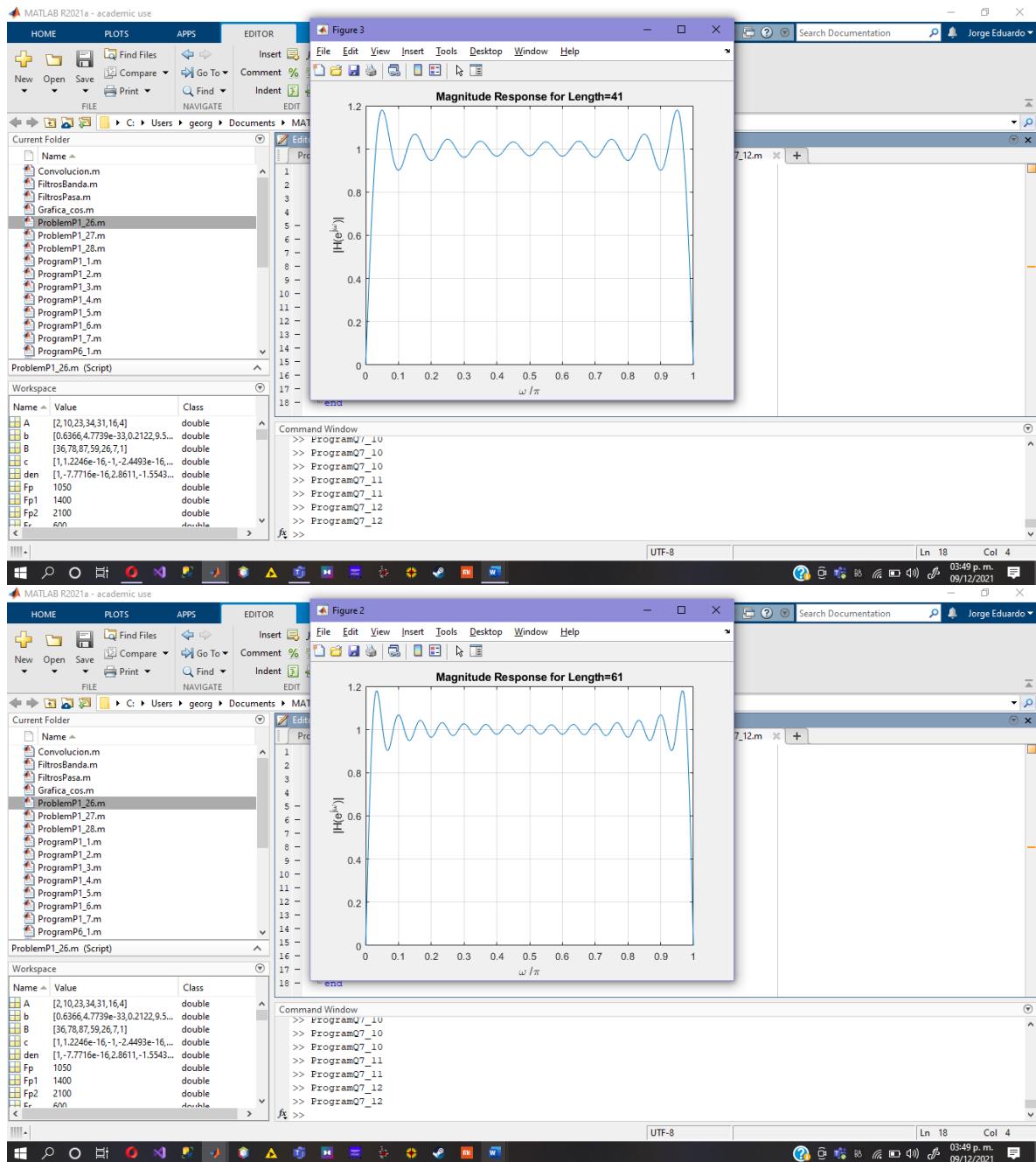
```

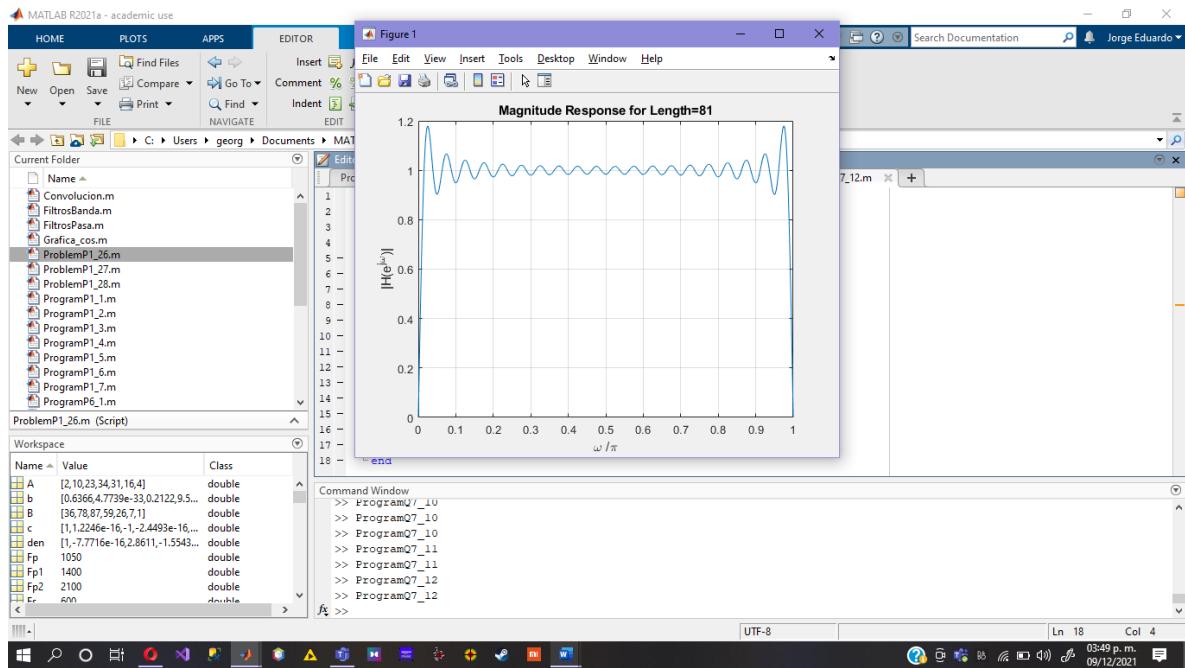
9. n = 1:M(TRIAL);
10. c = sin(pi*n*0.5);
11. b = 2*(c.*c)./(pi^n);
12. hn = [-fliplr(b) 0 b];
13. Hz = abs(freqz(hn,1,omega)); % 1024 samples of |H(e^jw)|
14. figure(TRIAL);
15. plot(W,Hz);grid;
16. xlabel('\omega /\pi'); ylabel('|H(e^{j\omega})|');
17. title(['Magnitude Response for
Length=',int2str(2*M(TRIAL)+1)]);
18. end

```

El programa se ejecutó para los siguientes valores diferentes de longitud 81, 61, 41 y 21. A partir de los gráficos generados se observa el comportamiento oscilatorio de las respuestas de magnitud en cada caso debido al fenómeno de Gibb.







La relación entre el número de ondulaciones y la longitud del filtro es: El número de ondulaciones disminuye en proporción directa a la disminución de la duración de la respuesta al impulso.

La relación entre las alturas de las ondulaciones más grandes y la longitud del filtro es: la ondulación máxima no se ve afectada por la longitud.

D. Project 7.4 Estimation of Order of FIR Filter

- **Questions:**

- 13) Q7.13 Using the function `kaiord`, estimate the order of a linear-phase lowpass FIR filter with the following specifications: passband edge = 2 kHz, stopband edge = 2.5 kHz passband ripple $\delta_p = 0.005$, stopband ripple $\delta_s = 0.005$, and sampling rate of 10 kHz. What are the purposes of the commands `ceil` and `nargin` in the function `kaiord`?

El orden estimado de un filtro FIR de paso bajo de fase lineal con las siguientes especificaciones: $\omega_p = 2 \text{ kHz}$, $\omega_s = 2.5 \text{ kHz}$, $\delta_p = 0.005$, $\delta_s = 0.005$ y $F_T = 10 \text{ kHz}$ obtenido usando `kaiord` es $-N = 46$. El valor correcto la llamada es `kaiord(2000,2500,0.005,0.005,10000)`.

El propósito del comando `ceil` es: redondear el orden estimado al siguiente entero más grande; el orden debe ser entero, por lo que si la fórmula devuelve una fracción, debe redondearse al siguiente número entero.

El propósito del comando `nargin` es: detectar si se ha llamado a `kaiord` con cuatro argumentos o con cinco. Si es cinco, se supone que todas las frecuencias son analógicas y que el último argumento es la frecuencia de muestreo. Si es cuatro, entonces la frecuencia de muestreo tiene un valor predeterminado de 2, lo que implica que los otros argumentos de frecuencia están en unidades de ciclos por muestra.

- 14) Q7.14 Repeat Question Q7.13 for the following cases: (a) sampling rate of 20 kHz, (b) $\delta_p = 0.002$ and $\delta_s = 0.002$, and (c) stopband edge = 2.3 kHz. Compare the filter length obtained in each case with that obtained in Question Q7.13. Comment on the effect of the sampling rate, ripples, and the transition bandwidth on the filter order.

- El orden estimado del filtro FIR de fase lineal con frecuencia de muestreo cambiada a $F_T = 20 \text{ kHz}$ es $-N = 91$.
- El orden estimado del filtro FIR de fase lineal con ondulaciones cambiado a $\delta_p = 0.002$ y $\delta_s = 0.002$ es $-N = 57$.
- El orden estimado del filtro FIR de fase lineal con el borde de la banda de supresión cambiado a $\omega_s = 2.3 \text{ kHz}$ es $-N = 76$.

De los resultados anteriores y del obtenido en la Pregunta P7.13 observamos que:

La relación entre el orden del filtro y la frecuencia de muestreo es la siguiente: para un ancho de banda de transición analógico dado, un aumento en la frecuencia de muestreo da como resultado un aumento proporcional en el orden estimado, hasta redondear al siguiente entero.

Esto se puede ver claramente en (7.7) en la p. 110 del manual de laboratorio. El ancho de banda de transición analógica en Hz viene dado por $|F_p - F_s|$. Sin embargo, el ancho de banda de transición en el denominador de (7.7) es digital y tiene unidades de radianes por muestra. La relación entre $|F_p - F_s|$ y $\Delta\omega$ viene dado por

$$\Delta\omega = 2\pi |F_p - F_s| / F_T.$$

Por lo tanto, aumentar F_T en un factor de 2 disminuirá $\Delta\omega$ en un factor de dos, lo que duplica el orden estimado en (7.7) hasta el redondeo al siguiente entero más grande.

La relación entre el orden del filtro y las ondulaciones es la siguiente: el orden estimado es aproximadamente proporcional al logaritmo (base 10) de las ondulaciones.

La relación entre el orden del filtro y la banda de transición es la siguiente: dentro del redondeo, el orden cambia en proporción al ancho de banda de transición. Aumentar el ancho de banda de transición en un factor de dos tiende a reducir a la mitad el orden. De manera similar, disminuir el ancho de banda de transición en un factor de dos tiende a duplicar el orden. Esto puede entenderse nuevamente por la aparición de $\Delta\omega$ en el denominador de (7.7).

- 15) Q7.15 Repeat Question Q7.13 using the function `kaiserord`. Compare the value of the filter order obtained with that obtained in Question Q7.13.

El orden estimado de un filtro FIR de paso bajo de fase lineal con las especificaciones dadas en la Pregunta Q7.13 y obtenido usando kaiserord es - N = 54. La llamada correcta es:

kaiserord ([2000 2500], [1 0], [0,005 0,005], 10000)

Al comparar el valor anterior del orden con el obtenido en la Pregunta P7.13, observamos que el orden estimado por kaiserord es menor. Esto se debe a que kaiserord utiliza una aproximación diferente para el pedido. Específicamente, kaiserord usa la aproximación dada en (7.37) en la pág. 115 del manual de laboratorio; esta aproximación normalmente se asocia con un diseño FIR de ventana donde la ventana será específicamente una ventana de Kaiser.

16) Q7.16 Repeat Question Q7.13 using the function firpmord. Compare the value of the filter order obtained with those obtained in Questions Q7.13 and Q7.15.

El orden estimado de un filtro FIR de paso bajo de fase lineal con las especificaciones dadas en la Pregunta Q7.13 y obtenido usando firpmord es - N = 47. La llamada correcta es

firpmord ([2000 2500], [1 0], [0,005 0,005], 10000)

Al comparar el valor anterior del orden con el obtenido en las Preguntas Q7.13 y Q7.15, observamos: en este caso, firpmord ofrece un resultado que es uno más grande que kaiord y aún bastante menor que kaiserord. Nuevamente, debe tenerse en cuenta que la aproximación utilizada por kaiord es bastante general; está destinado a ser utilizado para cualquier técnica de diseño de filtros FIR. Estas otras fórmulas utilizadas por kaiserord y firpmord son más especializadas.

La aproximación utilizada por firpmord está diseñada específicamente para dar una buena estimación de orden para un diseño mediante el algoritmo de Parks-McClellan.

17) Q7.17 Using the function kaiord, estimate the order of a linear-phase bandpass FIR filter with the following specifications: passband edges = 1.8 and 3.6 kHz, stopband edges 1.2 and 4.2 kHz, passband ripple $\delta_p = 0.1$, stopband ripple $\delta_s = 0.02$, and sampling rate of 12 kHz.

El orden estimado de un filtro FIR de paso de banda de fase lineal con las siguientes especificaciones: bordes de banda de paso a 1.8 y 3.6 kHz, bordes de banda de parada a 1.2 y 4.2 kHz, $\delta_p = 0.01$, $\delta_s = 0.02$ y $FT = 12$ kHz, obtenido usando kaiord es - Aquí hay otra discrepancia. En el manual de laboratorio, establece para Q7.17 que $\delta_p = 0.1$. Si usa esa cifra, entonces su llamada se ve así:

kaiord ([1800 3600], [1200 4200], 0.1,0.02,12000)

y obtienes N = 20.

Pero aquí, en el archivo del informe de laboratorio, dice $\delta_p = 0.01$. Usando en su lugar esa cifra, su llamada es

kaiord ([1800 3600], [1200 4200], 0.01,0.02,12000)

y obtiene N = 33. Como no está claro, aceptaré cualquiera de las respuestas.

18) Q7.18 Repeat Question Q7.17 using the function kaiserord. Compare the value of the filter order obtained with that obtained in Question Q7.17.

El orden estimado de un filtro FIR de paso de banda de fase lineal con las especificaciones dadas en la Pregunta Q7.17 y obtenido usando kaiserord es - Nuevamente hay discrepancia. Si usa $\delta_p = 0.1$, entonces su llamada se ve así:

kaiserord ([1200 1800 3600 4200], [0 1 0], [0.02 0.1 0.02], 12000)

y obtienes N = 37. Si, en cambio, toma $\delta_p = 0.01$, entonces su llamada es:

kaiserord ([1200 1800 3600 4200], [0 1 0], [0.02 0.01 0.02], 12000)

y obtienes $N = 45$. Aceptaré cualquiera de las respuestas.

Al comparar el valor anterior del pedido con el obtenido en la Pregunta Q7.17, observamos que los pedidos estimados por Kaiserord son mucho más altos, pero probablemente más precisos si va a hacer un diseño de ventana de Kaiser.

19) Q7.19 Repeat Question Q7.17 using the function firpmord. Compare

El orden estimado de un filtro FIR de paso de banda de fase lineal con las especificaciones dadas en la Pregunta Q7.17 y obtenido usando firpmord es - si toma $\delta p = 0.01$, entonces su llamada es

firpmord ([1200 1800 3600 4200], [0 1 0], [0.02 0.1 0.02], 12000)

y tu respuesta es $N = 22$. Si, en cambio, toma $\delta p = 0.01$, entonces su llamada es

firpmord ([1200 1800 3600 4200], [0 1 0], [0.02 0.01 0.02], 12000)

y tu respuesta es $N = 35$. Aceptaré cualquiera de las respuestas.

Al comparar el valor anterior del orden con el obtenido en las Preguntas Q7.17 y Q7.18, observamos: El orden estimado por firpmord está nuevamente entre los otros dos, lo que probablemente sea exacto para un diseño de Parks-McClellan.

E. Project 7.5 FIR Filter Design

- **Questions:**

- 20) Q7.20 Using the function **fir1**, design a linear-phase FIR lowpass filter meeting the specifications given in Question Q7.13 and plot its gain and phase responses. Use the order estimated using Kaiser's formula in Question Q7.13. Show the filter coefficients in a tabular form. Does your design meet the specifications? If it does not, adjust the filter order until the design meets the specifications. What is the order of the filter meeting the specifications?

