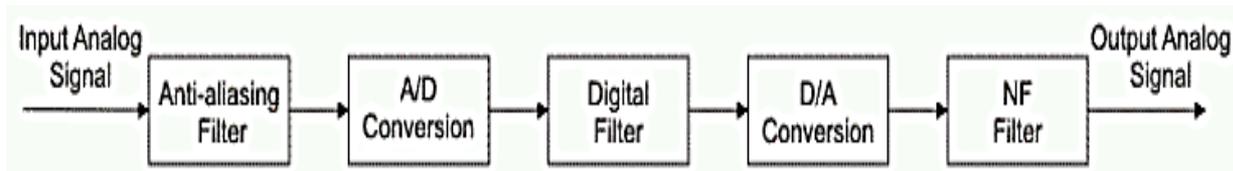


Projeto de Filtro Digital

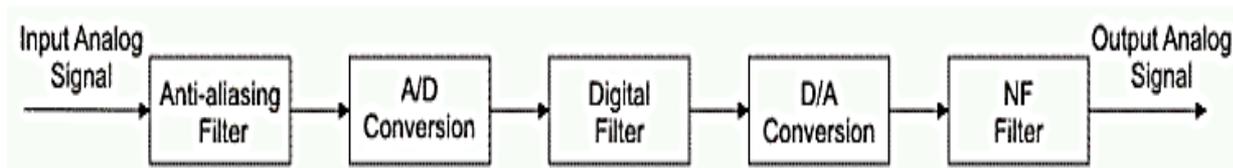
Conceitos básicos

- O processo de conversão de um sinal analógico para o formato digital é realizado a partir de uma amostragem com uma frequência de amostragem finita f_s .
- Se um sinal de entrada contiver componentes de frequência mais altas que a metade da frequência de amostragem ($f_s/2$), isso causará distorções no espectro original.
 - Esta é a razão porque é necessário realizar a filtragem de um sinal de entrada usando um filtro passa baixa, que elimina as componentes de alta frequência no espectro da frequência de entrada.
 - Este filtro é chamado de *anti-aliasing* pois previne as distorções conhecidas como *aliasing*.



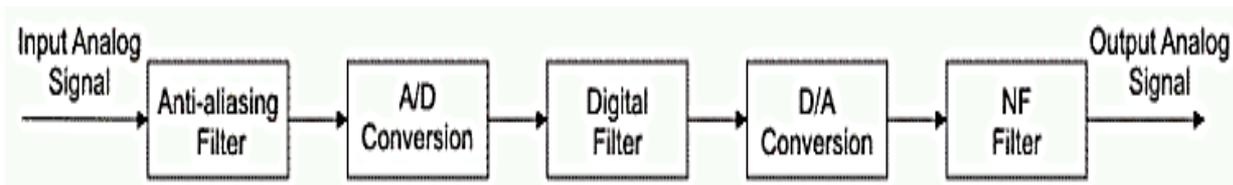
Conceitos básicos

- Após o processo de filtragem e amostragem, o sinal digital está pronto para o próximo processamento que, neste caso, é a filtragem por meio do filtro digital apropriado.
- O sinal de saída é também um sinal digital que, em alguns casos, precisa ser convertido de volta ao formato analógico.
- Após a conversão digital-analógica, o sinal contém algumas componentes de frequência mais altas que $f_s/2$ que precisam ser eliminadas.
 - Novamente, é necessário o uso de um filtro passa-baixa com frequência de amostragem $f_s/2$.



