

# PROJETO DE FILTROS FIR

## 1. Introdução

FILTRAGEM: alteração das amplitudes das componentes frequenciais de um sinal.

A função **fir1** do **Matlab** é usada no projeto e análise de filtros digitais de *Resposta Impulsiva Finita* (FIR), usando a técnica de janelamento.

Ambas funções implementam o método clássico de projeto de filtros digitais do tipo FIR com **fase linear** baseado na técnica do janelamento [1]. Ele projeta filtros nas configurações padrões do tipo passa-baixa, passa-alta, passa-faixa e rejeita-faixa. O filtro projetado é normalizado, ou seja, a *Resposta em Frequência* tem magnitude máxima igual a 0 dB na frequência central da banda passante.

A função **fir1** apresenta diversas formas de uso (chamada):

- 1) **b = fir1(N,Wn)**
- 2) **b = fir1(N,Wn,'tipofiltro')**
- 3) **b = fir1(N,Wn,janela)**
- 4) **b = fir1(N,Wn,'tipofiltro',janela)**
- 5) **b = fir1(...,'normalização')**

**Sintaxe 1: b = fir1(n,Wn)**

---

Retorna o vetor **b** contendo os **N+1** coeficientes de um filtro **passa-baixa** de ordem **N**. Trata-se de um filtro de fase linear, baseado na **janela de Hamming**, com frequência normalizada de corte **Wn**. Os coeficientes do filtro, **b**, estão ordenados em potências decrescentes de **z**:

$$b(z) = b(1) + b(2).z^{-1} + \dots + b(n+1).z^{-n}$$

**Wn** é um número entre **0** e **1**, onde **1** corresponde à frequência de Nyquist (metade da frequência de amostragem,  $fs/2$ ).

Se  $W_n$  é um vetor com dois elementos,  $W_n = [w_1 \ w_2]$ , *fir1* retorna um filtro **passafaixa** com banda passante  $w_1 < w < w_2$ .

Se  $W_n$  é um vetor multi-elementos,  $W_n = [w_1 \ w_2 \ w_3 \ w_4 \ w_5 \dots \ w_n]$ , *fir1* retorna um **filtro multibanda** de ordem  $N$  com faixas  $0 < w < w_1, w_1 < w < w_2, \dots, w_n < w < 1$ .

### **Sintaxe 2: $b = \text{fir1}(N, W_n, \text{'tipofiltro'})$**

---

Nesta forma de uso deve-se especificar o tipo de filtro a ser projetado:

- **'high'** para **filtro passa alta** com frequência de corte  $W_n$ .
- **'stop'** para **filtro rejeita faixa**, com  $W_n = [w_1 \ w_2]$  para a faixa de rejeição.
- **'DC-0'** para filtro multibanda, com primeira banda sendo do tipo rejeição.
- **'DC-1'** para filtro multibanda, com primeira banda sendo do tipo passante.

A função **fir1** sempre projeta **filtro de ordem par** para configurações passa alta e rejeita faixa, pois em filtros de ordem ímpar, a *Resposta em Frequência* na frequência de Nyquist é nula, o que não é apropriado aos filtros passa alta e rejeita faixa.

### **Sintaxe 3: $b = \text{fir1}(N, W_n, \text{janela})$**

---

Usa a janela especificada no vetor coluna **janela** para projetar o filtro. Esse vetor deve ter  $N+1$  elementos. Se nenhuma janela é especificada, **fir1** usa a janela de *Hamming*.

### **Sintaxe 4: $b = \text{fir1}(n, W_n, \text{'tipofiltro'}, \text{janela})$**

---

Idem às sintaxes 2 e 3 simultaneamente.