El programa MATLAB para diseñar y trazar la ganancia y las respuestas de fase de un filtro FIR de fase lineal utilizando **fir1** se muestra a continuación. El orden de los filtros se estima usando **kaiord**. Los datos de salida son los coeficientes de filtro.

```
1. % Program Q7_20
2. % Design a linear phase Lowpass FIR Digital Filter
3. % meeting the design specification given in Q7.13.
4. % - Print out the numerator coefficients
5. % for the transfer function.
6. % - Compute and plot the gain function.
7. % - Compute and plot the phase response.
8. % - Compute and plot the unwrapped phase response.
9. %%%%%%%%%%%%%%%%
10. clear;
11. % Design spec as given in Q7.13.
12. Fp = 2*10^3;
13. Fs = 2.5*10^3;
14. FT = 10*10^3;
15. Rp = 0.005;
16. Rs = 0.005;
17. % Estimate the filter order and print to console
18. N = kaiord(Fp,Fs,Rp,Rs,FT)
19. % Design the filter; Hamming window by default
20. Wp = 2*Fp/FT; % These freqs are normalized: they go
21. Ws = 2*Fs/FT; % zero to one, not zero to pi.
22. Wn = Wp + (Ws - Wp)/2;
23. h = fir1(N,Wn);
24. % Show the Numerator Coefficients
25. disp('Numerator Coefficients are ');disp(h);
26. % Compute and plot the gain response
27. [g, w] = gain(h,[1]); % same "gain" fcn as in Lab 4
28. figure(1);
29. plot(w/pi,g);grid;
30. %axis([0 1 -60 5]);
31. xlabel('\omega /\pi'); ylabel('Gain in dB');
32. title('Gain Response');
33. % Compute the frequency response
34. w2 = 0:pi/511:pi;
35. Hz = freqz(h,[1],w2);
```

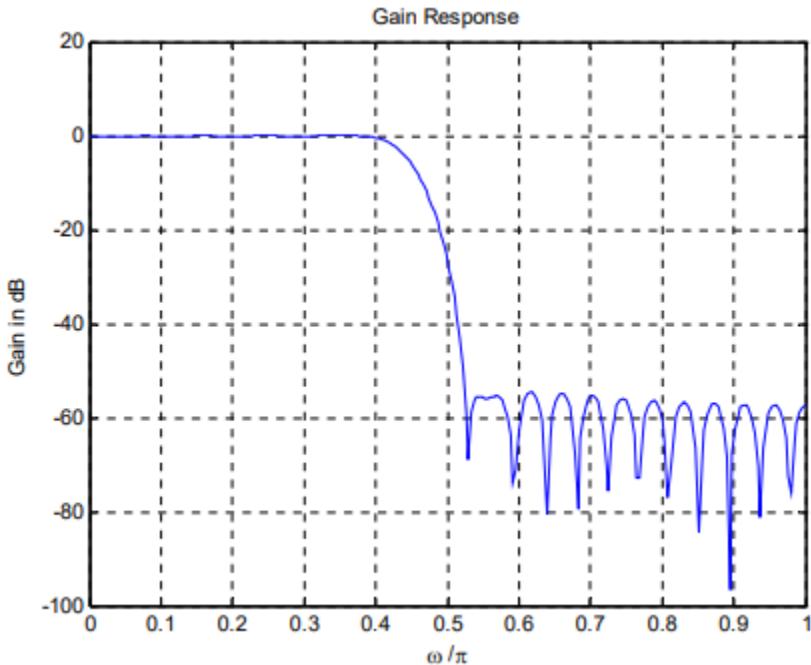
```

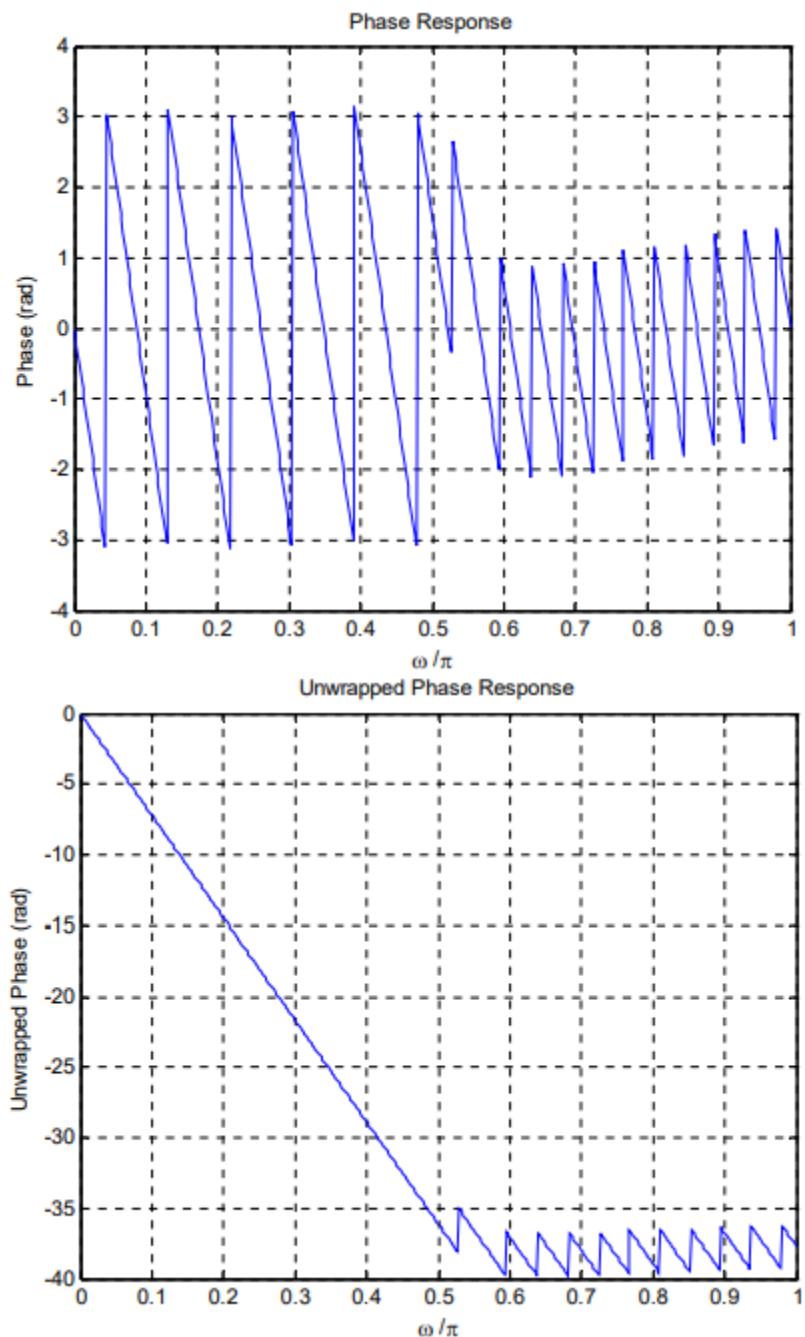
36.      % TEST: did we meet the spec?
37.      MagH = abs(Hz);
38.      T1 = 1.005*ones(1,length(w2));
39.      T2 = 0.995*ones(1,length(w2));
40.      T3 = 0.005*ones(1,length(w2));
41.      figure(4);
42.      plot(w2/pi, MagH, w2/pi, T1, w2/pi, T2, w2/pi, T3); grid;
43.      % Find and plot the phase
44.      figure(2);
45.      Phase = angle(Hz);
46.      plot(w2/pi, Phase); grid;
47.      xlabel('omega /pi'); ylabel('Phase (rad)');
48.      title('Phase Response');
49.      figure(3);
50.      UPhase = unwrap(Phase);
51.      plot(w2/pi, UPhase); grid;
52.      xlabel('omega /pi'); ylabel('Unwrapped Phase (rad)');
53.      title('Unwrapped Phase Response');

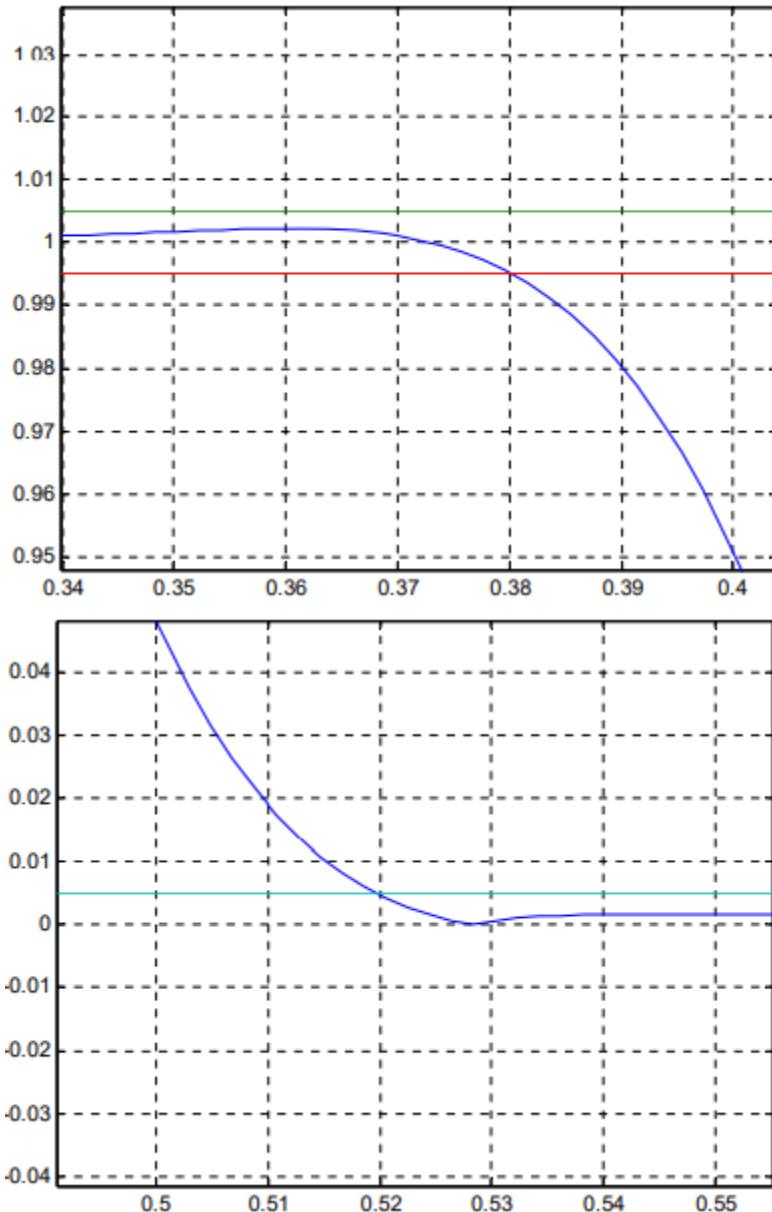
```

Los coeficientes del filtro de paso bajo correspondientes a las especificaciones dadas en la Pregunta 7.20 son los que se muestran a continuación.

Las respuestas de fase y ganancia generadas se dan a continuación:





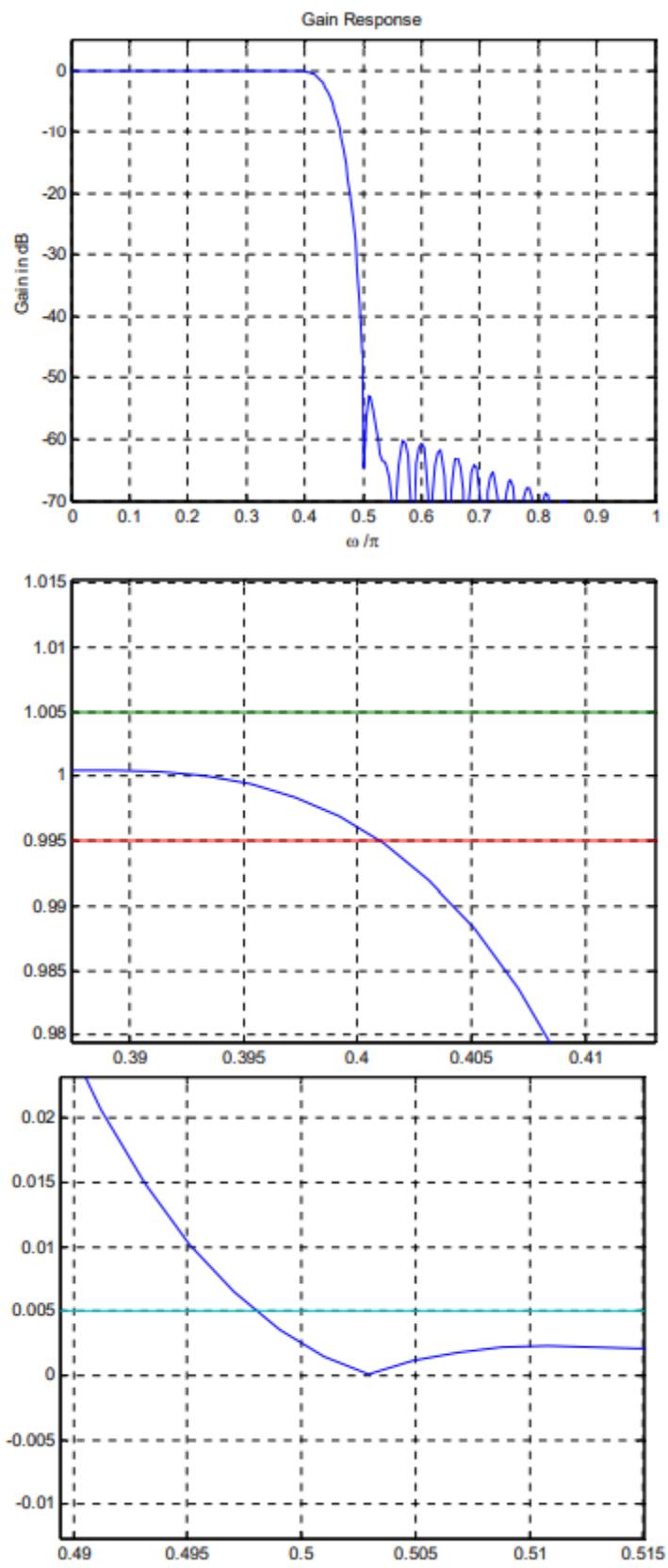


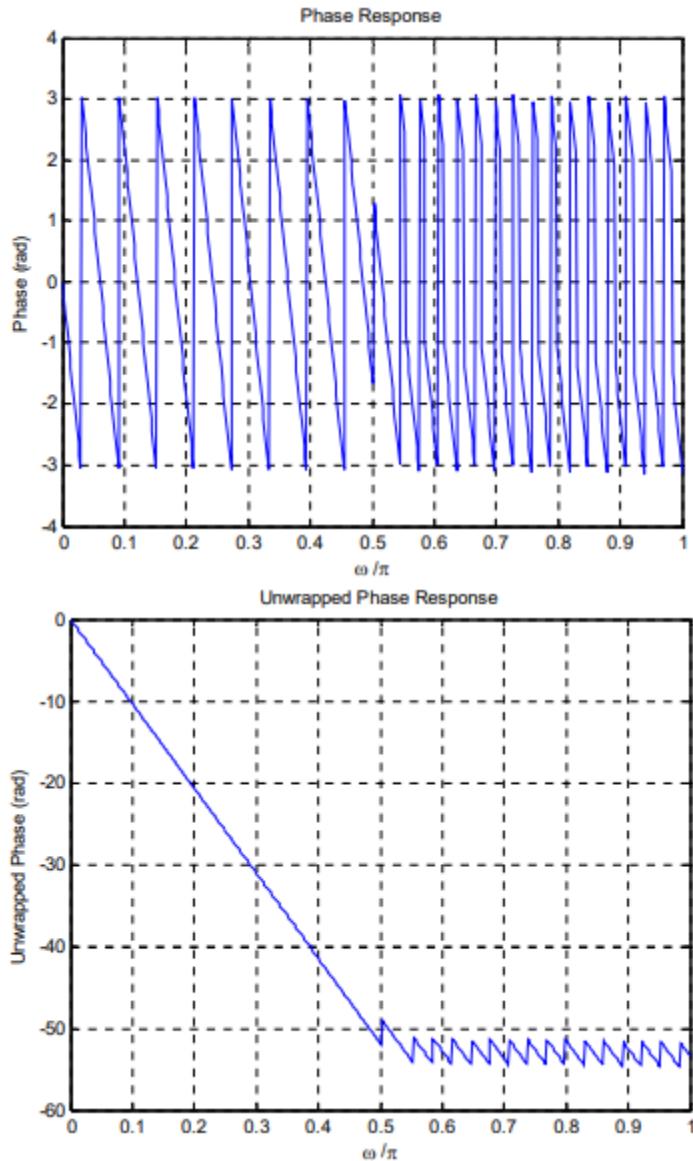
Del gráfico de ganancia observamos que el filtro diseñado NO cumple con las especificaciones.

COMO SE MUESTRA en los dos gráficos de detalle anteriores, con $N = 46$, no se cumplen ni la especificación de banda de paso en $W_p = 0,4$ (frecuencia normalizada) ni la especificación de banda de supresión en $W_s = 0,5$ (frecuencia normalizada). Entonces este diseño NO cumple con las especificaciones.

El orden de filtrado que cumple con las especificaciones es - $N = 66$

Para el filtro que CUMPLE CON LAS ESPECIFICACIONES, aquí están las gráficas:





- 21) Q7.21 Repeat Question Q7.20 using each of the following windows: Hanning, Blackman, and Dolph–Chebyshev windows.

El programa MATLAB de la pregunta Q7.20 se modificó como se indica a continuación para usar ventanas diferentes a la ventana predeterminada de Hamming.

1. % Program Q7_21
2. % Design a linear phase Lowpass FIR Digital Filter
3. % meeting the design specification given in Q7.13. Use other
4. % than Hamming window.
5. % - Print out the numerator coefficients
6. % for the transfer function.
7. % - Compute and plot the gain function.
8. % - Compute and plot the phase response.

```

9. % - Compute and plot the unwrapped phase response.
10.    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
11.    clear;
12.    % Design spec as given in Q7.13.
13.    Fp = 2*10^3;
14.    Fs = 2.5*10^3;
15.    FT = 10*10^3;
16.    Rp = 0.005;
17.    Rs = 0.005;
18.    % Estimate the filter order and print to console
19.    N = kaiord(Fp,Fs,Rp,Rs,FT);
20.    N = N+13
21.    % Design the filter
22.    Wp = 2*Fp/FT; % These freqs are normalized: they go
23.    Ws = 2*Fs/FT; % zero to one, not zero to pi.
24.    Wn = Wp + (Ws - Wp)/2;
25.    %h = fir1(N,Wn); % Default Hamming window
26.    %Wdw = hann(N+3);Wdw=Wdw(2:N+2); % Hann; see footnote on
      p. 563 of text.
27.    %Wdw = blackman(N+3);Wdw=Wdw(2:N+2); % Blackman; see
      text p. 563.
28.    %%%%%% Dolph-Chebyshev Section
      %%%%%%
29.    Wdw = chebwin(N+1,38);
30.    figure(5);
31.    plot([0:1/511:1],abs(freqz(Wdw,[1],0:pi/511:pi)));grid;
      % plot window
32.    xlabel('omega
      /pi');ylabel('|Window|');title('Window');
33.    %%%%%% end Dolph Cheby window
      %%%%%%
34.    h = fir1(N,Wn,Wdw);
35.    % Show the Numerator Coefficients
36.    disp('Numerator Coefficients are ');disp(h);
37.    % Compute and plot the gain response
38.    [g, w] = gain(h,[1]);
39.    figure(1);
40.    plot(w/pi,g);grid;
41.    %axis([0 1 -60 5]);
42.    xlabel('omega /pi'); ylabel('Gain in dB');
43.    title('Gain Response');
44.    % Compute the frequency response
45.    w2 = 0:pi/511:pi;
46.    Hz = freqz(h,[1],w2);
47.    % TEST: did we meet the spec?

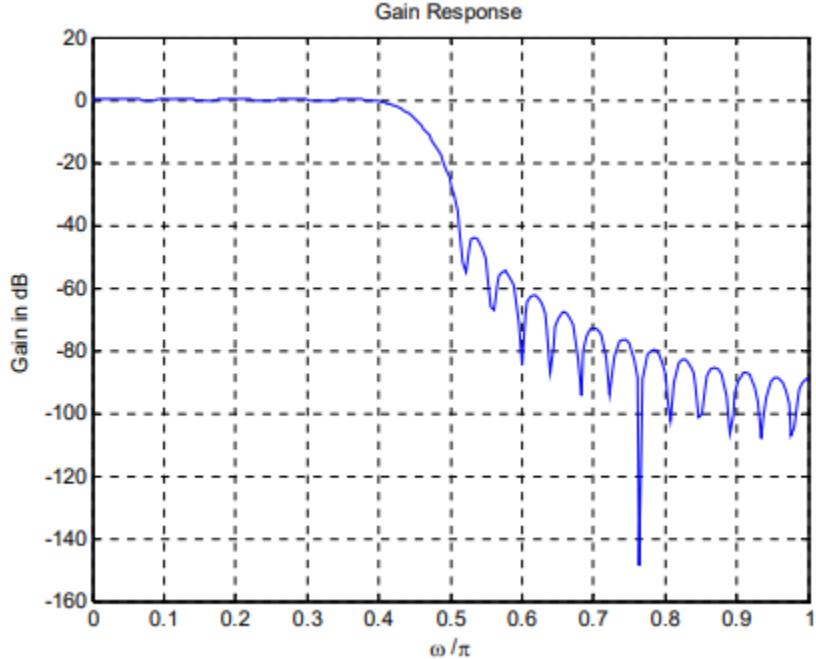
```