Conceitos básicos

- A atenuação do filtro digital é usualmente expressa em escala logarítmica (dB).
- A atenuação medida em decibéis pode ser obtida usando-se a seguinte expressão genérica: $a = 20 * \log(H(f))$



Tipos de filtros digitais

- Os filtros digitais podem ser classificados em vários grupos, dependendo do critério de classificação. Os dois tipos mais importantes são os de resposta impulsiva finita e infinita, *finite impulse response* (FIR) e *infinite impulse response* (IIR).
- Ambos os tipos apresentam vantagens e desvantagens e devem ser considerados cuidadosamente quando do projeto do filtro. Além disso, é necessário levar em conta todas as características fundamentais do sinal a ser filtrado. Na maioria dos casos, a característica que realmente importa é se é necessário que o filtro tenha característica de fase linear ou não.
- Sinal de voz, por exemplo, pode ser processado em sistemas com característica de fase não linear. A fase não é essencial para um sinal de voz e portanto pode ser desprezada, o que resulta na possibilidade de uso de uma gama mais ampla de sistemas para seu processamento.
- Existem sinais onde a característica de fase é essencial. Um exemplo típico é o sinal obtido de vários sensores numa indústria. Portanto, é necessário o uso de um filtro com característica de fase linear para prevenir perda de informação importante.

Escolha de filtros digitais

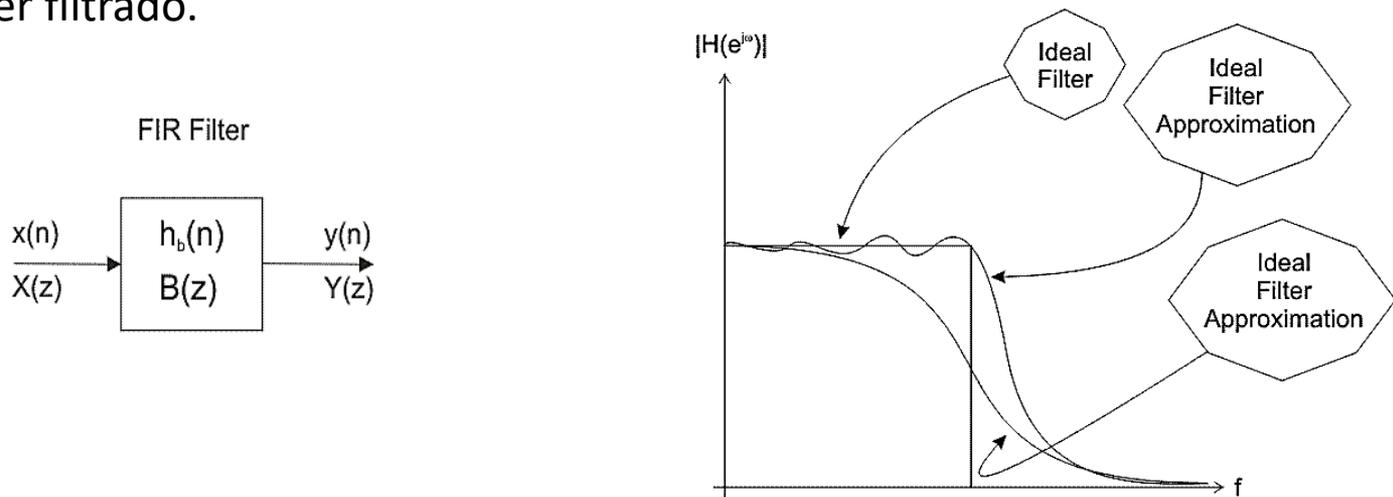
- Considerar a característica de fase pode facilitar a decisão sobre qual tipo de filtro digital é melhor para ser usado:
 - Se a fase é uma característica essencial, o uso de filtros FIR é indicado pois eles têm uma característica de fase linear. Tais filtros são de ordem mais alta e mais complexo, portanto.
 - Por outro lado, quando é apenas a resposta em frequência que importa, é preferível o uso de filtros digitais do tipo IIR, que envolvem uma ordem mais baixa, i.e., são menos complexos e portanto mais fáceis de serem implementados.

Sumário das características

- FIR
 - Característica de fase linear;
 - Filtro de ordem mais alta (circuitos mais complexos); e
 - estabilidade.
- IIR
 - característica de fase não linear;
 - filtro de ordem mais baixa (circuitos menos complexos); e
 - o filtro digital resultante tem o potencial de se tornar instável.

