



Chapter 7: Thiết kế bộ lọc IIR, FIR

| | |
|----------|----------------|
| 📅 Date | @April 1, 2024 |
| 🌟 Status | Done |

▼ Lý thuyết

▼ Bộ lọc số

$$H(z) = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + \dots + b_M z^{-M}}{1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_N z^{-N}} = \frac{\sum_{k=0}^M (b_k z^{-k})}{\sum_{k=0}^N (a_k z^{-k})},$$

- $a_0 = 1$. Các vector $\{a_k\}, \{b_k\}$ là các hệ số đặc trưng cho hàm truyền. Trong trường hợp, các hệ số $a_k = 0, \forall k > 0, H(z) = b_0 + b_1 z^{-1} + \dots + b_M z^{-M}$ nên đáp ứng xung của hệ thống $h(n) = \{b_0, b_1, b_2, \dots, b_M\}$ có chiều dài hữu hạn. Hệ thống được gọi là Finite Impulse Response hay FIR \Rightarrow Hệ thống FIR luôn ổn định.
- Ngược lại nếu có $a_k \neq 0 (k > 0)$ thì đáp ứng xung của hệ sẽ dài vô hạn. Hệ thống gọi là Infinite Impulse Response hay IIR.
- Thông thường khi xem xét hệ thống về mặt tần số, ta sử dụng đáp ứng tần số (đáp ứng biên độ và đáp ứng pha). Lệnh `freqz` của Matlab sẽ tính (hoặc vẽ) đáp ứng tần số tương ứng của hệ thống.

▼ Ví dụ

$$H(z) = \frac{(1 + z^{-1})}{(1 - 1.4461z^{-1} + 0.7957z^{-2})}$$

- `freqz([1, 1], [1, -1.4461, 0.7957])` ta vẽ được đáp ứng biên độ và đáp ứng pha của bộ lọc
- `[H, W] = freqz(b, a, n)`: sẽ tiến hành tính đáp ứng tần số H , tại các giá trị tần số W đã được chuẩn hoá về $[0,1]$.

▼ Lọc IIR

Các bộ lọc số IIR thường được thiết kế dựa trên các bộ lọc tương tự. Một số loại bộ lọc IIR phổ biến, Tuỳ vào loại bộ lọc, ta có các lệnh Matlab tương ứng: `butter`, `cheby1`, `cheby2`, `ellip`.

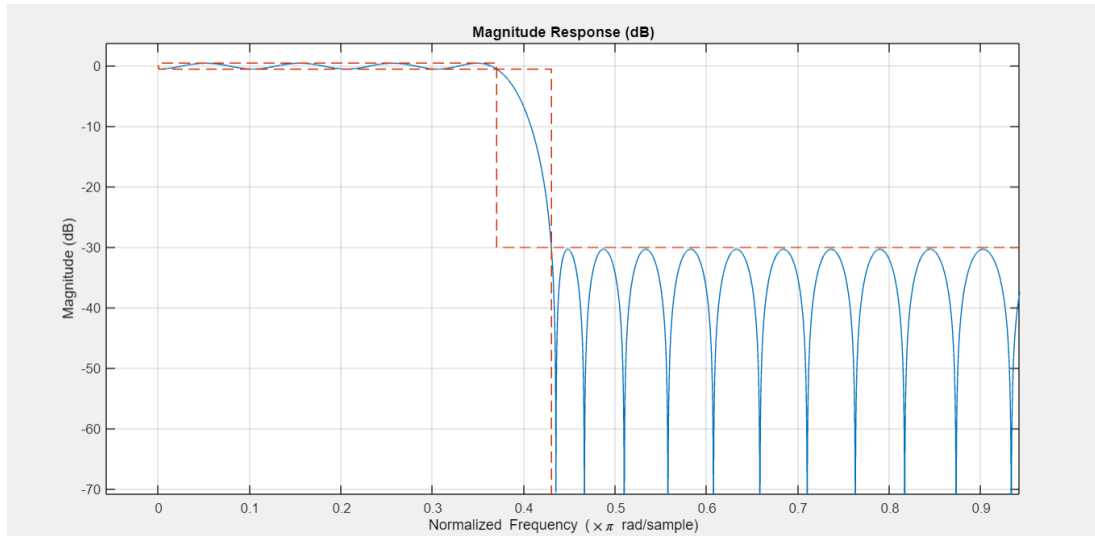
```
clf;
clear all; close all;
N = 512;
n = 6; %bac bo loc
fc = input("Nhap tan so cat:...");
fs = input("Nhap tan so lay mau:...");
Wc = 2 * fc / fs; %Tan so chuan hoa
[b,a]=butter(n,Wc,"low"); %Tinh he so cua ham truyen bo
freqz(b,a,N,fs); %Ve dap ung tan so
title("Dap ung bien do va pha cua bo loc thong thap But
```

▼ fdatool và fvtool

Đây là các công cụ nằm trong thư viện Signal processing toolbox của Matlab.