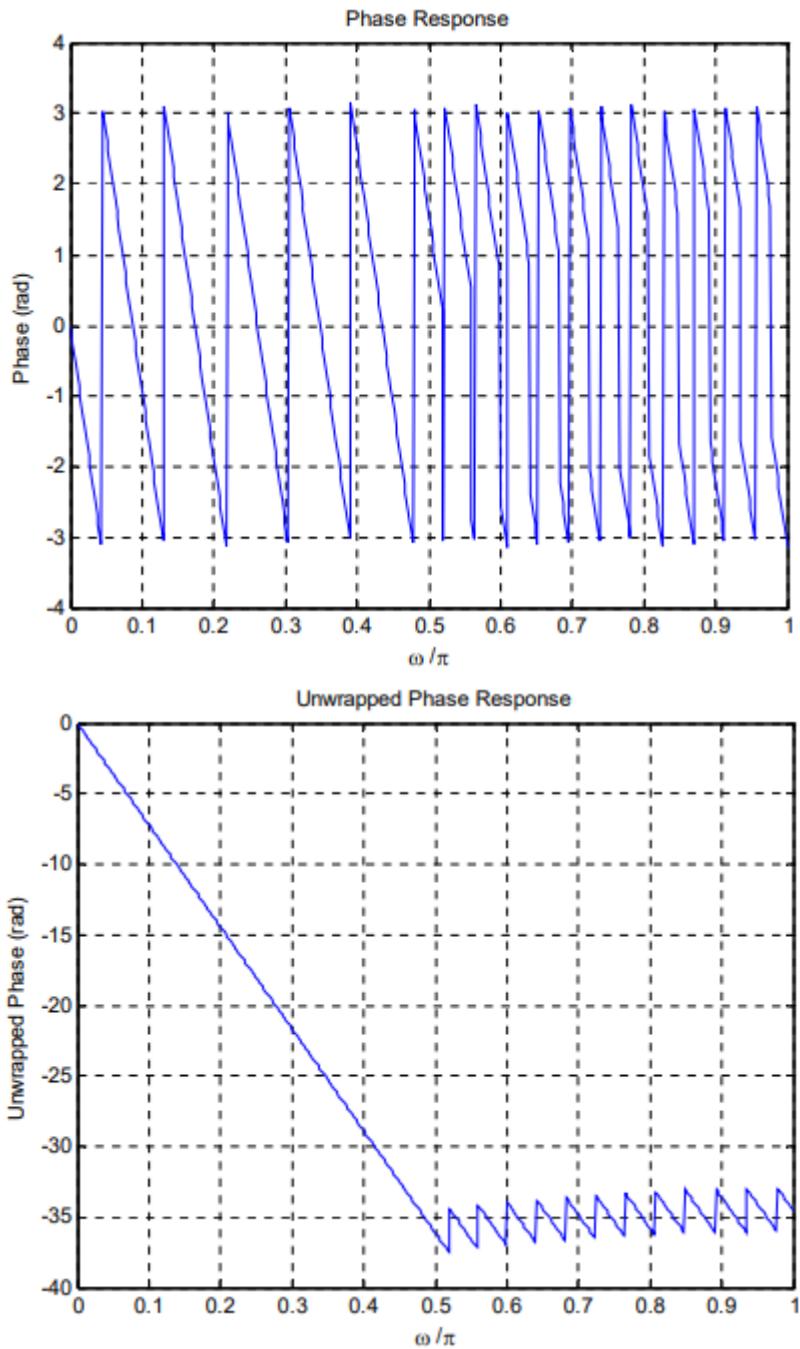
```

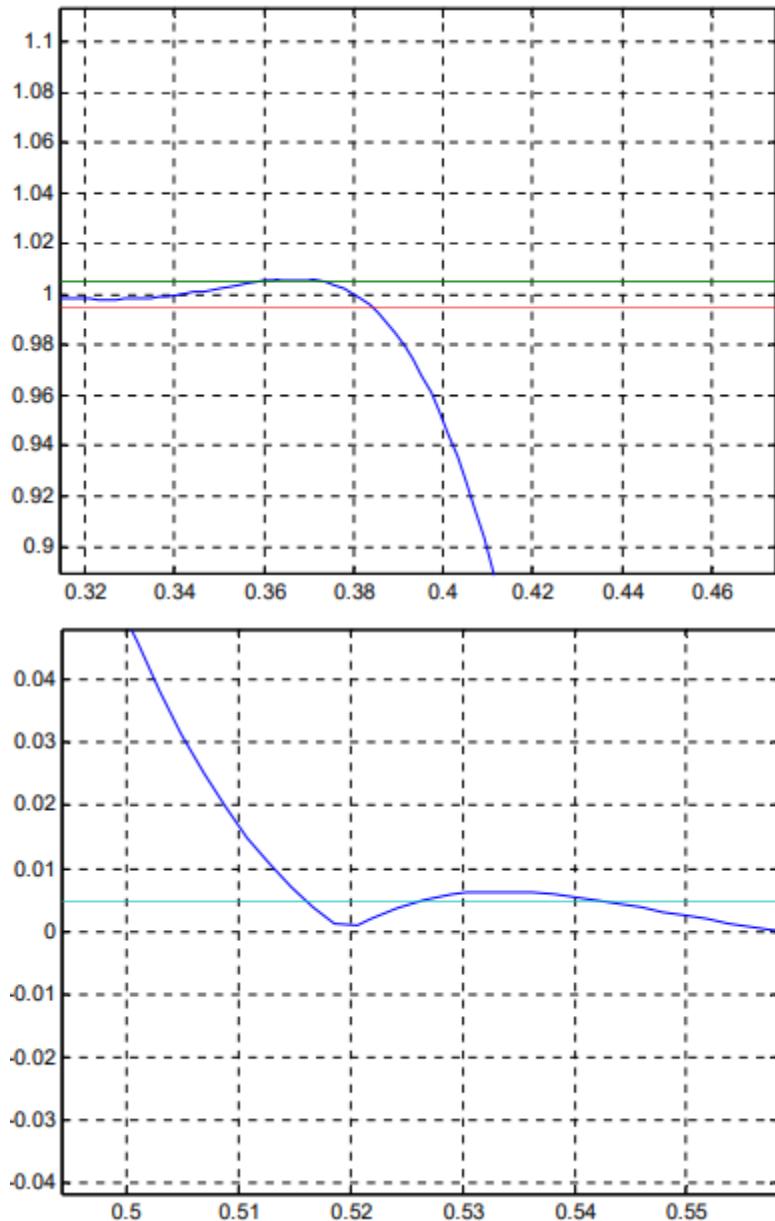
48. MagH = abs(Hz);
49. T1 = 1.005*ones(1,length(w2));
50. T2 = 0.995*ones(1,length(w2));
51. T3 = 0.005*ones(1,length(w2));
52. figure(4);
53. plot(w2/pi, MagH, w2/pi, T1, w2/pi, T2, w2/pi, T3); grid;
54. % Find and plot the phase
55. figure(2);
56. Phase = angle(Hz);
57. plot(w2/pi, Phase); grid;
58. xlabel('omega /pi'); ylabel('Phase (rad)');
59. title('Phase Response');
60. figure(3);
61. UPhase = unwrap(Phase);
62. plot(w2/pi, UPhase); grid;
63. xlabel('omega /pi'); ylabel('Unwrapped Phase (rad)');
64. title('Unwrapped Phase Response');

```

a) Uso de la ventana de Hanning: las respuestas de fase y ganancia generadas se indican a continuación:





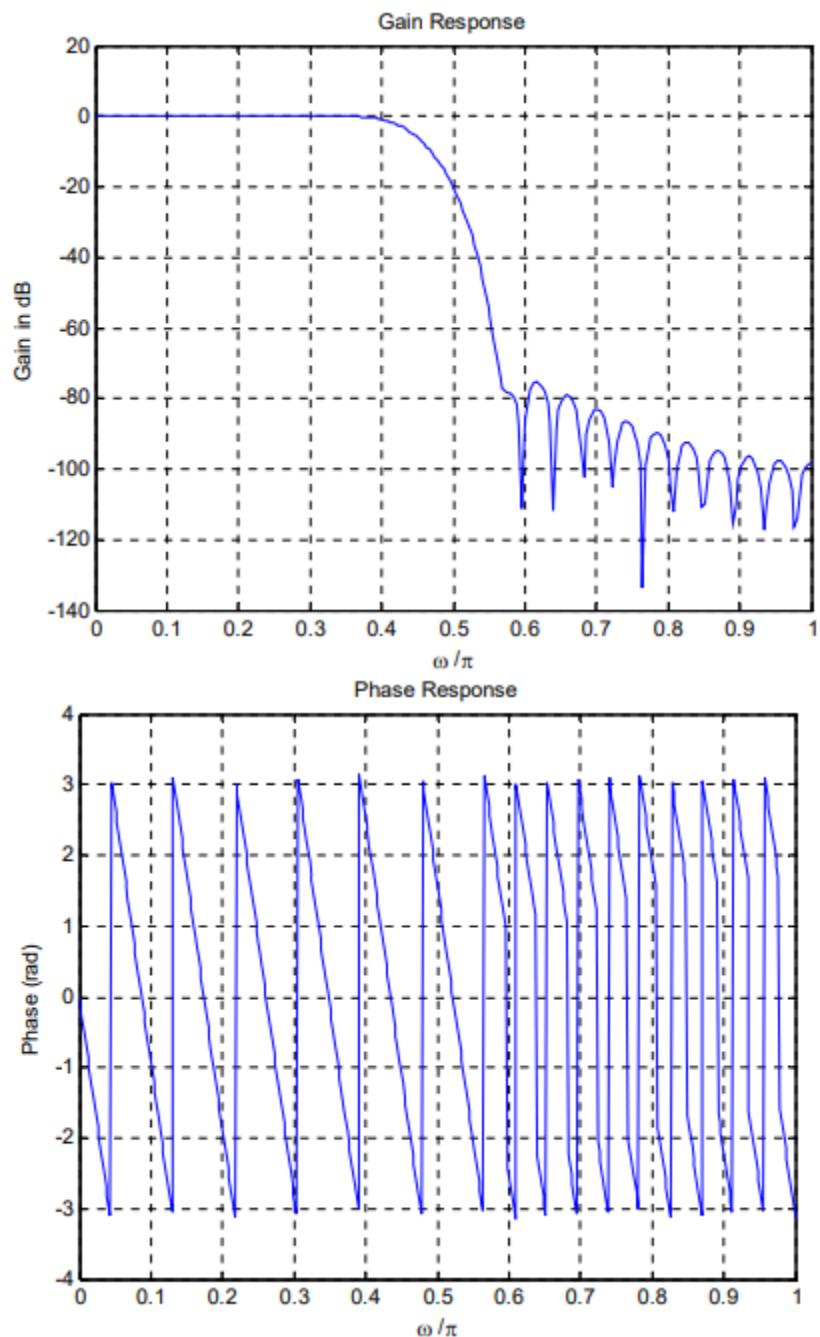


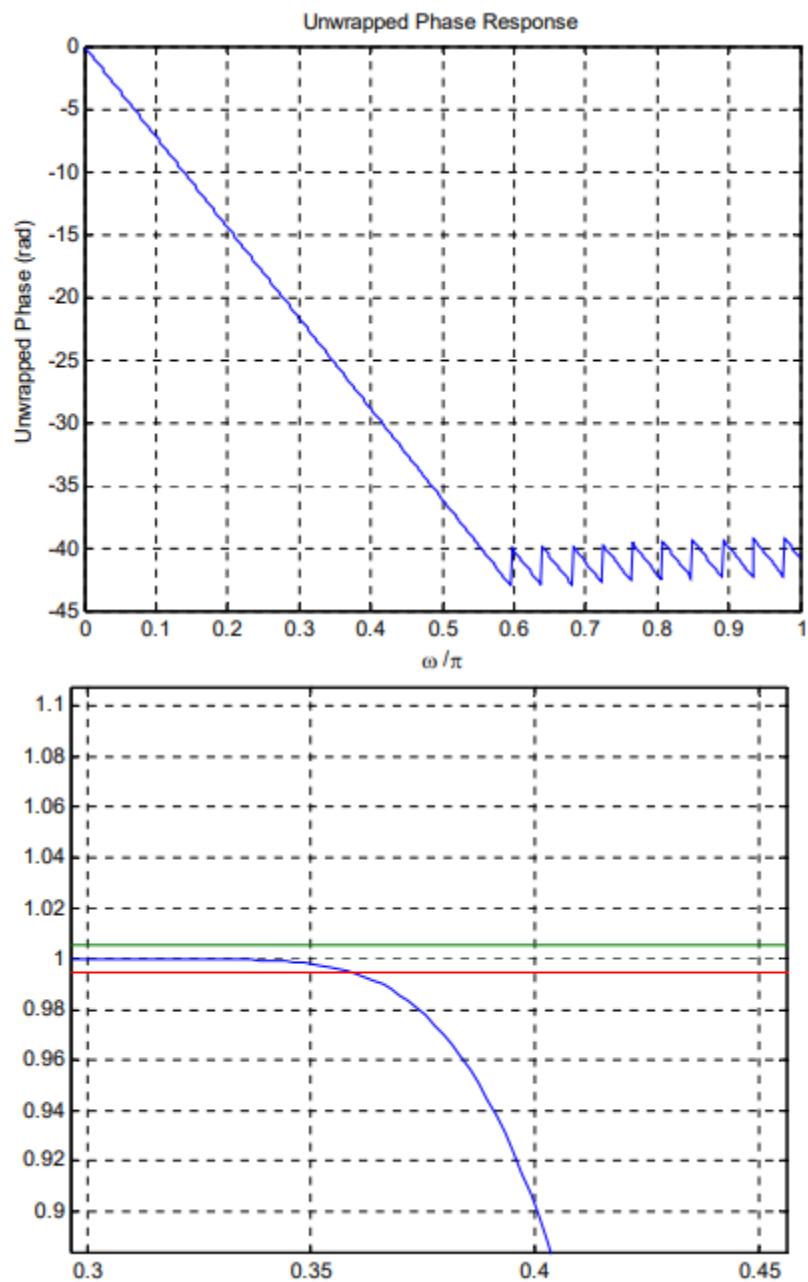
De la gráfica de ganancia observamos que el filtro diseñado NO cumple con las especificaciones. COMO SE MUESTRA en los dos gráficos de detalle anteriores, con $N = 46$, no se cumplen ni la especificación de banda de paso en $wp = 0,4$ (frecuencia normalizada) ni la especificación de banda de supresión en $ws = 0,5$ (frecuencia normalizada). **Entonces este diseño NO cumple con las especificaciones.**

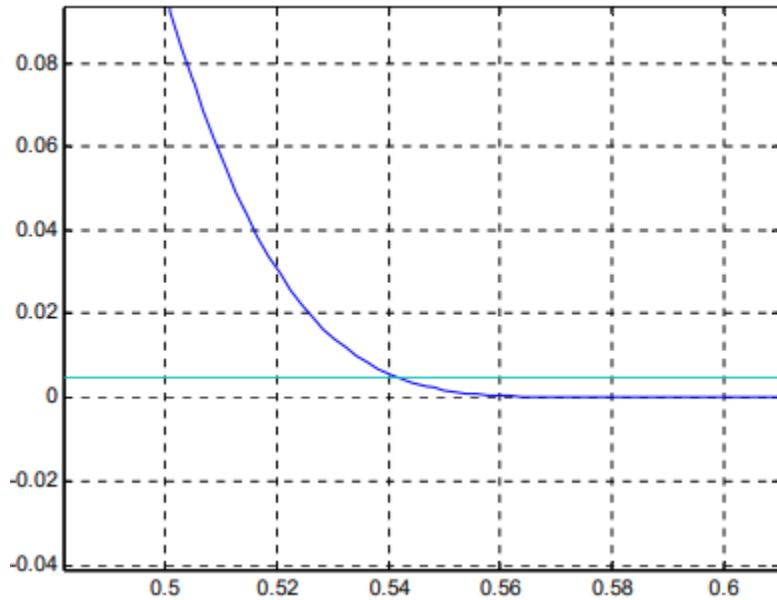
El orden de filtro que cumple con las especificaciones es: **¡¡OTRA PREGUNTA CON TRUCO !!** La ventana de Hann NO PUEDE CUMPLIR CON ESTA ESPECIFICACIÓN.