Filtros FIR

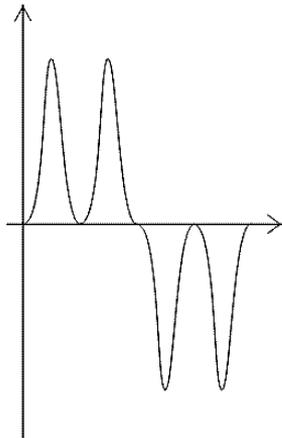
- Também conhecidos como **filtros digitais não recursivos** pois não possuem realimentação (parte recursiva de um filtro), ainda que algoritmos recursivos possam ser usados para sua realização.
- Podem ser projetados usando-se diferentes métodos, mas a maioria deles é baseada na aproximação de um filtro ideal. O objetivo não é obter características ideais, o que é impossível, mas obter características suficientemente boas.
 - A função de transferência do filtro FIR aproxima-se do ideal na medida em que a ordem do filtro aumenta o que também aumenta sua complexidade e quantidade de tempo necessário para o processamento das amostras do sinal a ser filtrado.



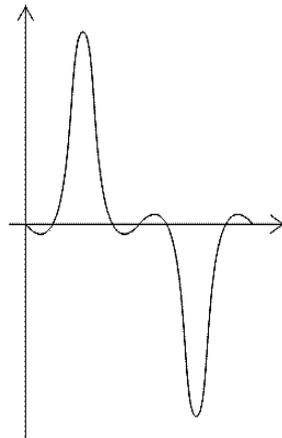
Filtros FIR

- Aqui serão descritos os métodos para projeto de filtro FIR mais populares que usam funções de janelas. A característica da função de transferência, bem como seu desvio da resposta de frequência ideal, depende da ordem do filtro e da janela usada.
- A figura ilustra os sinais de entrada (esquerda) e saída (direita) de sistemas de fase não linear. Suponhamos que um sistema induza uma mudança de fase de 0 radianos na frequência ω , e π radianos quando a frequência triplica. O sinal de entrada consiste de uma frequência natural ω e uma harmônica com mesma amplitude na frequência três vezes maior. Estes dois sinais têm formas de onda diferentes; a potência dos sinais não muda, nem a amplitude dos harmônicos, apenas a fase do segundo harmônico muda.

$$\sin(\omega t) + \sin(3\omega t)$$



$$\sin(\omega t) - \sin(3\omega t)$$



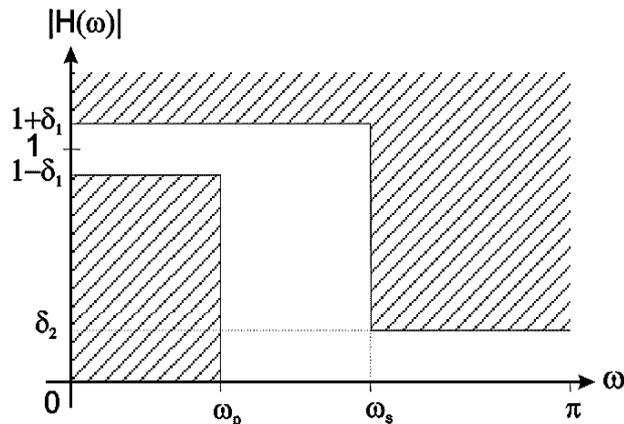
Se assumimos que a entrada é um sinal de voz cuja característica de fase não é essencial, esta distorção na fase do sinal não será importante. Neste caso, o sistema satisfaz a todos os requisitos necessários. Entretanto, se a característica de fase é importante, esta distorção deve ser evitada.