`fvtool` thực hiện dưới dạng dòng lệnh

`fdatool` cho phép sử dụng giao diện người dùng trực quan, GUI.



▼ Thiết kế bộ lọc FIR

Bộ lọc thông cao, thông dải, chặn dải có thể thu được từ bộ lọc thông thấp, thông qua các phép chuyển đổi khác.

Trong đó:

- Phổ biên độ của bộ lọc $|H(\omega)|$ hoặc theo đơn vị dB: $|H(\omega)| = -20\log_{10} \frac{H(\omega)}{\max(H(\omega))}$
- ω là tần số được chuẩn hoá, được xác định theo công thức: $\omega = 2F/F_s$.
- ω_p, ω_s lần lượt là tần số dải thông, tần số dải triệt (chuẩn hoá).
- δ_1, δ_2 lần lượt là độ gợn sóng cực đại ở dải thông và dải triệt.
- R_p hoặc A_p lần lượt là độ gợn sóng dải thông: $R_p = -20\log \frac{1-\delta_1}{1+\delta_1}$
- A_s độ suy giảm ở dải triệt, được xác định theo công thức: $A_s = -20\log \frac{\delta_2}{1+\delta_1}$
- Tần số cắt ω_c thường được chọn là trung bình của ω_p và ω_s .

▼ 7.4.1 Thiết kế sử dụng phương pháp cửa sổ

Người ta dùng phương pháp cửa sổ để thiết kế bộ lọc thông thấp, bằng cách giới hạn chiều dài đáp ứng xung của một bộ lọc thông thấp lý tưởng:

$$h_{ideal}^{LP} = \sin(\omega_c n) / (\pi n).$$

Gọi $w(n)$ là hàm cửa sổ có chiều dài L , thì đáp ứng xung của bộ lọc sẽ được xác định:

$$h^{LP}(n) = h_{ideal}^{LP} \cdot w(n).$$

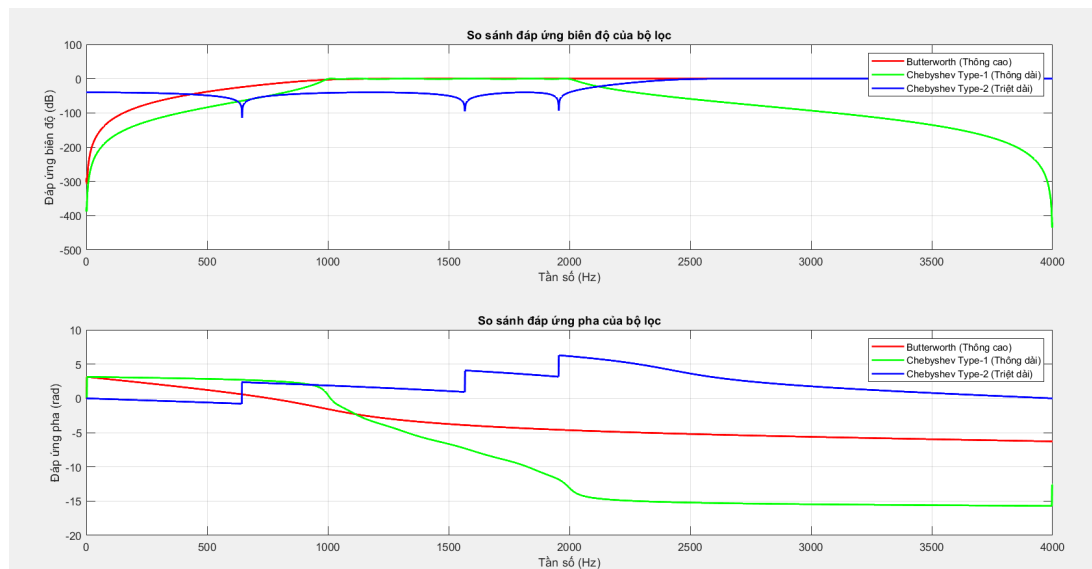
▼ 7.4.2 Chuyển đổi giữa các bộ lọc

Để thiết kế các bộ lọc thông cao, thông dải, chặn dải từ các bộ lọc thông thấp ta có thể dùng các công thức chuyển đổi sau:

- $h^{HP}(n) = \delta(n) - h^{LP}(n)$ trong đó $h^{LP}(n)$ là đáp ứng xung của bộ lọc thông thấp tần số cắt w_c
- $h^{HP}(n) = (-1)^n h^{LP}(n)$ trong đó $h^{LP}(n)$ là đáp ứng xung của bộ lọc thông thấp tần số cắt $(\pi - w_c)$
- $h^{BP}(n) = \cos((w_1 + w_2).n/2) h^{LP}(n)$ trong đó w_1, w_2 là dải thông cần thiết kế; $h^{LP}(n)$ là đáp ứng xung của bộ lọc thông thấp tần số cắt $(w_2 - w_1)/2$
- $h^{BS}(n) = \delta(n) - h^{BP}(n)$, trong đó $h^{BP}(n)$ là đáp ứng xung của bộ lọc thông dải w_1, w_2

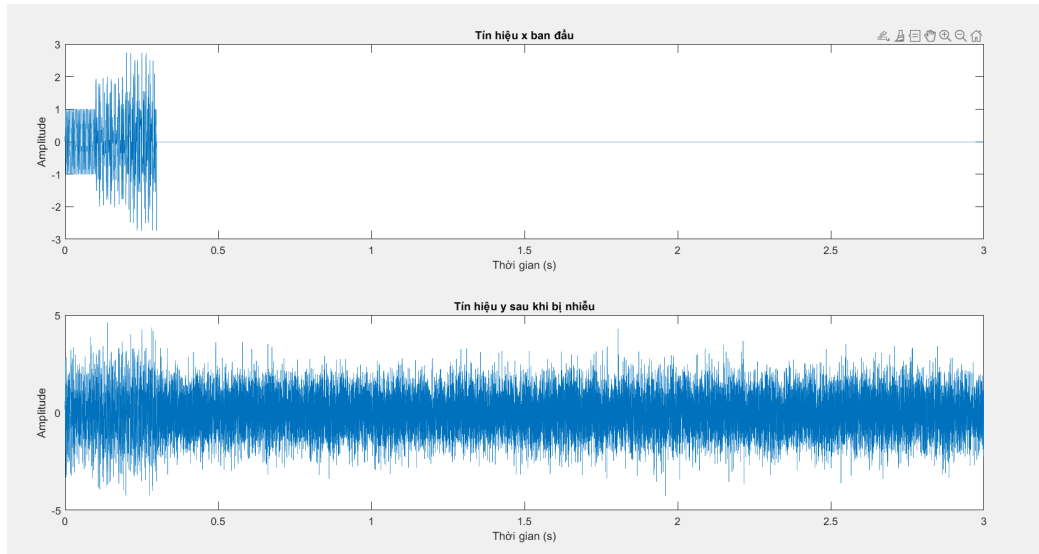
▼ Bài tập

▼ Câu 1

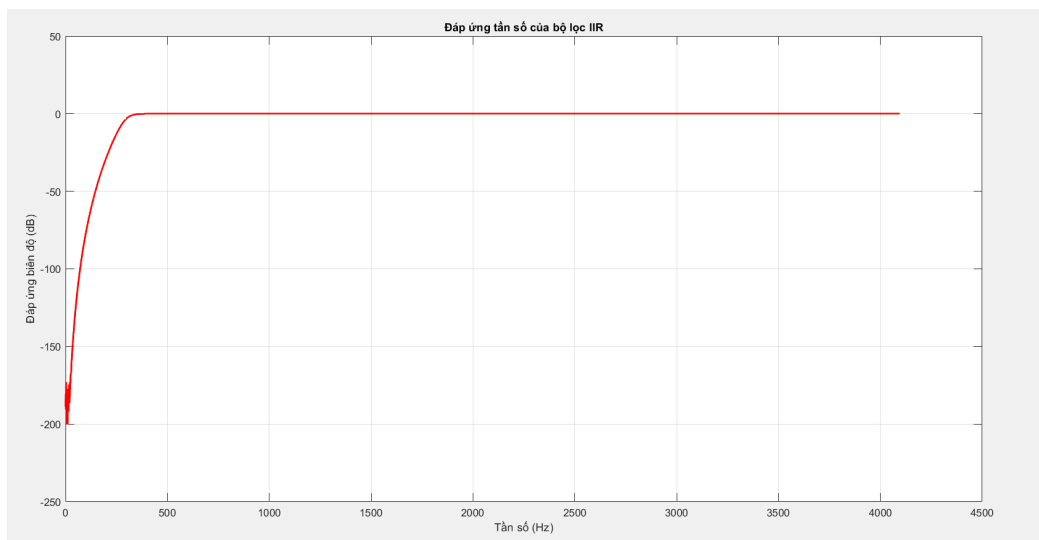


▼ Câu 2

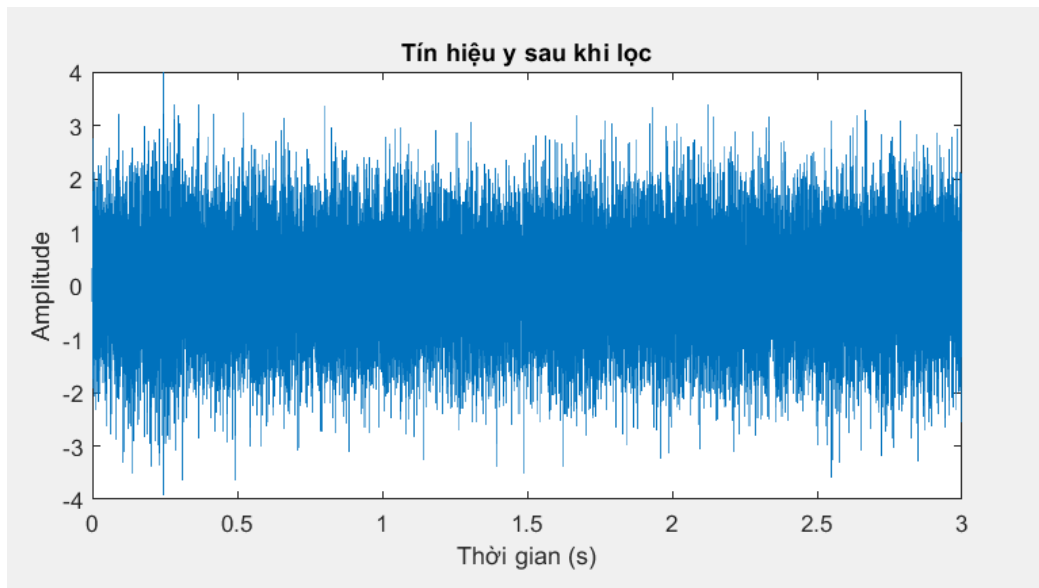
- Vẽ tín hiệu x ban đầu và tín hiệu y sau khi bị nhiễu