Este es el porqué: La atenuación mínima de la banda de supresión viene dada por $\alpha_s = -20\log \delta_p = 46.0206 \text{ dB}$. Sin embargo, si observa la Tabla 10.2 en la página 535 del texto, verá que la ventana de Hann es capaz de proporcionar una atenuación de banda cerrada de sólo 43.9 dB por ejemplo, **¡no es suficiente!**

b) Uso de la ventana de Blackman: las respuestas de fase y ganancia generadas se indican a continuación:





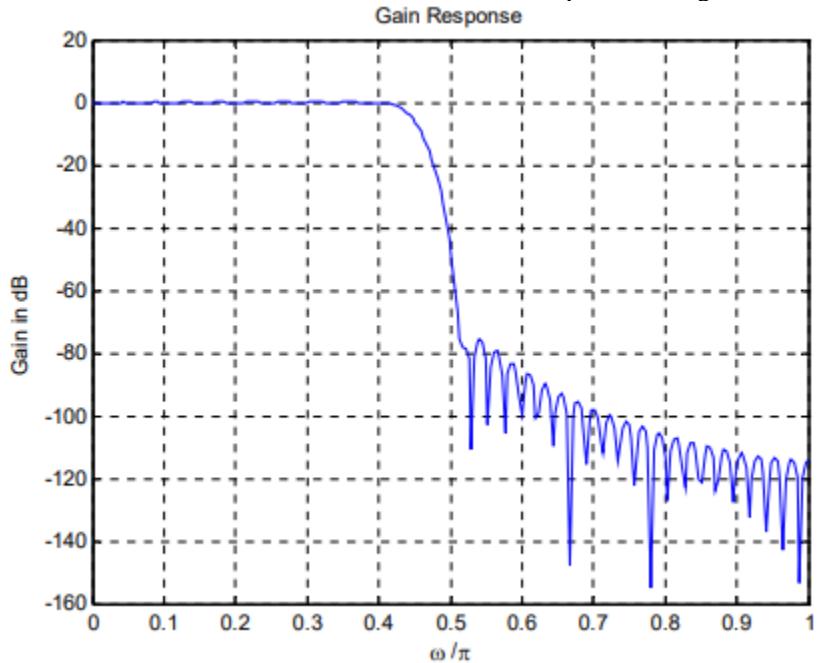


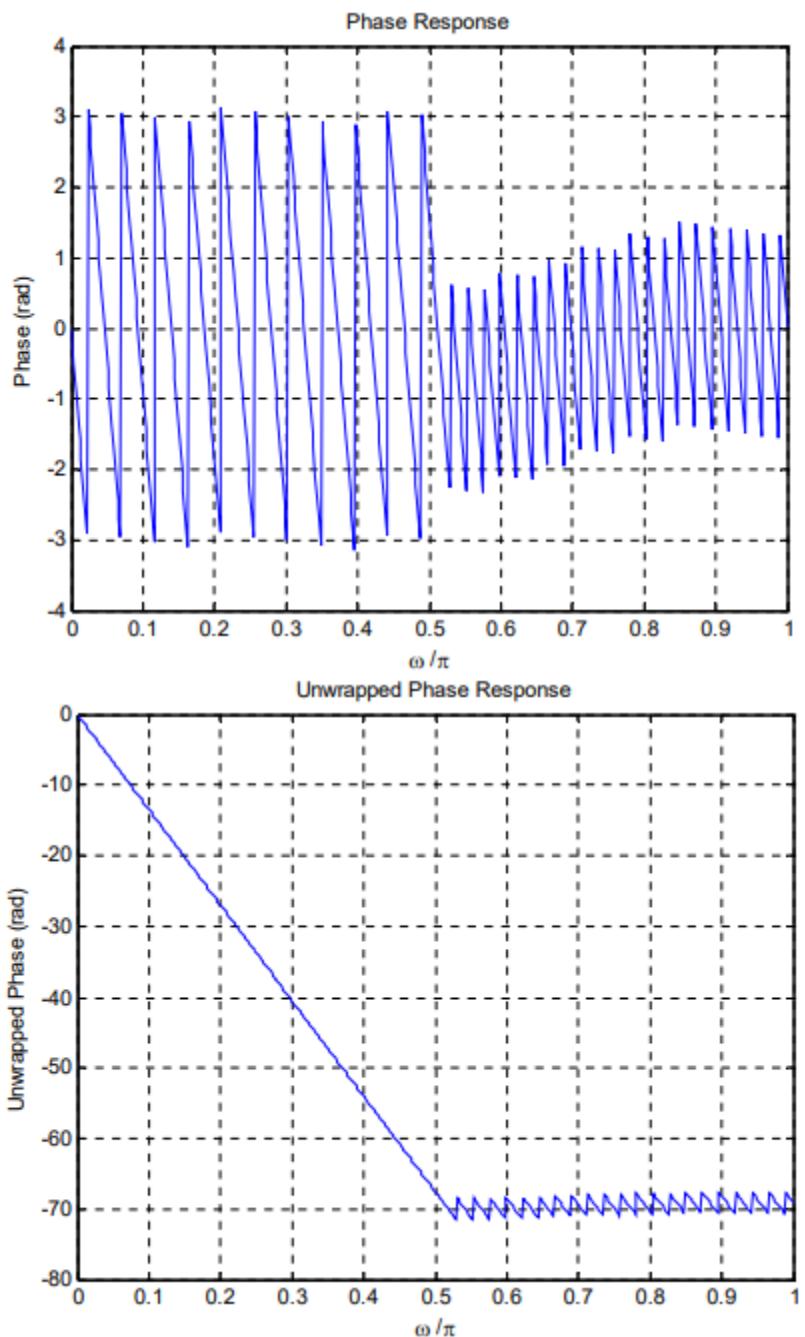
Del gráfico de ganancia observamos que el filtro diseñado NO cumple con las especificaciones.

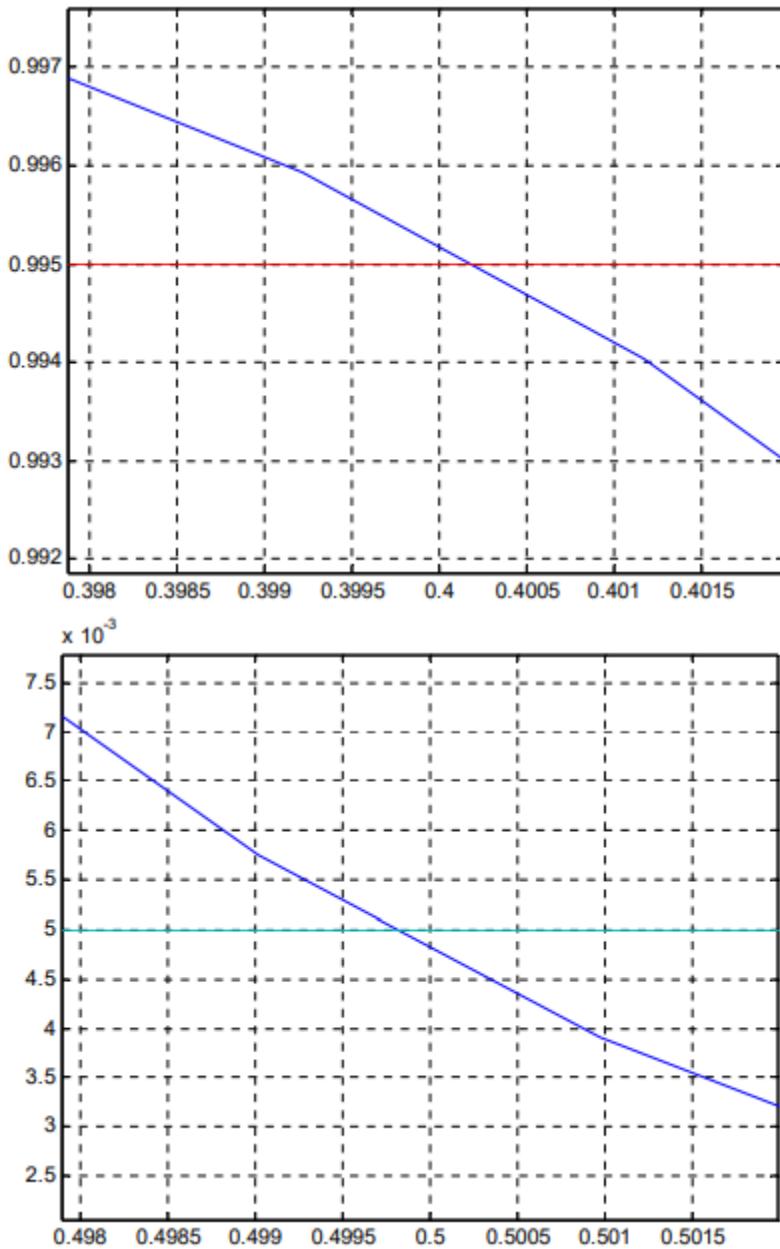
COMO SE MUESTRA en los dos gráficos de detalle anteriores, con $N = 46$, no se cumplen ni la especificación de banda de paso en $wp = 0,4$ (frecuencia normalizada) ni la especificación de banda de supresión en $ws = 0,5$ (frecuencia normalizada). **Entonces este diseño NO cumple con las especificaciones.**

El orden de filtrado que cumple con las especificaciones es - $N = 86$

Para el filtro que CUMPLE CON LAS ESPECIFICACIONES, aquí están las gráficas:



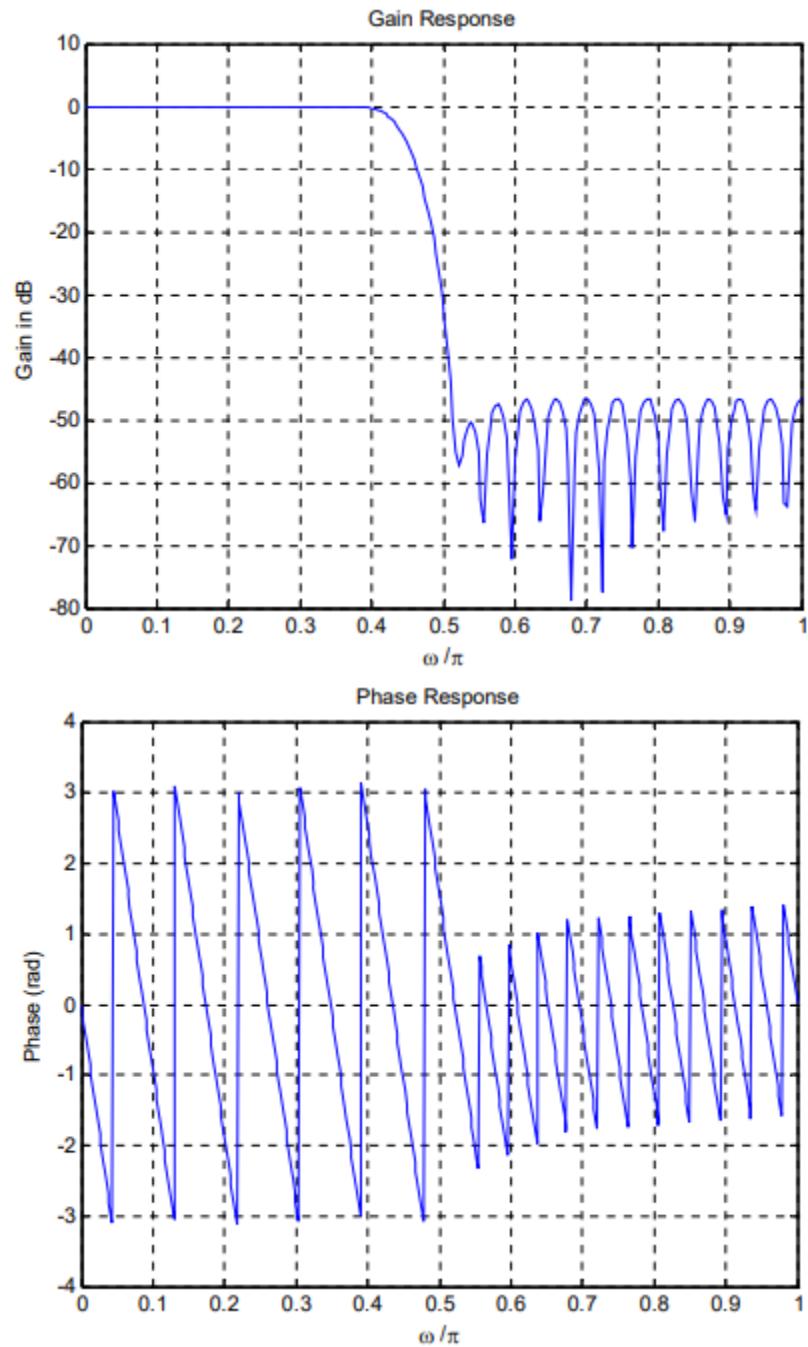


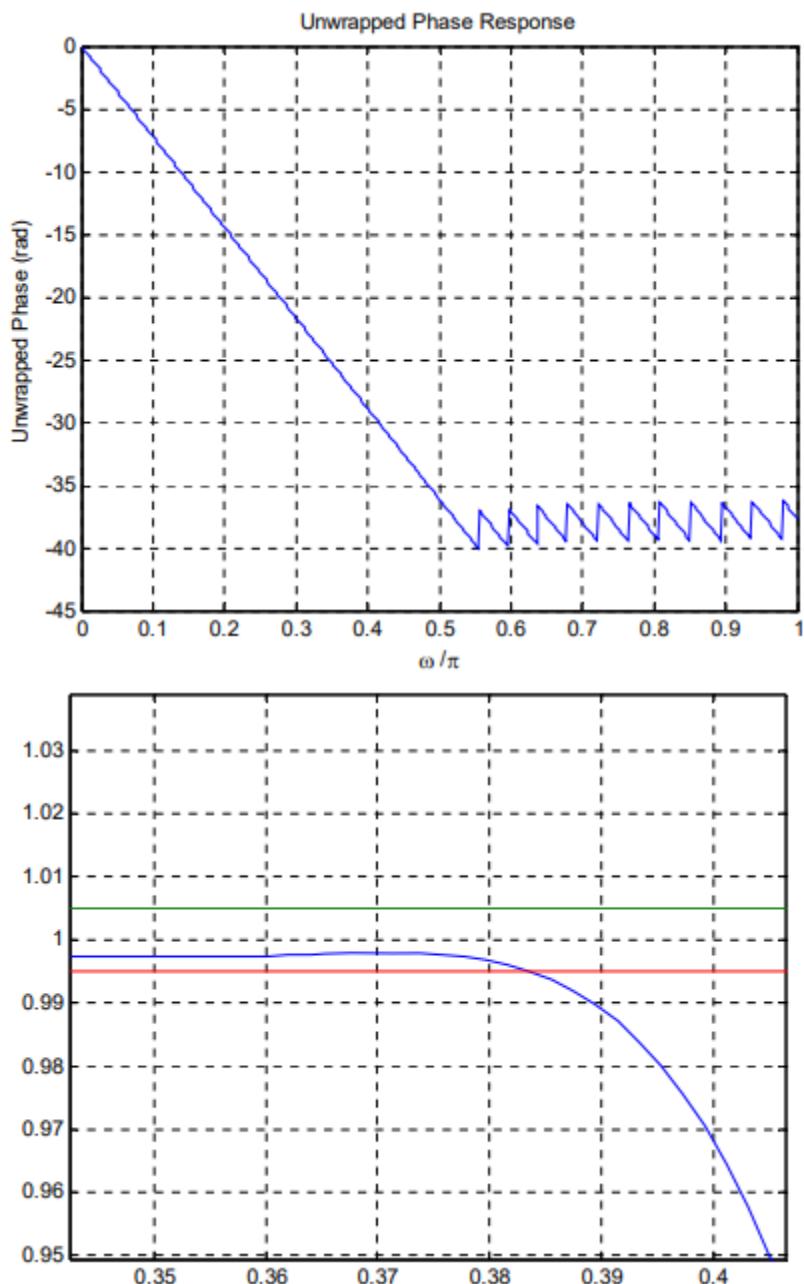


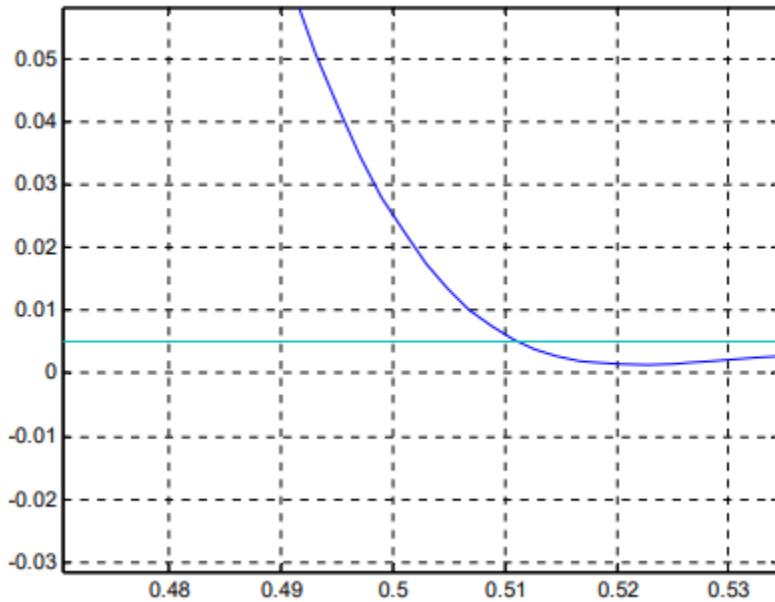
- c) Uso de la ventana Dolph-Chebyshev: las respuestas de fase y ganancia generadas se indican a continuación:

He ajustado empíricamente el parámetro de atenuación del lóbulo lateral "chebwin" R hasta que la ondulación de la banda de paso y de la banda de detención cumplan con la especificación. Esto resultó en $R = 38$ dB.

Con este valor R , luego llamo a `chebwin` y ejecuto el programa con el orden $N = 46$ diseñado por `kaiord`





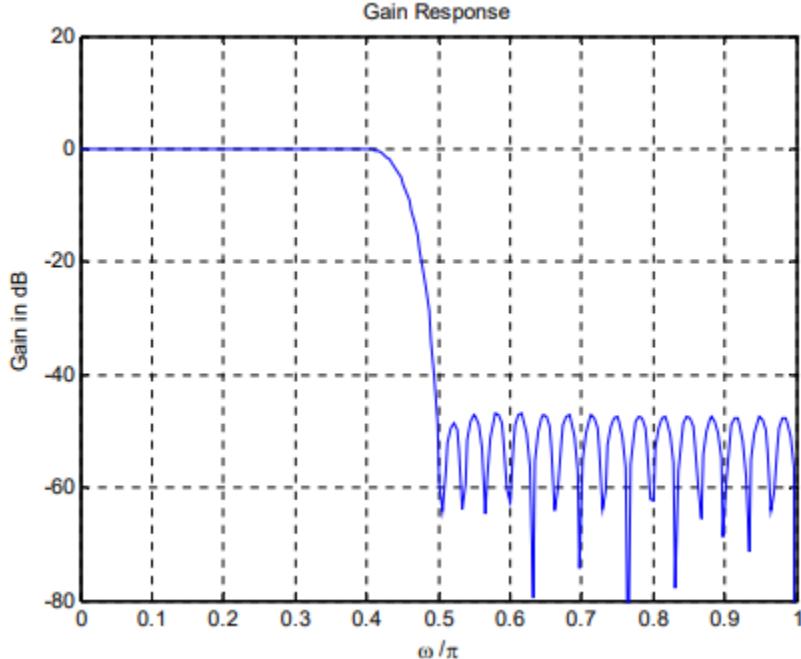


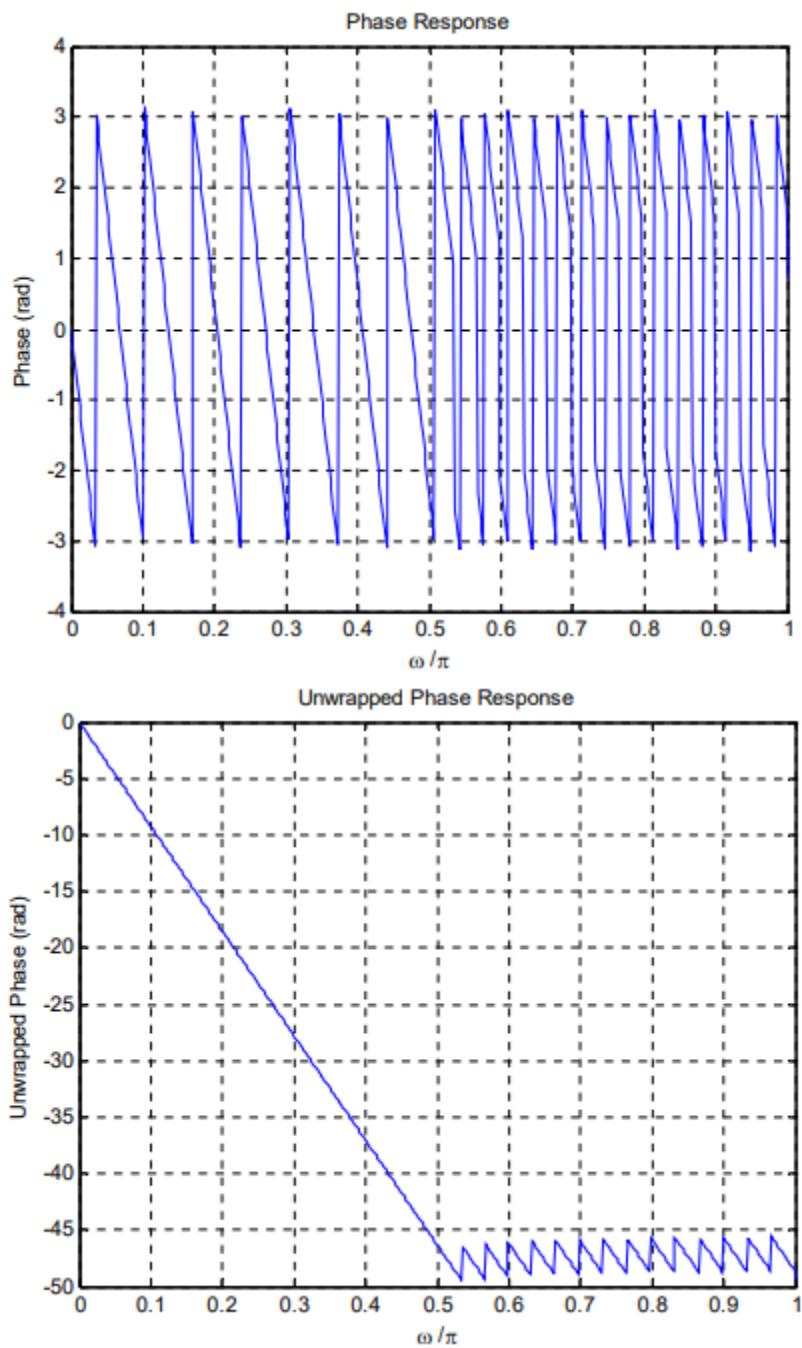
Los parámetros γ y β usados en el diseño son - Tenemos $-20 \log \gamma = 38$, lo que implica que $\gamma = 10^{-38/20} = 12.59 \times 10^{-3}$. Por tanto, también tenemos de (7.32) en la p. 115 del manual de laboratorio con $2M = 46$; $\beta = 1.00608$.