### **Sintaxe 5: $b = \text{fir1}(\dots, \text{'normalização'})$**

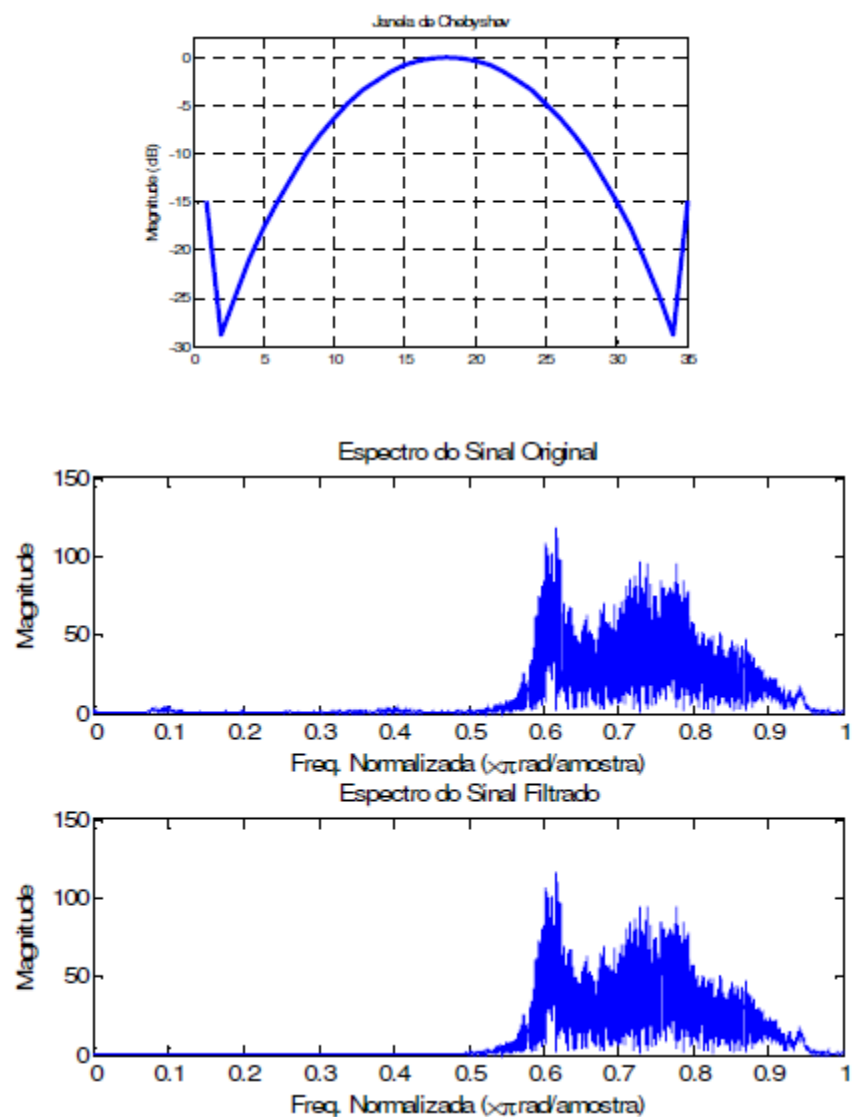
---

Indica se a magnitude do filtro deve ser normalizada, admitindo os valores: **'scale'** (*default*) se a normalização for desejada (a magnitude da resposta em frequência do

filtro na frequência central da banda passante é 0 dB); **'noscale'** se a normalização não for desejada.

O **atraso de grupo** do filtro FIR projetado por **fir1** é  $n/2$  amostras.

**Exemplo 1:** O arquivo *chirp.mat* do Matlab contém um sinal  $y[n]$  com a gravação do gorjeio de um pássaro. Esse sinal tem a maior parte de sua potência acima da frequência  $fs/4$  (metade da frequência de Nyquist). Projete um filtro FIR passa alta de ordem 34 para atenuar as componentes do sinal abaixo de  $fs/4$ . Use frequência de corte  $0,48\pi$  rad, e janela de Chebyshev, com atenuação mínima de 30 dB na banda de rejeição.