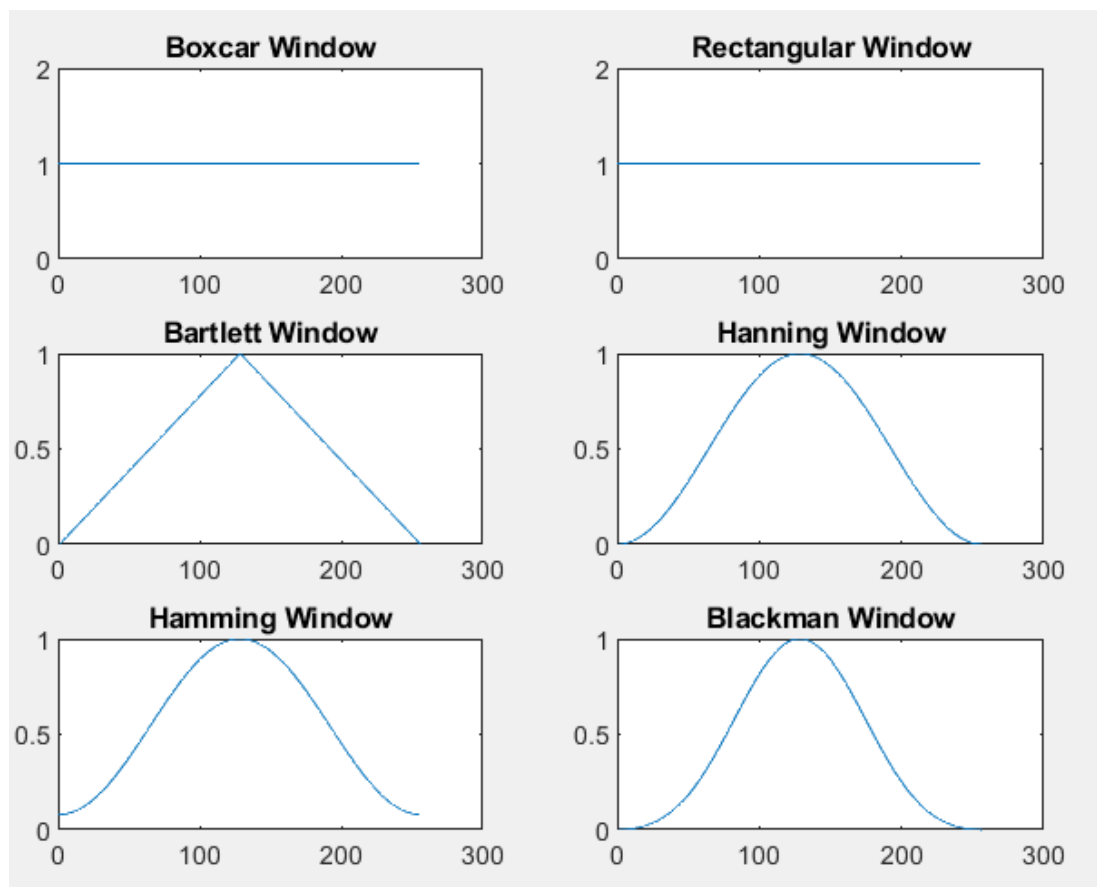
- b. Thiết kế bộ phát IIR đáp ứng tùy ý để loại bỏ thành phần 300Hz khỏi tín hiệu ba âm có nhiễu trắng y. Vẽ đáp ứng tần số của bộ lọc



- c. Lọc tín hiệu y với bộ chỉnh âm được thiết kế.