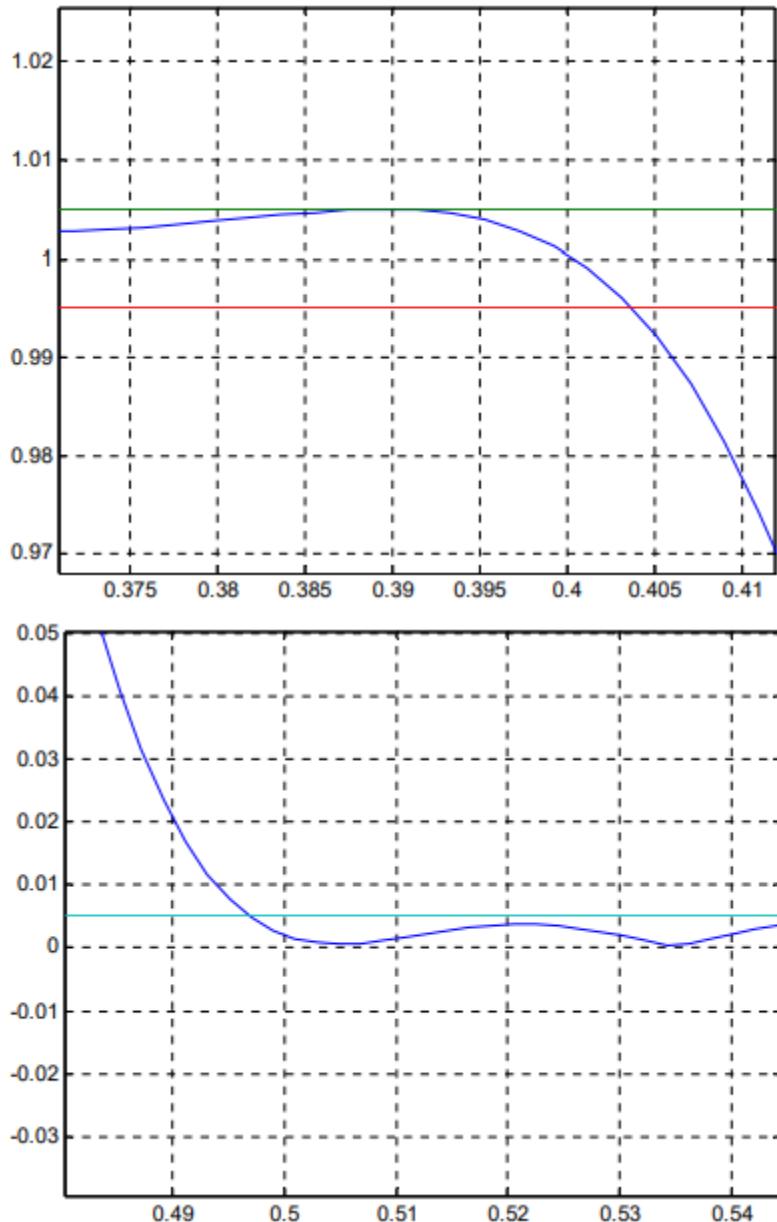
Del gráfico de ganancia observamos que el filtro diseñado NO cumple con las especificaciones. COMO SE MUESTRA en los dos gráficos de detalle anteriores, con $N = 46$, no se cumplen ni la especificación de banda de paso en $wp = 0.4$ (frecuencia normalizada) ni la especificación de banda de supresión en $ws = 0.5$ (frecuencia normalizada). Entonces este diseño NO cumple con las especificaciones.

El orden de filtro que cumple con las especificaciones es - $N = 61$.

Para el filtro que CUMPLE CON LAS ESPECIFICACIONES, aquí están las gráficas:







22) Q7.22 Repeat Question Q7.20 using the function firpm.

A continuación se muestra el programa MATLAB para diseñar y trazar la ganancia y las respuestas de fase de un filtro FIR de fase lineal utilizando firpm. El orden de los filtros se estima usando kaiord. Los datos de salida son los coeficientes de filtro.

```

1. % Program Q7_22
2. % Use Parks-McClellan to design a linear phase Lowpass
3. % FIR Digital Filter meeting the design specification given
4. % in Q7.13.
5. % - Print out the numerator coefficients
6. % for the transfer function.
7. % - Compute and plot the gain function.

```

```

8. % - Compute and plot the phase response.
9. % - Compute and plot the unwrapped phase response.
10.    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
11.    clear;
12.    % Design spec as given in Q7.13.
13.    Fp = 2*10^3;
14.    Fs = 2.5*10^3;
15.    FT = 10*10^3;
16.    Rp = 0.005;
17.    Rs = 0.005;
18.    % Estimate the filter order and print to console
19.    N = kaiord(Fp,Fs,Rp,Rs,FT)
20.    % Design the filter using Parks-McClellan
21.    Wp = 2*Fp/FT; % These freqs are normalized: they go
22.    Ws = 2*Fs/FT; % zero to one, not zero to pi.
23.    F = [0 Wp Ws 1];
24.    A = [1 1 0 0];
25.    h = firpm(N,F,A);
26.    % Show the Numerator Coefficients
27.    disp('Numerator Coefficients are ');disp(h);
28.    % Compute and plot the gain response
29.    [g, w] = gain(h,[1]);
30.    figure(1);
31.    plot(w/pi,g);grid;
32.    %axis([0 1 -60 5]);
33.    xlabel('\omega /\pi'); ylabel('Gain in dB');
34.    title('Gain Response');
35.    % Compute the frequency response
36.    w2 = 0:pi/511:pi;
37.    Hz = freqz(h,[1],w2);
38.    % TEST: did we meet the spec?
39.    MagH = abs(Hz);
40.    T1 = 1.005*ones(1,length(w2));
41.    T2 = 0.995*ones(1,length(w2));
42.    T3 = 0.005*ones(1,length(w2));
43.    figure(4);
44.    plot(w2/pi,MagH,w2/pi,T1,w2/pi,T2,w2/pi,T3);grid;
45.    % Find and plot the phase
46.    figure(2);
47.    Phase = angle(Hz);
48.    plot(w2/pi,Phase);grid;
49.    xlabel('\omega /\pi'); ylabel('Phase (rad)');
50.    title('Phase Response');
51.    figure(3);
52.    UPhase = unwrap(Phase);

```

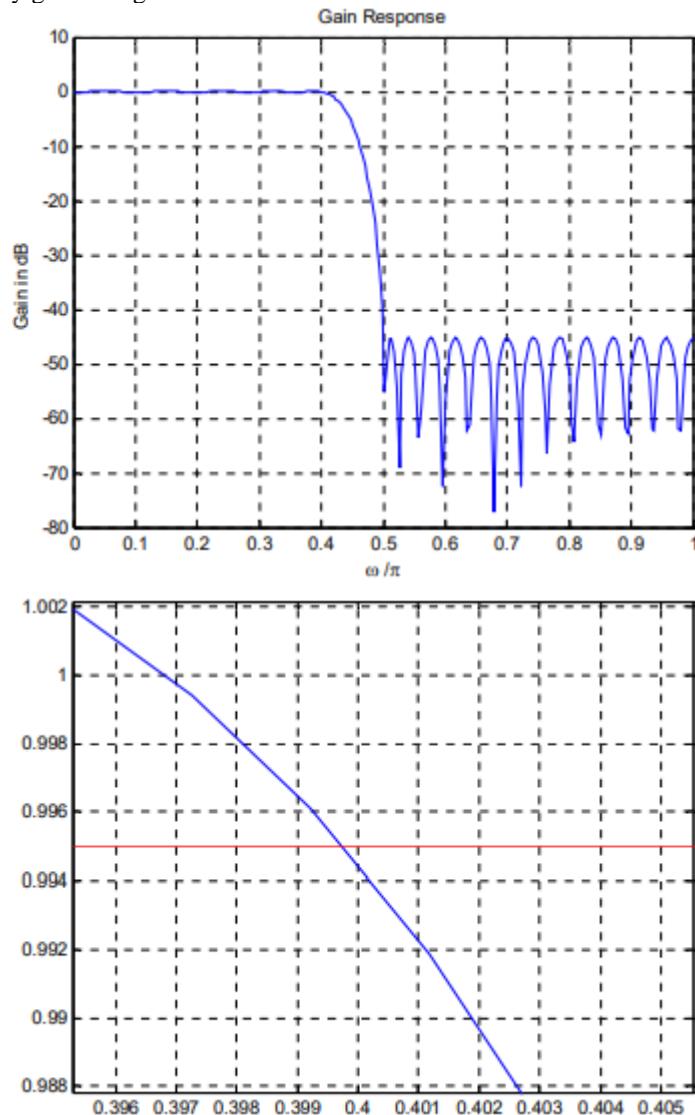
```

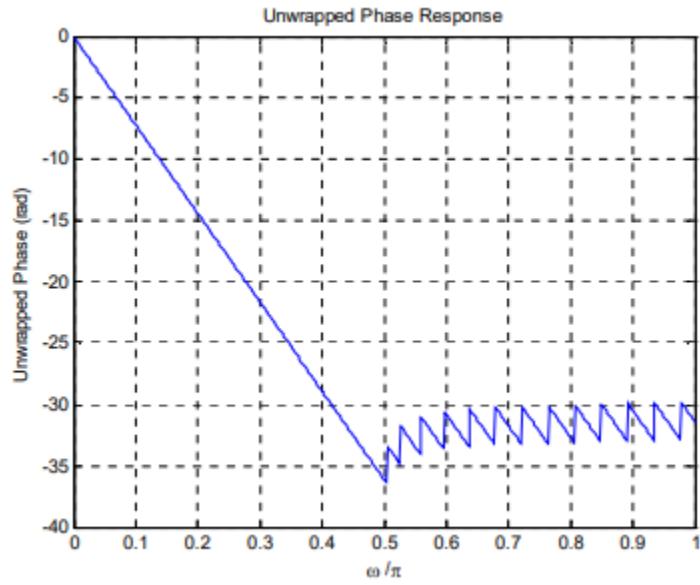
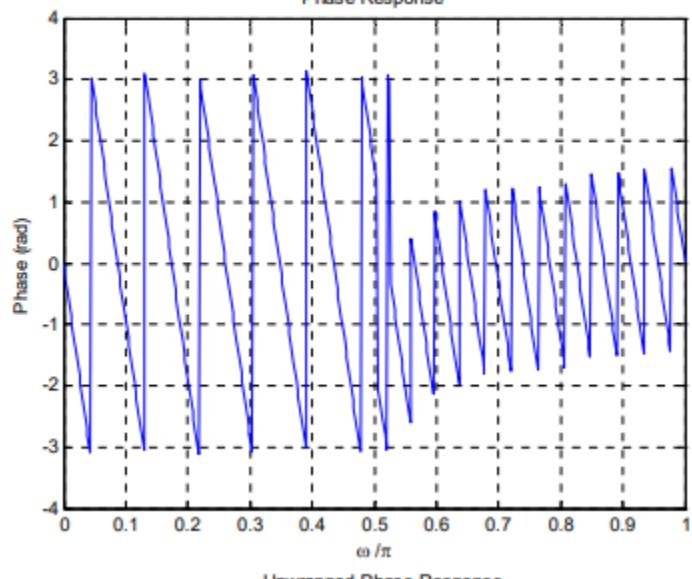
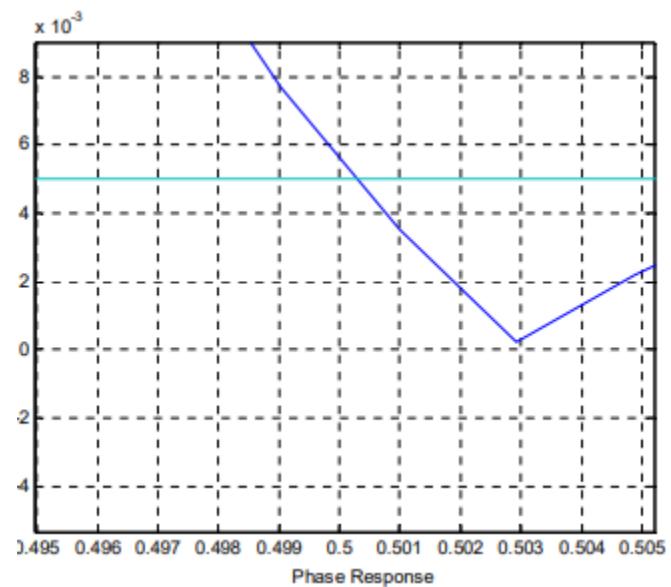
53. plot(w2/pi, UPhase); grid;
54. xlabel('omega /pi'); ylabel('Unwrapped Phase (rad)');
55. title('Unwrapped Phase Response');

```

Los coeficientes del filtro de paso bajo correspondientes a las especificaciones dadas en la Pregunta 7.20 son los que se muestran a continuación.

Las respuestas de fase y ganancia generadas se dan a continuación:

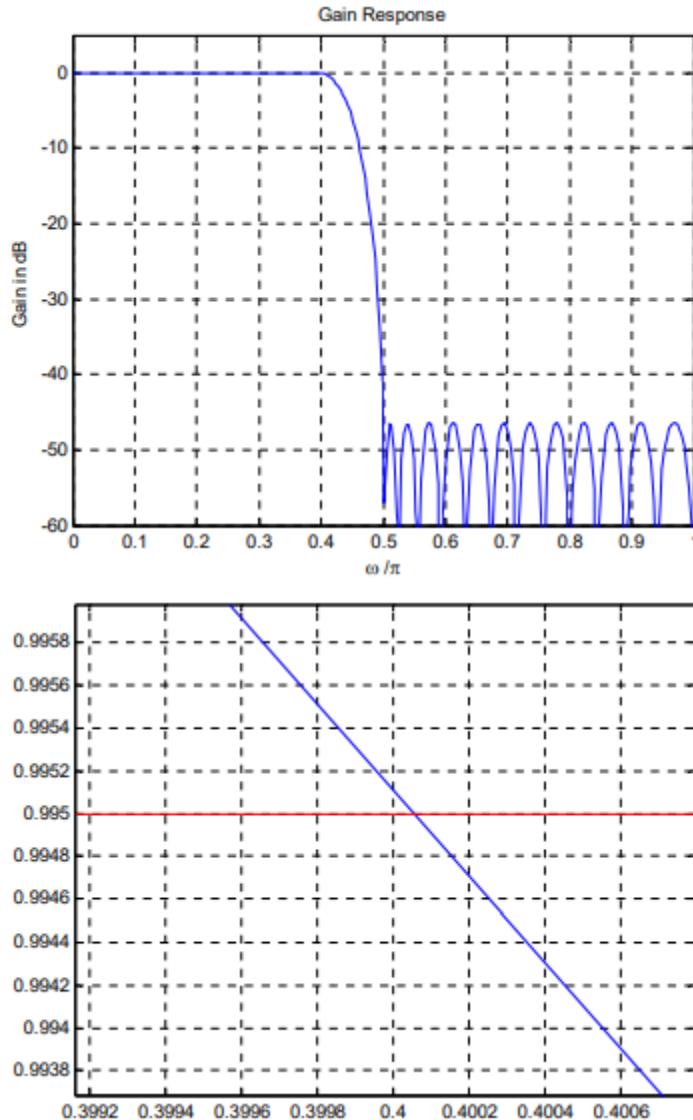


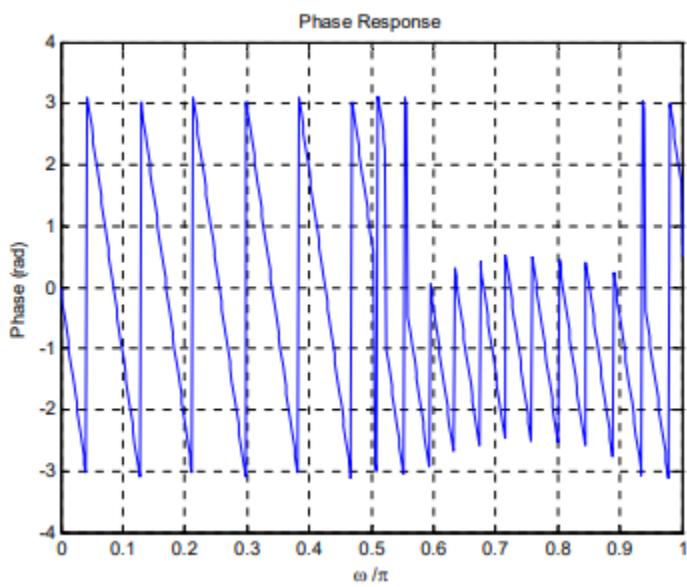
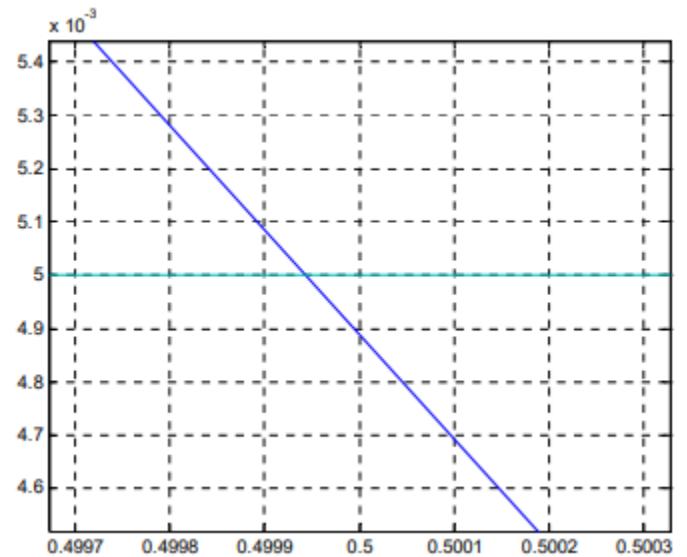


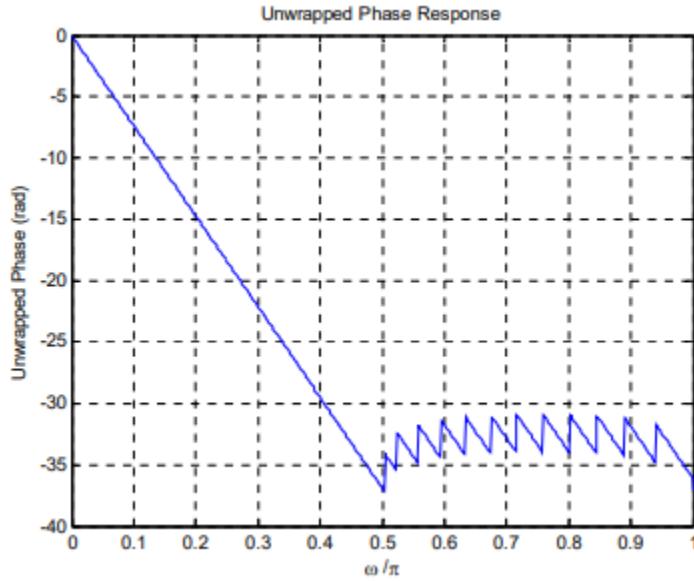
Del gráfico de ganancia observamos que el filtro diseñado **NO** cumple con las especificaciones.