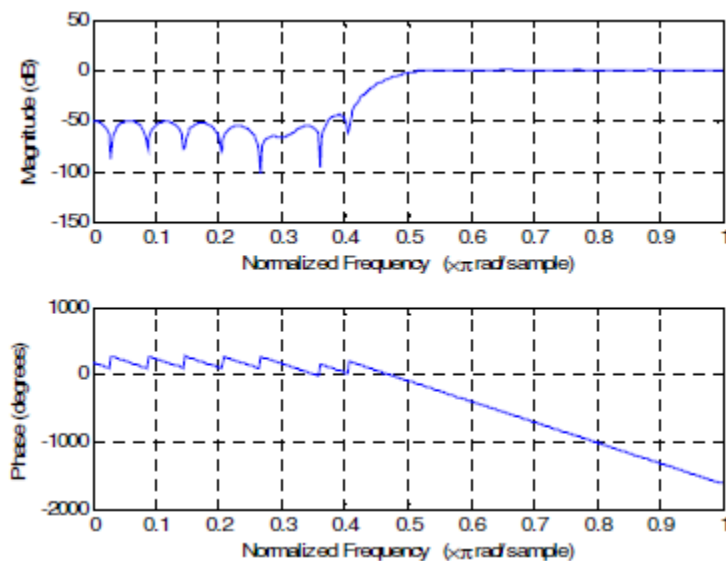


```
load chirp % som monofônico, -1 <= y <= +1, fs = 8192 Hz ou amostra/s
amostras = length(y);
```

```

w = 0:1/(amostras/2):1*(1-1/(amostras/2)); % freq.Dig.: 0-pi (rad/amostra)
% Filtro Passa Alta, com janela de Chebyshev, ordem 34, wc = 0.48*pi rad/am
% fc = 0.48*fs/2 = 1966 Hz
b = fir1(34,0.48,'high',chebwin(35,30)); % coef.s do filtro FIR desejado
freqz(b,1,512); % resposta em frequência do filtro digital
f = filter(b,1,y); % filtragem do sinal original y(n)
sound(y) % som original
pause(3)
sound(f) % som filtrado
Y = fft(y); % espectro do sinal original
F = fft(f); % espectro do sinal filtrado
figure; o = length(w);
subplot(2,1,1), plot(w,abs(Y(1:o))); title('Espectro do Sinal Original');
xlabel('Freq. Normalizada (\times\pi rad/amostra)'); ylabel('Magnitude');
subplot(2,1,2), plot(w,abs(F(1:o))); title('Espectro do Sinal Filtrado');
xlabel('Freq. Normalizada (\times\pi rad/amostra)'); ylabel('Magnitude');

```



Outras funções janelas disponíveis no *Matlab* estão disponíveis via função:

**window(@WNAME,N)**: retorna uma janela de **N** pontos do tipo especificado pelo manipulador de função **@WNAME** em um vetor coluna. **@WNAME** pode ser qualquer nome de função janela reconhecido pelo Matlab.

@bartlett - Janela de Bartlett  
@barthannwin - Janela de Bartlett-Hanning  
@blackman - Janela de Blackman  
@blackmanharris - Janela Mínima de Blackman-Harris com 4 termos  
@bohmanwin - Janela de Bohman  
@chebwin - Janela de Chebyshev  
@flattopwin - Janela Flat Top  
@gausswin - Janela Gaussiana  
@hamming - Janela de Hamming  
@hann - Janela de Hann  
@kaiser - Janela de Kaiser  
@nuttallwin - Janela de Nuttall  
@parzenwin - Janela de Parzen (de la Valle-Poussin)  
@rectwin - Janela Retangular (boxcar)  
@taylorwin - Janela de Taylor  
@tukeywin - Janela de Tukey  
@triang - Janela Triangular

## 2. Procedimentos Experimentais

1. Para o mesmo sinal usado no Exemplo 1, realize nova filtragem passa alta, modificando a frequência de corte do filtro projetado  $3.f_s/8$ .
2. Compare os espectros de frequência dos sinais filtrados no Exemplo 1 e no item anterior, e identifique as mudanças ocorridas no sinal de áudio (ouvindo ambos os sinais).
3. Projete um filtro FIR passa faixa de ordem 48, com banda passante de  $0,65 \leq \Omega \leq 0,75$  (rad/amostra). Repita o item anterior.

## 3. Questões

1. Descreva a função *fir2* (Matlab).
2. O que se pode dizer do volume de som do áudio resultante dos itens do procedimento 2 e 3? O que pode ser feito para se manter o volume de som do áudio original em ambos os casos?