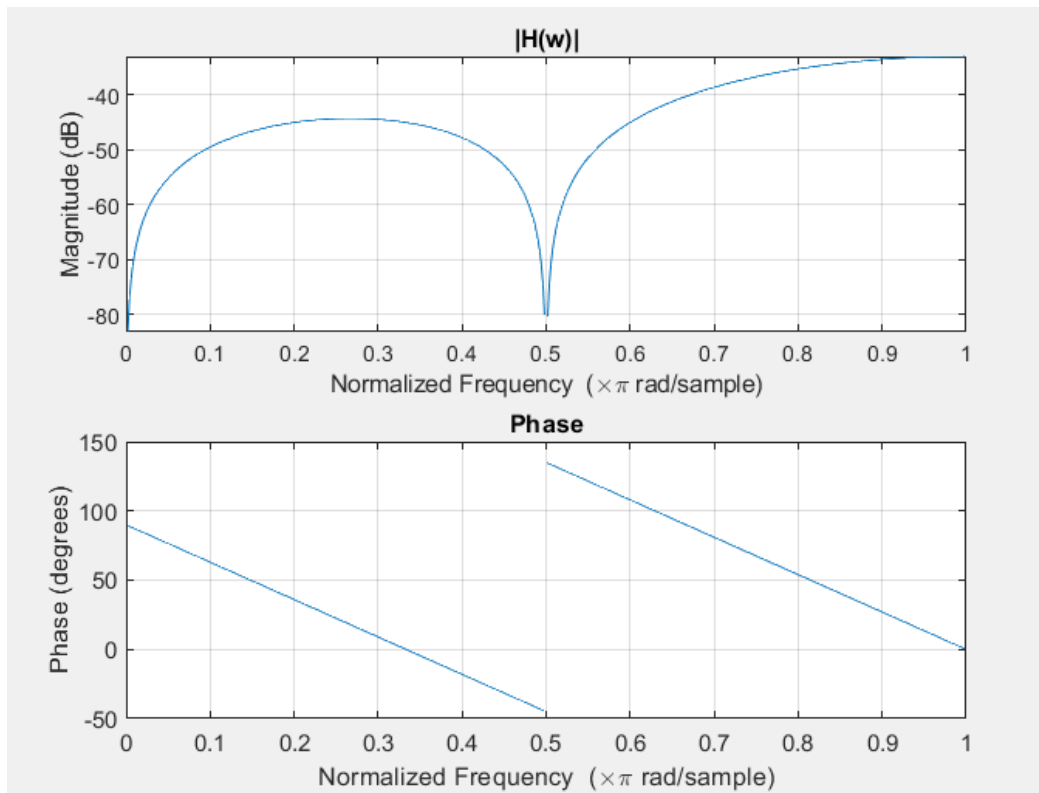


▼ Câu 3

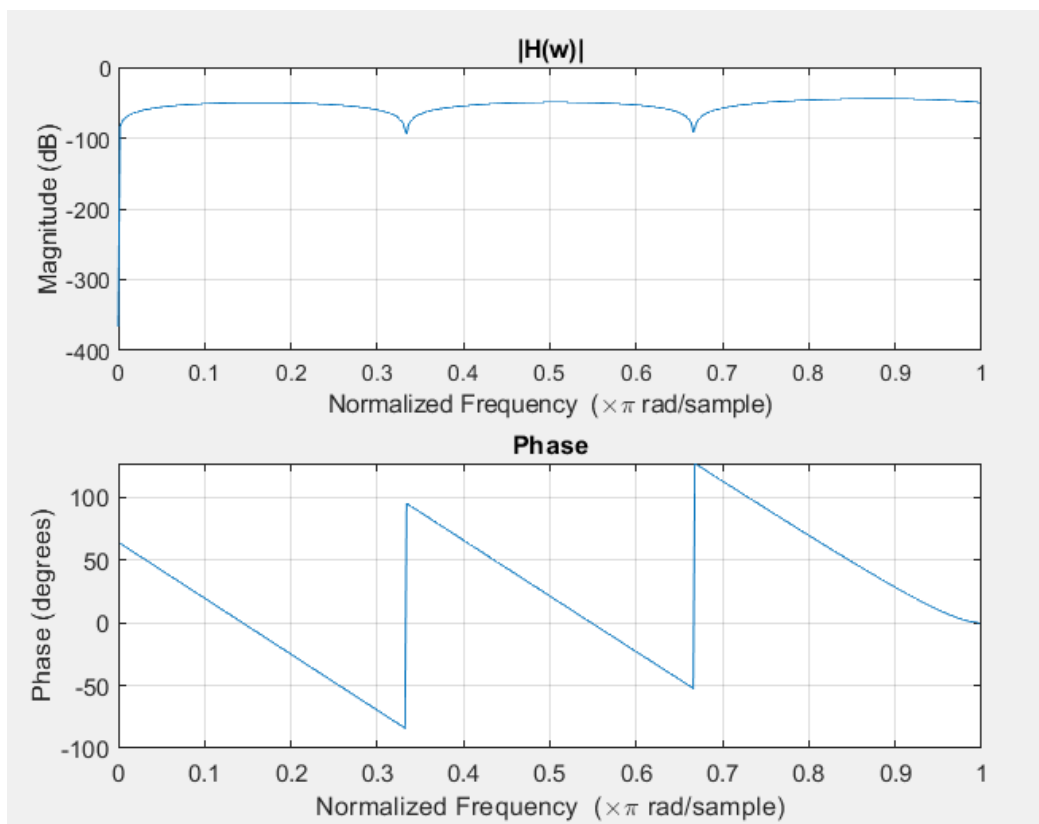


▼ Câu 5