El orden de filtro que cumple con las especificaciones es - $N = 47$.

Para el filtro con $N = 47$ que **CUMPLIÓ CON LAS ESPECIFICACIONES**, las gráficas se muestran a continuación:







- 23) Q7.23 Design an FIR lowpass filter using a Kaiser window. The filter specifications are: $\omega_p = 0.31$, $\omega_s = 0.41$, and $A_s = 50$ dB. Note that the function `kaiser` requires the values of the parameter β and the order N which must first be calculated using Eqs. (7.36) and (7.37), respectively. Does your design meet the specifications?

El programa MATLAB para diseñar y trazar la ganancia y las respuestas de fase de un filtro FIR de fase lineal usando `fir1` y `kaiser` se muestra a continuación. El orden de filtro N se estima usando la Ec. (7.37) y el parámetro β se calcula usando la Ec. (7.36). Los datos de salida son los coeficientes de filtro.

```

1. % Program Q7_23
2. % Use Kaiser window to design a linear phase Lowpass
3. % FIR Digital Filter meeting the design specification given
4. % in Q7.23.
5. %
6. % It is clear from the statement of the question that Mitra
7. % wants us to use (7.36) and (7.37) for this problem. That
8. % isn't the greatest thing to try because kaiserord already
9. % does
10. % exactly what we need.... but that's Q7_24! So here goes!
11. % - Print out the numerator coefficients
12. % for the transfer function.
13. % - Compute and plot the gain function.
14. %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
15. clear;
16. % Design spec as given in Q7.23.
17. Wp = 0.31;
18. Ws = 0.41;
19. Wn = Wp + (Ws-Wp)/2;
20. As = 50;
21. Ds = 10^(-As/20);

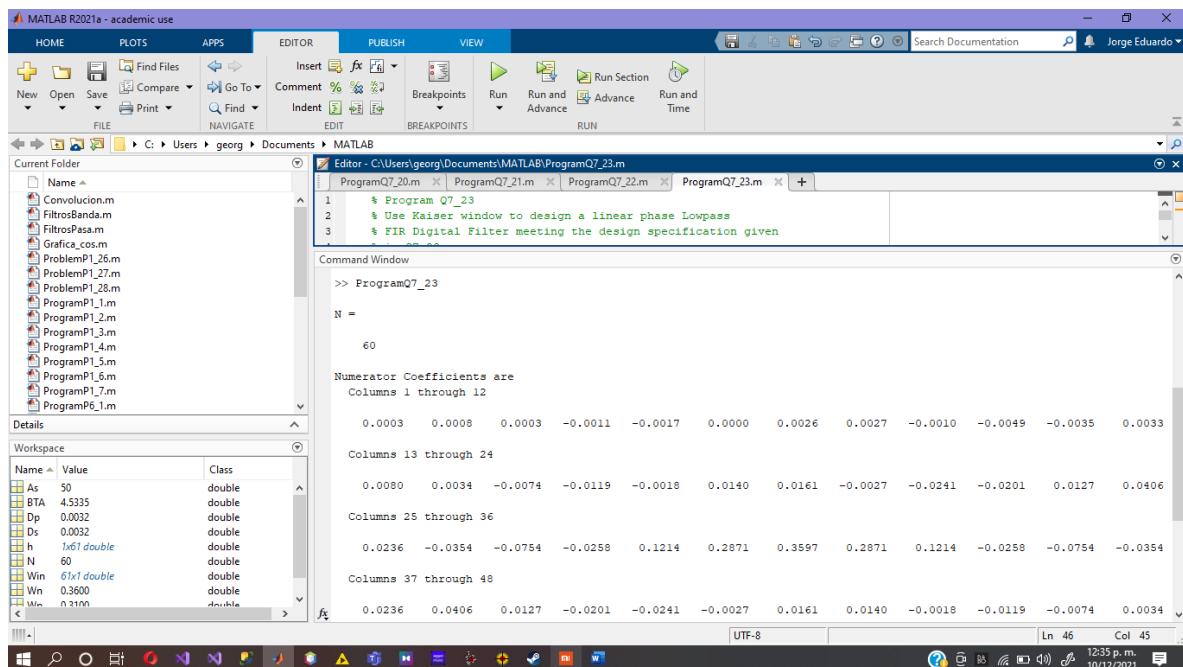
```

```

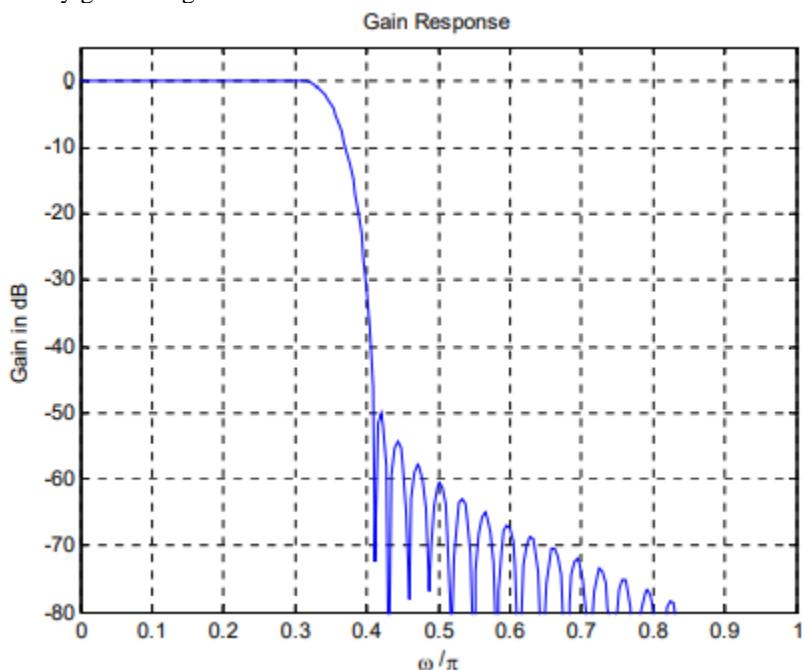
21. Dp = Ds; %Kaiser window design has equal ripple in
22. % passband and stopband.
23. % estimate order using (7.37)
24. if As > 21
25. N = ceil((As-7.95)*2/(14.36*abs(Wp-Ws))+1)
26. else
27. N = ceil(0.9222*2/abs(Wp-Ws)+1)
28. end
29. % Use (7.36) to get Beta
30. if As > 50
31. BTA = 0.1102*(As-8.7);
32. elseif As >= 21
33. BTA = 0.5842*(As-21)^0.4+0.07886*(As-21);
34. else
35. BTA = 0;
36. end
37. Win = kaiser(N+1,BTA);
38. h = fir1(N,Wn,Win);
39. % Show the Numerator Coefficients
40. disp('Numerator Coefficients are ');disp(h);
41. % Compute and plot the gain response
42. [g, w] = gain(h,[1]);
43. figure(1);
44. plot(w/pi,g);grid;
45. axis([0 1 -80 5]);
46. xlabel('\omega /\pi'); ylabel('Gain in dB');
47. title('Gain Response');
48. % Compute the frequency response
49. w2 = 0:pi/511:pi;
50. Hz = freqz(h,[1],w2);
51. % Find and plot the phase
52. figure(2);
53. Phase = angle(Hz);
54. plot(w2/pi,Phase);grid;
55. xlabel('\omega /\pi'); ylabel('Phase (rad)');
56. title('Phase Response');
57. figure(3);
58. UPhase = unwrap(Phase);
59. plot(w2/pi,UPhase);grid;
60. xlabel('\omega /\pi'); ylabel('Unwrapped Phase (rad)');
61. title('Unwrapped Phase Response');

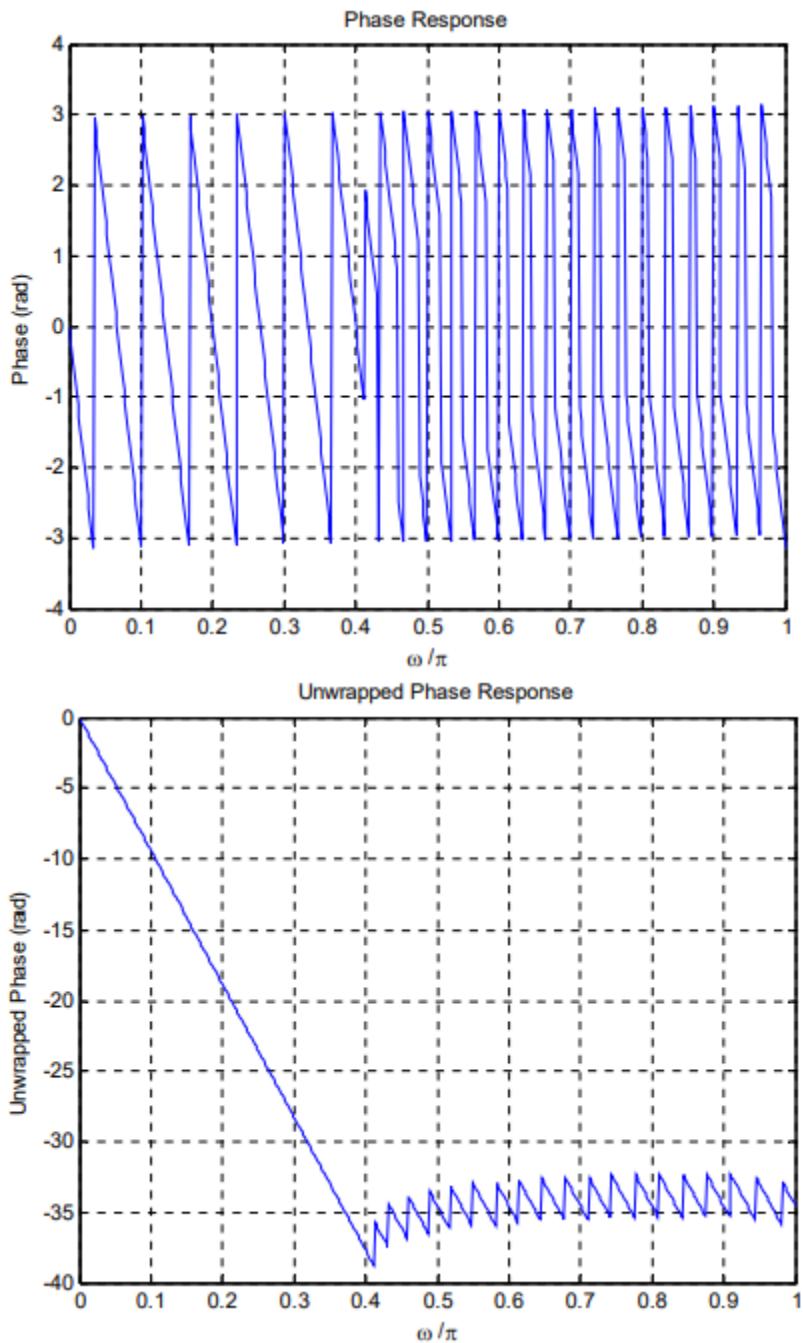
```

Los coeficientes del filtro de paso bajo correspondientes a las especificaciones dadas en la Pregunta 7.23 son los que se muestran a continuación:



Las respuestas de fase y ganancia generadas se dan a continuación:





De la gráfica de ganancia observamos que el filtro diseñado cumple con las especificaciones. El orden de filtrado que cumple con las especificaciones es - $N = 60$.

24) Q7.24 Repeat Question Q7.23 using the functions kaiserord and fir1.

El programa MATLAB para diseñar y trazar la ganancia y las respuestas de fase de un filtro FIR de fase lineal usando fir1 y kaiser se muestra a continuación. El orden de filtrado N y el parámetro β se evalúan utilizando kaiserord. Los datos de salida son los coeficientes de filtro.

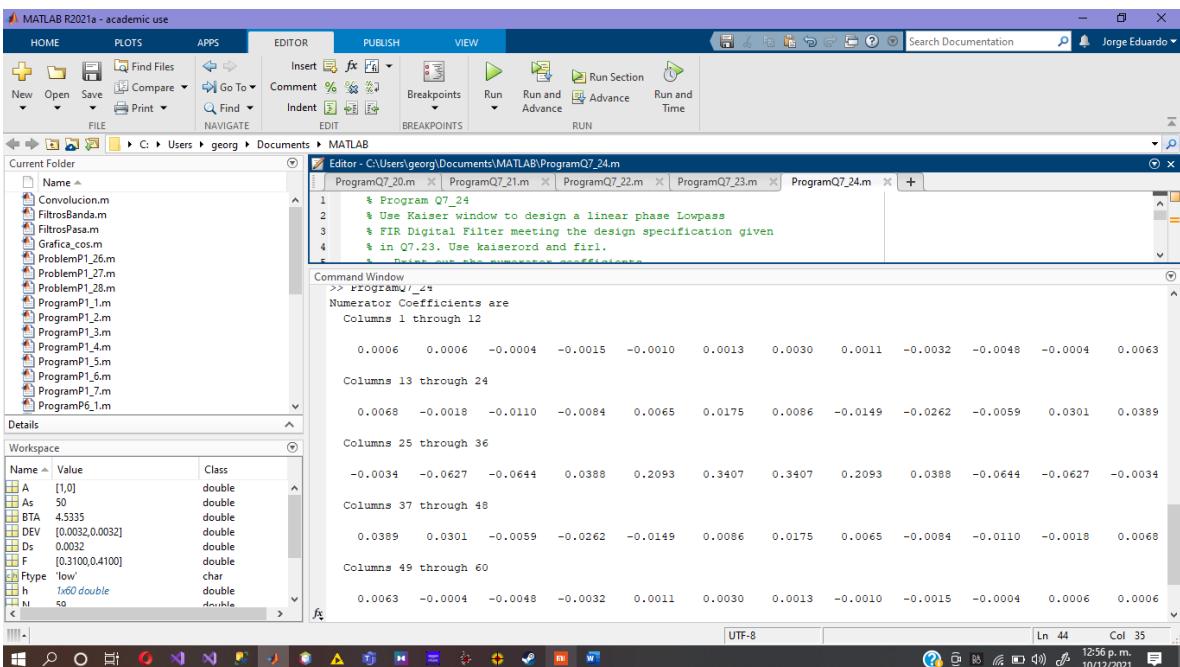
```
1. % Program Q7_24
2. % Use Kaiser window to design a linear phase Lowpass
```

```

3. % FIR Digital Filter meeting the design specification given
4. % in Q7.23. Use kaiserord and fir1.
5. % - Print out the numerator coefficients
6. % for the transfer function.
7. % - Compute and plot the gain function.
8. %%%%%%%%%%%%%%
9. clear;
10. % Design spec as given in Q7.23.
11. Wp = 0.31;
12. Ws = 0.41;
13. As = 50;
14. Ds = 10^(-As/20);
15. % Design the Filter
16. F = [Wp Ws];
17. A = [1 0];
18. DEV = [Ds Ds];
19. [N,Wn,BTA,Ftype] = kaiserord(F,A,DEV);
20. Win = kaiser(N+1,BTA);
21. h = fir1(N,Wn,Ftype,Win);
22. % Show the Numerator Coefficients
23. disp('Numerator Coefficients are ');disp(h);
24. % Compute and plot the gain response
25. [g, w] = gain(h,[1]);
26. figure(1);
27. plot(w/pi,g);grid;
28. axis([0 1 -80 5]);
29. xlabel('\omega /\pi'); ylabel('Gain in dB');
30. title('Gain Response');
31. % Compute the frequency response
32. w2 = 0:pi/511:pi;
33. Hz = freqz(h,[1],w2);
34. % Find and plot the phase
35. figure(2);
36. Phase = angle(Hz);
37. plot(w2/pi,Phase);grid;
38. xlabel('\omega /\pi'); ylabel('Phase (rad)');
39. title('Phase Response');
40. figure(3);
41. UPhase = unwrap(Phase);
42. plot(w2/pi,UPhase);grid;
43. xlabel('\omega /\pi'); ylabel('Unwrapped Phase (rad)');
44. title('Unwrapped Phase Response');

```

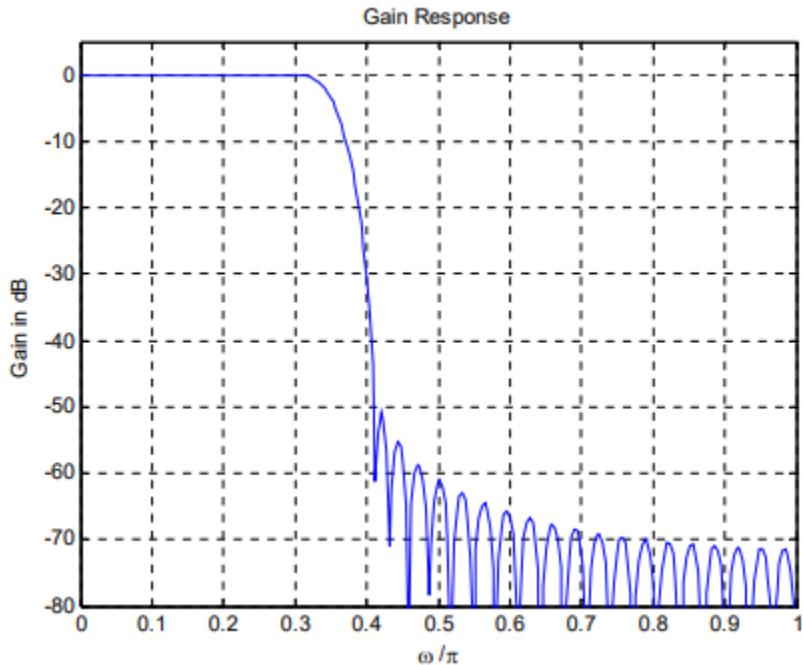
Los coeficientes del filtro de paso bajo correspondientes a las especificaciones dadas en la Pregunta 7.24 son los que se muestran a continuación:

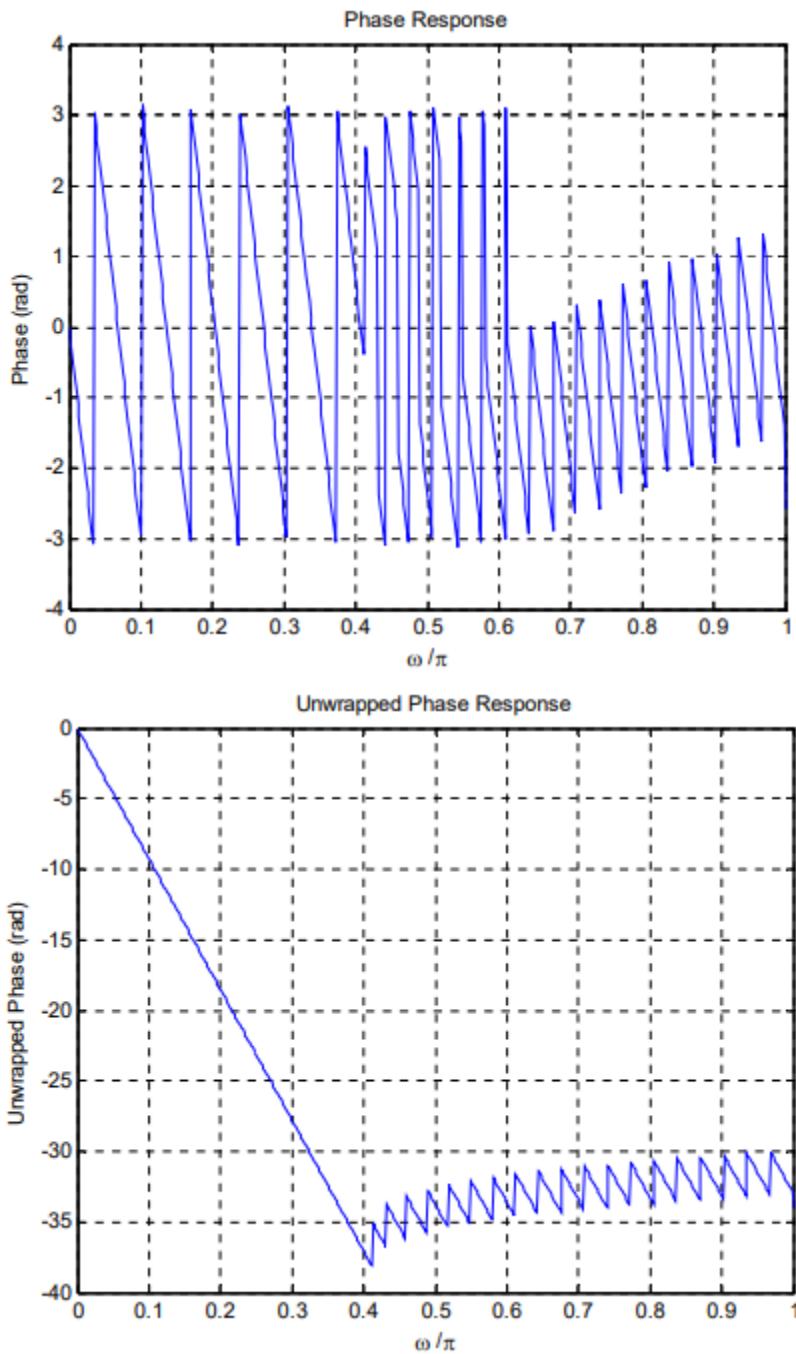


Los coeficientes del filtro de paso bajo correspondientes a las especificaciones dadas en la Pregunta 7.23 son las que se muestran a continuación:

$$Wp = 0.31; Ws = 0.41; As = 50 \text{ dB}.$$

Las respuestas de fase y ganancia generadas se dan a continuación:





De la gráfica de ganancia observamos que **el filtro diseñado cumple con las especificaciones**.
 El orden de filtro que cumple con las especificaciones es $-N = 59$.

- 25) Q7.25 Using `fir2` design an FIR filter of order 95 with three different constant magnitude levels: 0.4 in the frequency range 0 to 0.25, 1.0 in the frequency range 0.3 to 0.45, and 0.8 in the frequency range 0.5 to 1.0. Plot the magnitude response of the filter designed. Does your design meet the specifications?

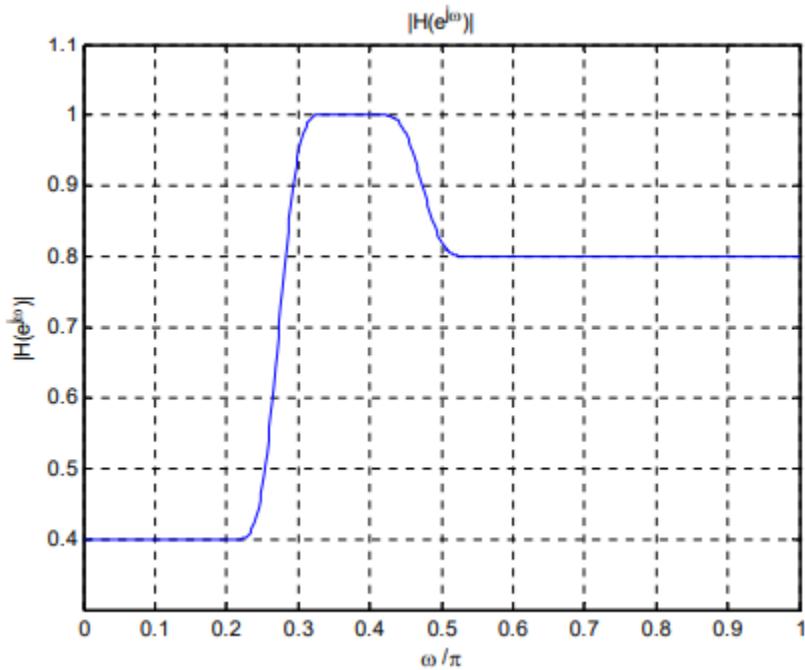
El programa MATLAB para diseñar y trazar la respuesta de magnitud de un filtro FIR multibanda de fase lineal utilizando `fir2` se muestra a continuación:

```

1. % Program Q7_25
2. % Use fir2 to design a linear phase Lowpass
3. % FIR Digital Filter meeting the design specification given
4. % in Q7.23.
5. % - Compute and plot the gain function.
6. %%%%%%
7. clear;
8. % Design spec as given in Q7.25.
9. N = 95;
10. A = [0.4 0.4 1.0 1.0 0.8 0.8];
11. F = [0 0.25 0.3 0.45 0.5 1.0];
12. % Design the Filter
13. h = fir2(N,F,A);
14. % Compute and plot the gain response
15. [g, w] = gain(h,[1]);
16. figure(1);
17. plot(w/pi,g);grid;
18. %axis([0 1 -80 5]);
19. xlabel('omega /pi'); ylabel('Gain in dB');
20. title('Gain Response');
21. % Compute the frequency response
22. w2 = 0:pi/511:pi;
23. Hz = freqz(h,[1],w2);
24. % Plot
25. figure(2);
26. plot(w2/pi,abs(Hz));grid;
27. xlabel('omega /pi'); ylabel('|H(e^{j\omega})|');
28. title('|H(e^{j\omega})|');
29. % Find and plot the phase
30. figure(3);
31. Phase = angle(Hz);
32. plot(w2/pi,Phase);grid;
33. xlabel('omega /pi'); ylabel('Phase (rad)');
34. title('Phase Response');
35. figure(4);
36. UPhase = unwrap(Phase);
37. plot(w2/pi,UPhase);grid;
38. xlabel('omega /pi'); ylabel('Unwrapped Phase (rad)');
39. title('Unwrapped Phase Response');

```

La respuesta de magnitud del filtro diseñado para las especificaciones dadas en la Pregunta Q7.25 se muestra a continuación:



De la gráfica de respuesta de magnitud observamos que el filtro diseñado **NO** cumple con las especificaciones

- 26) Q7.26 Using firpm, design the FIR bandpass filter with specifications given in Question Q7.17 and of order obtained using kaiserord. The vector $wgts$ needed in the function firpm is given by

$$wgts = \max(\delta_p, \delta_s) \left[\frac{1}{\delta_p}, \frac{1}{\delta_s} \right]$$

Does your design meet the specifications? If it does not, adjust the filter order until the design meets the specifications. What is the order of the filter meeting the specifications?

El programa MATLAB para diseñar y trazar la respuesta de ganancia de un filtro FIR de paso de banda de fase lineal usando firpm y kaiserord se muestra a continuación:

Nuevamente hay ambigüedad dependiendo del clima \ delta_p se supone que es 0.1 o 0.01. Si lo haces 0.01 obtienes esto:

```

1. % Program Q7_26
2. % Use kaiserord and firpm to design the linear phase bandpass
3. % FIR Digital Filter specified in Q7.17.
4. % - Print out the numerator coefficients
5. % for the transfer function.
6. % - Compute and plot the gain function.
7. %%%%%%
8. clear;
9. % Design spec as given in Q7.17.
10. F = [1200 1800 3600 4200];
11. A = [0 1 0];
12. DEV = [0.02 0.1 0.02];
13. Fs = 12000;
14. Dp = 0.1;
15. Ds = 0.02;

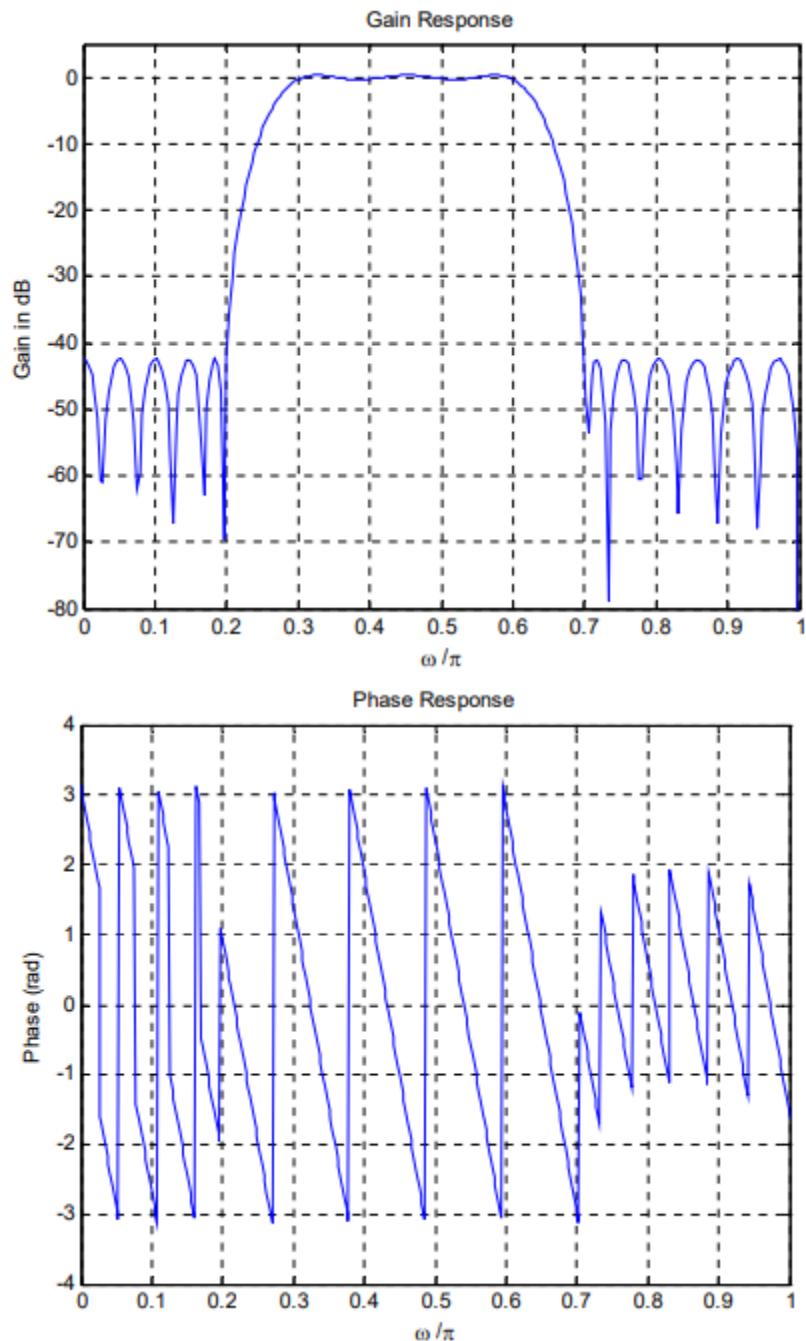
```

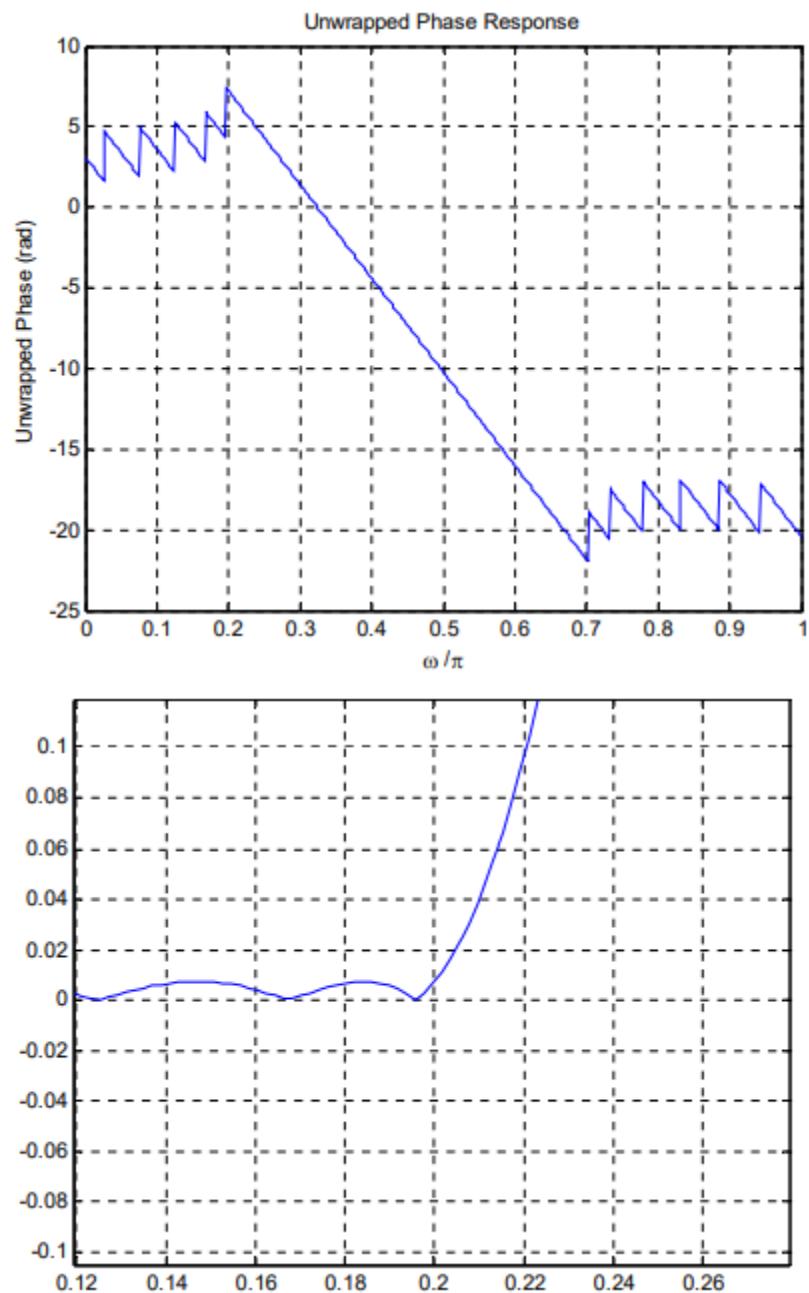
```

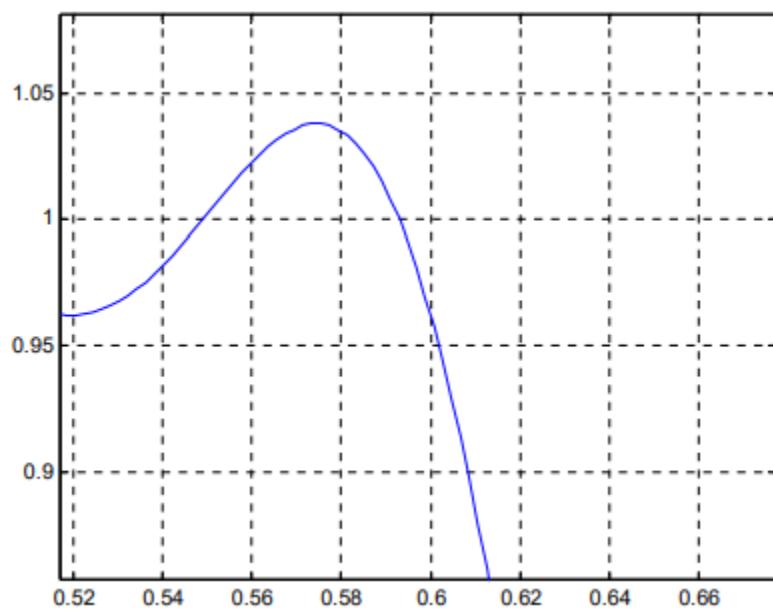
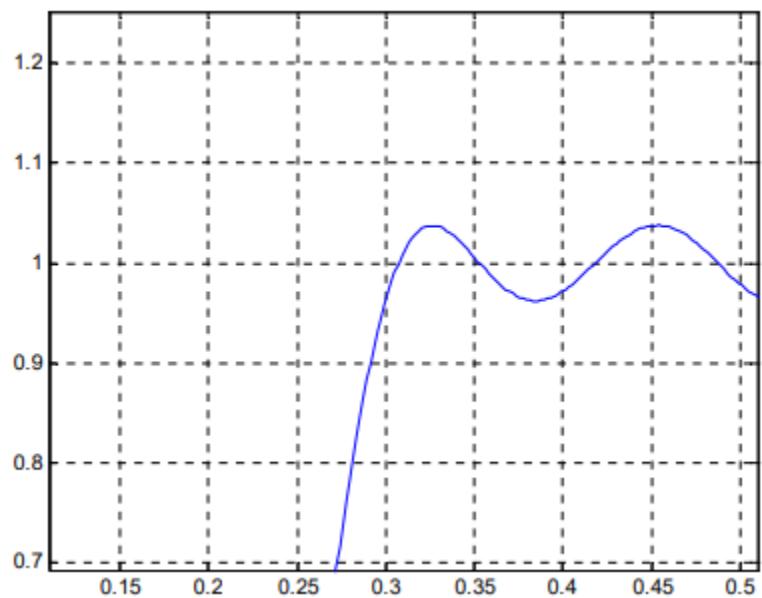
16.      [N,Wn,BTA,FILTYPE] = kaiserord(F,A,DEV,Fs);
17.      N
18.      % firpm setup
19.      F2 = 2*[0 1200 1800 3600 4200 6000]/Fs;
20.      A2 = [0 0 1 1 0 0];
21.      wgts = max(Dp,Ds)*[1/Ds 1/Dp 1/Ds];
22.      h = firpm(N,F2,A2,wgts);
23.      % Show the Numerator Coefficients
24.      disp('Numerator Coefficients are ');disp(h);
25.      % Compute and plot the gain response
26.      [g, w] = gain(h,[1]);
27.      figure(1);
28.      plot(w/pi,g);grid;
29.      axis([0 1 -80 5]);
30.      xlabel('omega /pi'); ylabel('Gain in dB');
31.      title('Gain Response');
32.      % Compute the frequency response
33.      w2 = 0:pi/511:pi;
34.      Hz = freqz(h,[1],w2);
35.      % Find and plot the phase
36.      figure(2);
37.      Phase = angle(Hz);
38.      plot(w2/pi,Phase);grid;
39.      xlabel('omega /pi'); ylabel('Phase (rad)');
40.      title('Phase Response');
41.      figure(3);
42.      UPhase = unwrap(Phase);
43.      plot(w2/pi,UPhase);grid;
44.      xlabel('omega /pi'); ylabel('Unwrapped Phase (rad)');
45.      title('Unwrapped Phase Response');

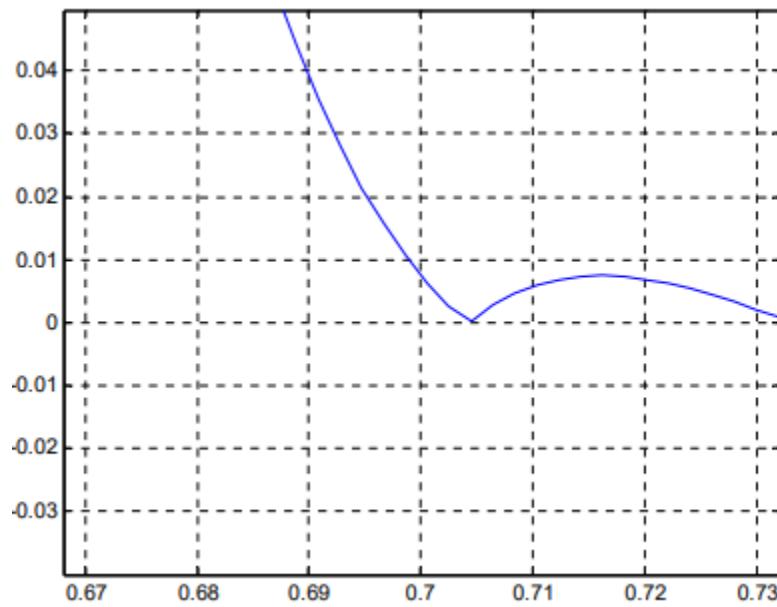
```