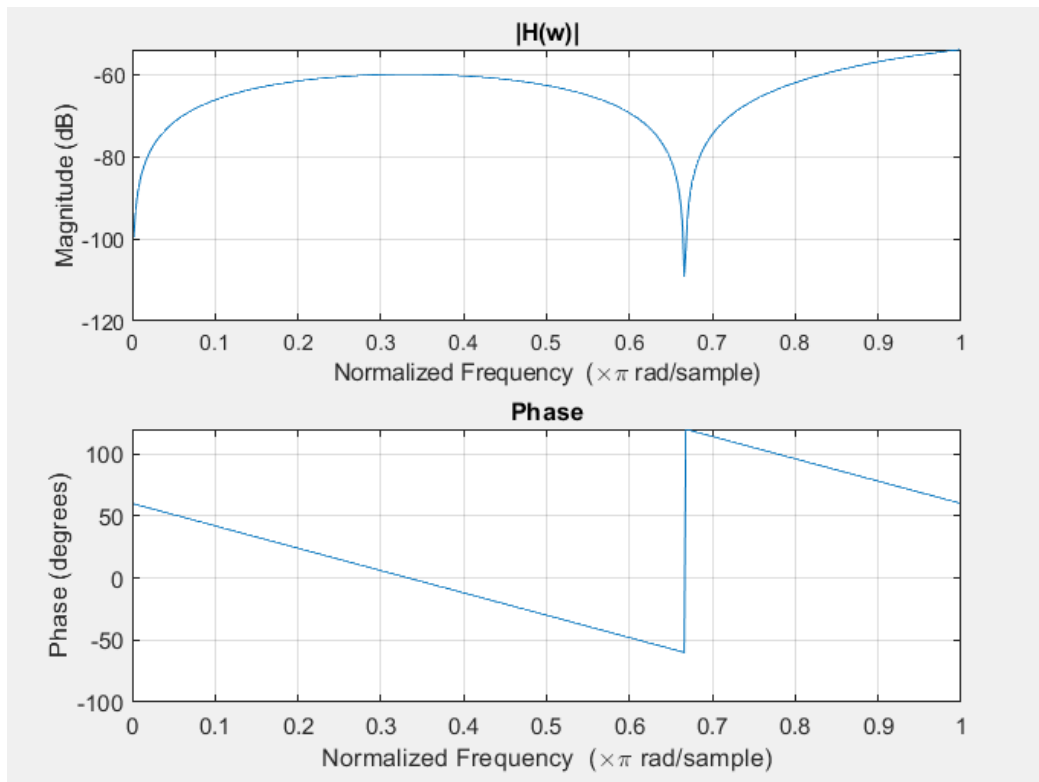
- $w_p = 0.3\pi$, $w_s = 0.6\pi$, $A_p = 0.5dB$, $A_s = 45dB$



- $w_p = 0.2\pi$, $w_s = 0.75\pi$, $A_p = 0.2\text{dB}$, $A_s = 50\text{dB}$



- $w_p = 0.4\pi$, $w_s = 0.6\pi$, $A_p = 0.25\text{dB}$, $A_s = 60\text{dB}$



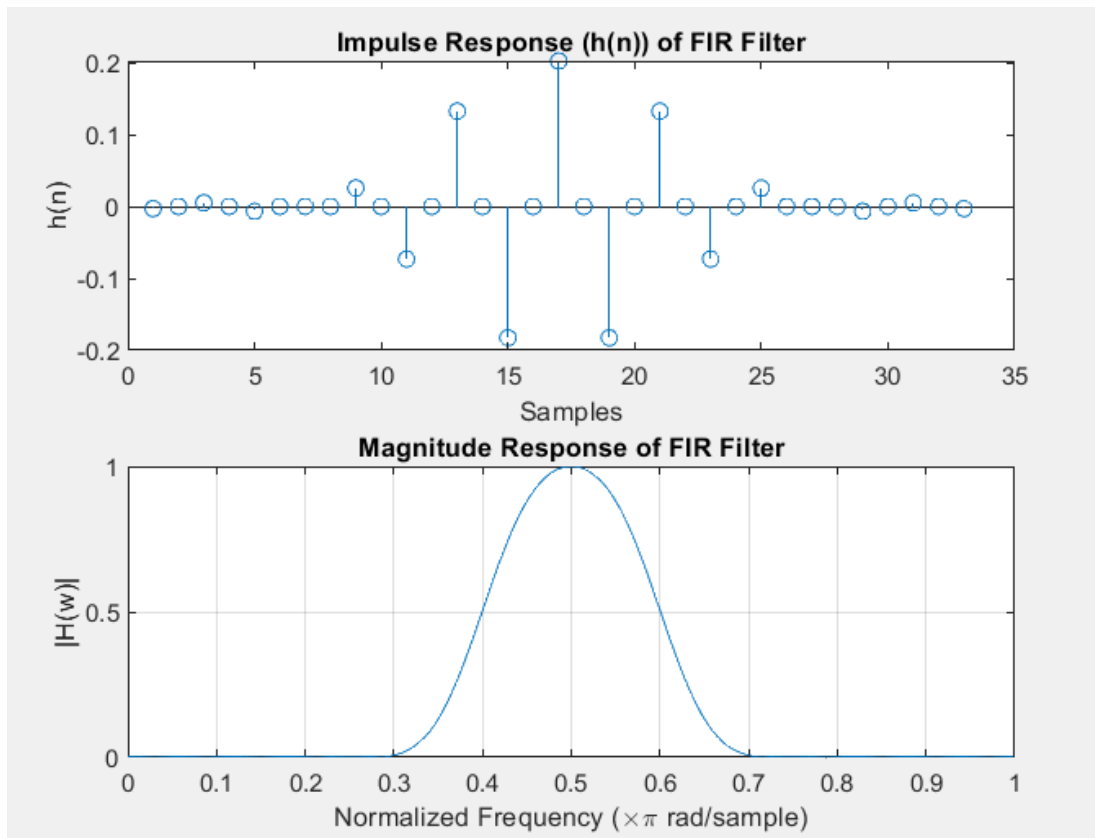
▼ Câu 6

Sử dụng lệnh `fir1`, tuy nhiên cần xác định trước các tham số của bộ lọc: tần số cắt chuẩn hóa (0,1), bậc hoặc chiều dài của bộ lọc, loại bộ lọc.