La respuesta de ganancia del filtro diseñado para las especificaciones dadas en la Pregunta Q7.17 se muestra a continuación:



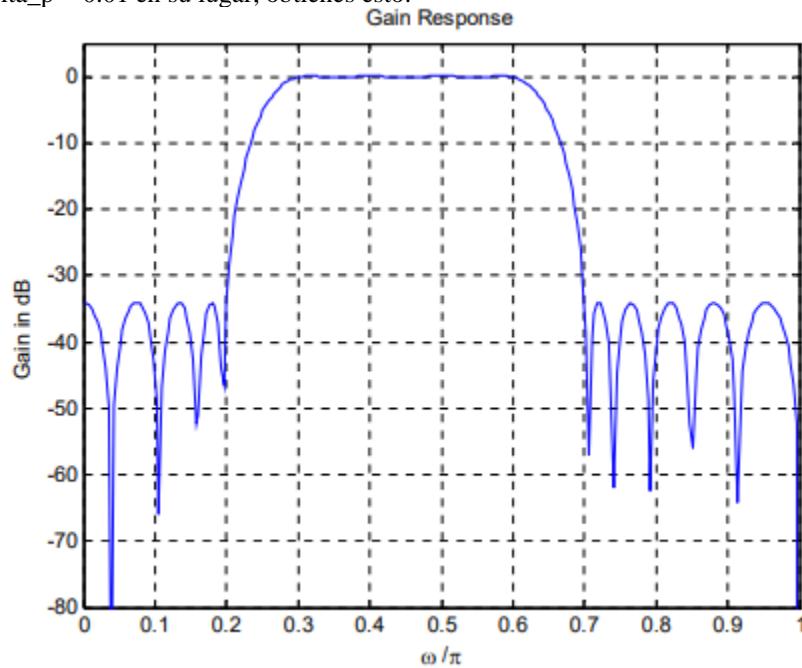


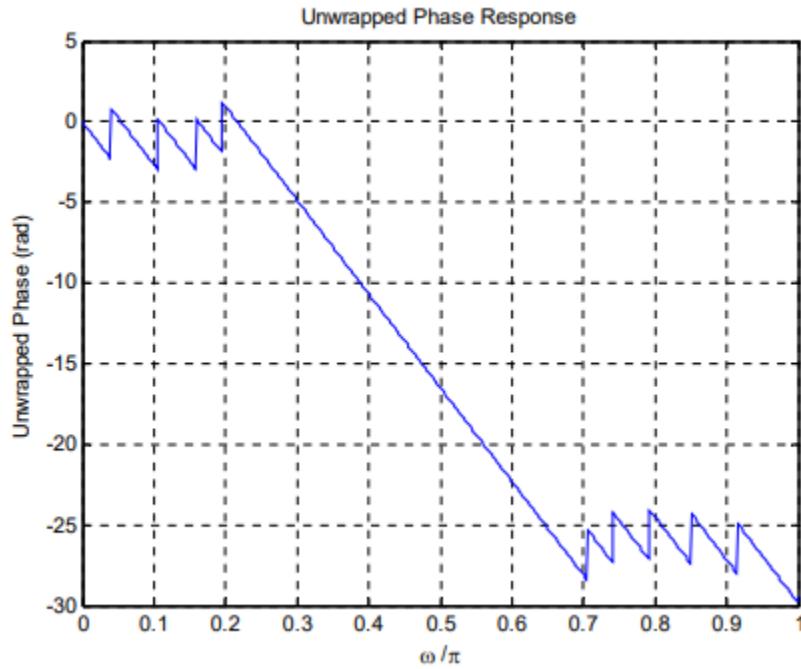




De la gráfica de respuesta de ganancia observamos que el filtro diseñado cumple con las especificaciones.

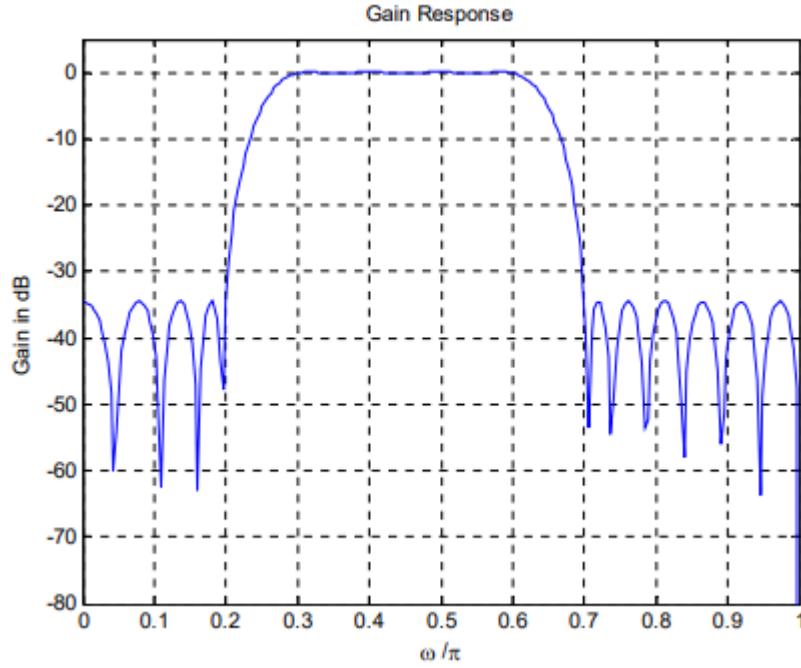
El orden de filtro que cumple con las especificaciones es - $N = 37$,
TOMANDO $\delta_p = 0.01$ en su lugar, obtienes esto:

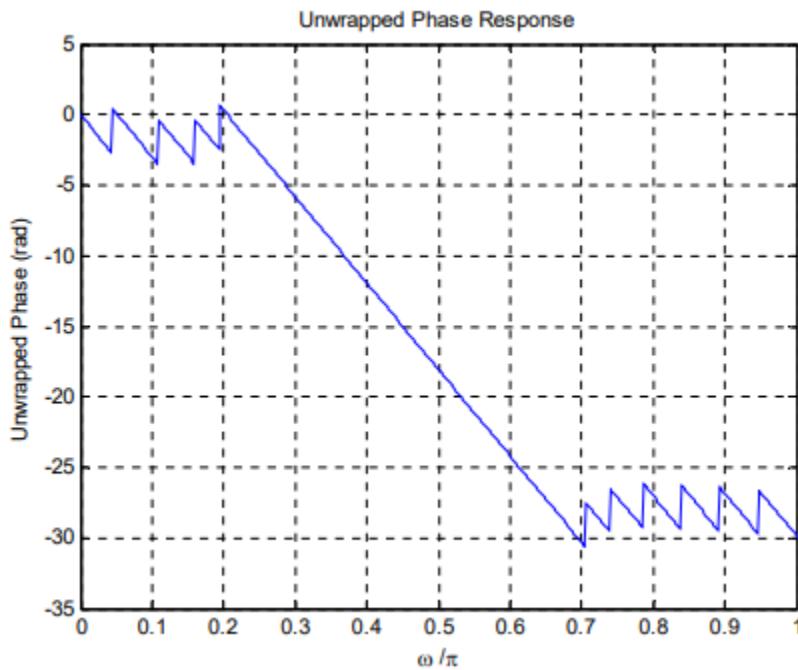




Si examina este de cerca, no cumple con la especificación en las frecuencias de borde de banda de paso. El orden es $N = 37$.

Aumentando ligeramente el pedido a $N = 39$, obtenemos el siguiente filtro que cumple con la especificación:





- 27) Q7.27 Using firpm, design an FIR bandpass filter with the following specifications: passband edges = 1.8 and 3.0 kHz, stopband edges = 1.5 and 4.2 kHz, passband ripple $\delta_p = 0.1$, stopband ripple $\delta_s = 0.02$, and sampling rate of 12 kHz. Estimate the filter order using kaiserord. Is your design an optimal FIR filter? Does your design meet the specifications? If it does not, does increasing the filter order help in meeting the specifications? Are the specifications met by a filter with a lower order than that obtained using kaiserord? In the case of unequal transition bands, the filter designed using firpm may exhibit unsatisfactory behavior in the larger transition bandwidth in its gain response. One way to improve the behavior is to reduce the transition band by moving the stopband edge until the design meets the specifications with smooth roll-off in the transition bands. Try this approach and determine the new specifications, with passband edges remaining fixed, that provide smooth roll-off in the transition bands.

Usando el programa MATLAB desarrollado en la Pregunta Q7.26, se diseña el filtro de paso de banda FIR de fase lineal para las especificaciones de la Pregunta Q7.27.

```

1. % Program Q7_27
2. % Use kaiserord and firpm to design the linear phase bandpass
3. % FIR Digital Filter specified in Q7.27.
4. %
5. %%%%%%%%%%%%%%
6. clear;
7. % Design spec as given in Q7.27.
8. Fs1 = 1500;
9. Fp1 = 1800;
10. Fp2 = 3000;
11. Fs2 = 4200;
12. Fs = 12000;
13. Dp = 0.1;
14. Ds = 0.02;

```

```

15.      F = [Fs1 Fp1 Fp2 Fs2];
16.      A = [0 1 0];
17.      DEV = [Ds Dp Ds];
18.      [N,Wn,BTA,FILTYPE] = kaiserord(F,A,DEV,Fs);
19.      % firpm setup
20.      ws1 = 2*Fs1/Fs;
21.      wp1 = 2*Fp1/Fs;
22.      wp2 = 2*Fp2/Fs;
23.      ws2 = 2*Fs2/Fs;
24.      F2 = [0 ws1 wp1 wp2 ws2 1];
25.      A2 = [0 0 1 1 0 0];
26.      wgt = max(Dp,Ds)*[1/Ds 1/Dp 1/Ds];
27.      h = firpm(N,F2,A2,wgt);
28.      % Show the Numerator Coefficients
29.      disp('Numerator Coefficients are ');disp(h);
30.      % Compute and plot the gain response
31.      [g, w] = gain(h,[1]);
32.      figure(1);
33.      plot(w/pi,g);grid;
34.      axis([0 1 -80 5]);
35.      xlabel('omega /pi'); ylabel('Gain in dB');
36.      title('Gain Response');
37.      % Compute the frequency response
38.      w2 = 0:pi/511:pi;
39.      Hz = freqz(h,[1],w2);
40.      % Find and plot the phase
41.      figure(2);
42.      Phase = angle(Hz);
43.      plot(w2/pi,Phase);grid;
44.      xlabel('omega /pi'); ylabel('Phase (rad)');
45.      title('Phase Response');
46.      figure(3);
47.      UPhase = unwrap(Phase);
48.      plot(w2/pi,UPhase);grid;
49.      xlabel('omega /pi'); ylabel('Unwrapped Phase (rad)');
50.      title('Unwrapped Phase Response');
51.      figure(4);
52.      % Add lines to the plot to help determine if the spec
         was met.
53.      hold on;
54.      tmpY = 0:1.4/4:1.4;
55.      tmpX = ones(1,length(tmpY))*wp1;
56.      plot(tmpX,tmpY,'r-'); % vertical line at passband edge
      freq
57.      tmpX = ones(1,length(tmpY))*wp2;

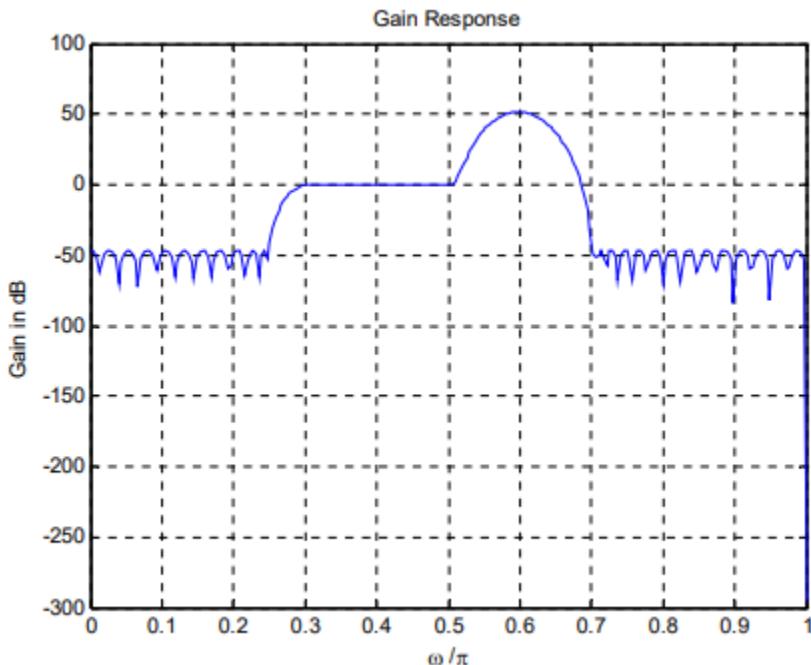
```

```

58. plot(tmpX,tmpY,'r-'); % vertical line at passband edge
   freq
59. tmpX = ones(1,length(tmpY))*ws1;
60. plot(tmpX,tmpY,'g-'); % vertical line at stopband edge
   freq
61. tmpX = ones(1,length(tmpY))*ws2;
62. plot(tmpX,tmpY,'g-'); % vertical line at stopband edge
   freq
63. tmpY = ones(1,length(w))*(Dp);
64. plot(w/pi,tmpY,'r-'); % horizontal line at Dp
65. tmpY = ones(1,length(w))*(Ds);
66. plot(w/pi,tmpY,'g-'); % horizontal line at Ds
67. % now plot the Frequency Response
68. plot(w2/pi,abs(Hz));grid;
69. hold off;

```

La respuesta de ganancia del filtro se muestra a continuación:



De la gráfica de respuesta de ganancia observamos que el filtro diseñado NO cumple con las especificaciones: **LA OBLIGACIÓN DE LA BANDA SE EXCEDE EN LA BANDA DE TRANSICIÓN SUPERIOR**. El filtro es óptimo en el sentido minimax, al igual que todos los diseños de ParksMcClellan. La razón por la que esto ocurrió es porque:

- Las bandas de transición no son regiones importantes para firpm.
- A menudo se observa un comportamiento errático de este tipo en las bandas de transición si los anchos de banda de transición no son iguales.

La solución habitual es hacer que todas las bandas de transición tengan el ancho de la banda de transición más pequeña; p.ej. para sobre diseñar las otras bandas de transición para hacer que las primeras "suenen bien".

El orden de filtro que cumple con las especificaciones es - N = 73 (según lo entregado por kaiserord).

Las nuevas especificaciones para un deslizamiento suave en las bandas de transición son:

$$Fs1 = 1500; Fp1 = 1800; Fp2 = 3000; Fs2 = 3300; Fs = 12000;$$