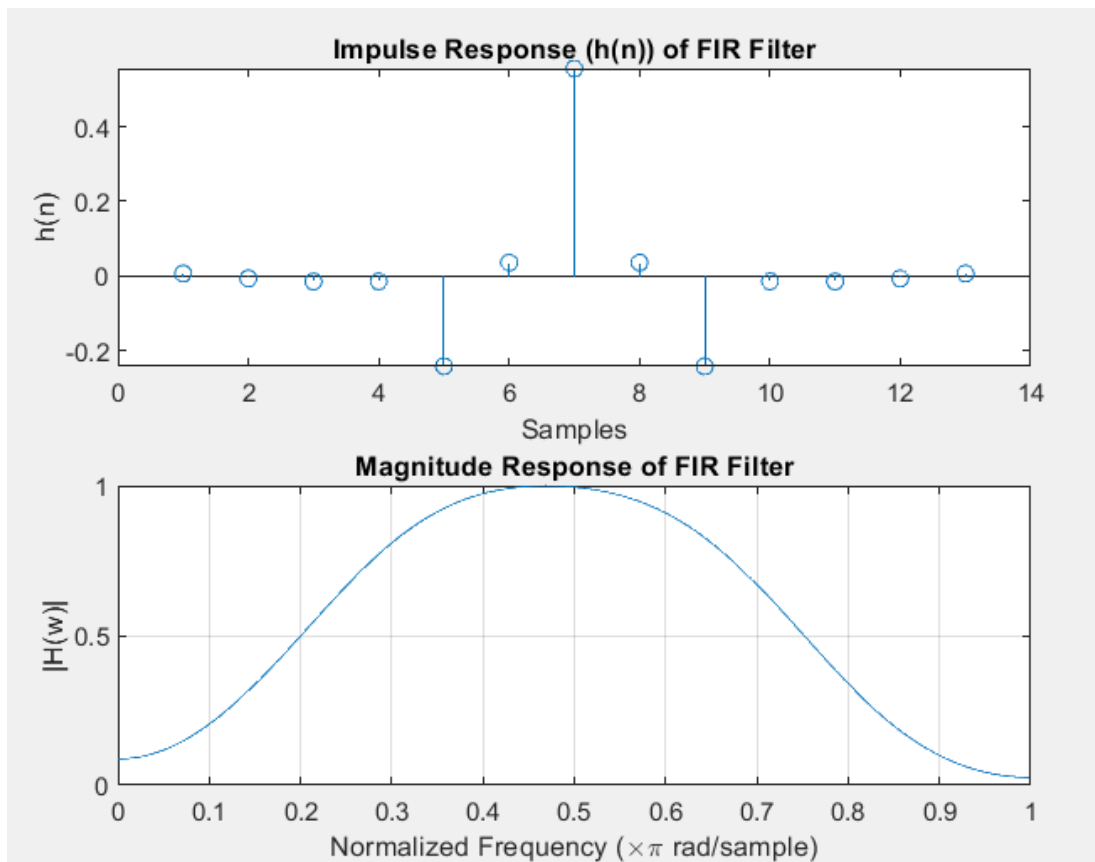
Công thức tính chiều dài của số như trên.

▼ Câu 7

- $\omega_p = 0.3\pi, \omega_s = 0.5\pi, A_p = 0.5dB, A_s = 45dB$



- $\omega_p = 0.2\pi, \omega_s = 0.75\pi, A_p = 0.2dB, A_s = 50dB$



- $\omega_p = 0.4\pi, \omega_s = 0.6\pi, A_p = 0.25dB, A_s = 60dB$