Filtros FIR

- Uma das dificuldades dos filtros FIR é a ordem alta do filtro projetado, marcadamente mais alta se comparada à de um filtro IIR com mesma resposta impulsiva. Por este motivo, somente quando a característica de fase linear é muito importante é que usamos os filtros FIR.
- O número de linhas de atraso contidas no filtro, i.e. um número de amostras da entrada que devem ser salvas com o propósito do cálculo das amostras de saída, determina a ordem do filtro. Por exemplo, se assumimos um filtro de ordem 10, isso significa que é necessário salvarmos 10 amostras da entrada precedentes à amostra atual. Todas as 11 amostras irão afetar a amostra de saída do filtro FIR.
- A função de transferência de um filtro FIR típico pode ser expressa como um polinômio de uma variável complexa z^{-1} . Todos os polos desta função estão localizados na origem. Por esta razão, filtros FIR são seguramente estáveis, enquanto os filtros IIR podem se tornar instáveis.

FIR: Métodos de Projeto

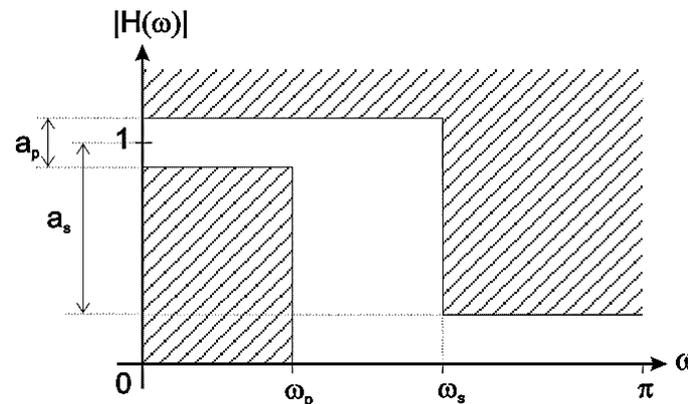
- Especificação do filtro digital **passa baixa**
 - ω_p – Frequência de corte normalizada na *passband*;
 - ω_s – Frequência de corte normalizada na *stopband*;
 - δ_1 – máximo ripple na *passband* ;
 - δ_2 – mínima atenuação na *stopband* [dB];
 - a_p – máximo ripple na *passband*; e
 - a_s – mínima atenuação na *stopband* [dB].



$$a_p = 20 \log_{10} \left(\frac{1 + \delta_1}{1 - \delta_1} \right)$$

$$\omega = \frac{2\pi f}{f_s}$$

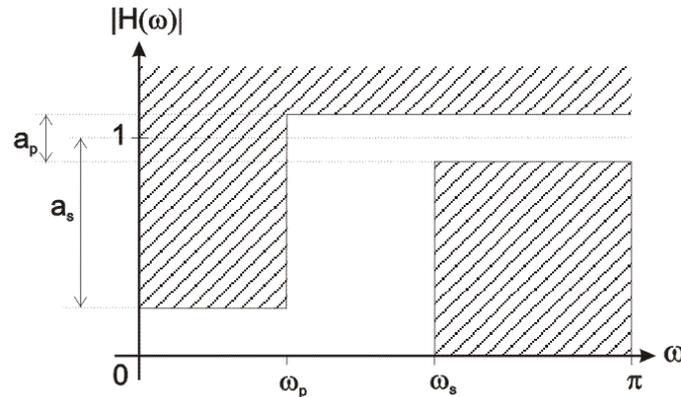
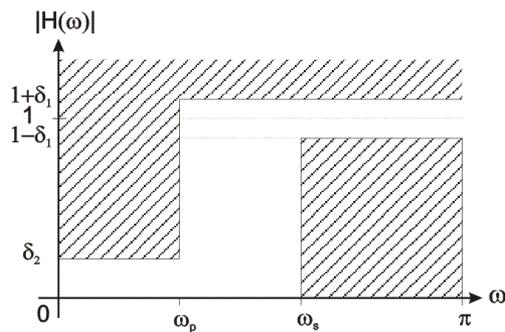
$$a_s = -20 \log_{10} \delta_2$$



f_s é a frequência de amostragem;
 f é a frequência a ser normalizada; e
 ω é a frequência normalizada.

FIR: Métodos de Projeto

- Especificação do filtro digital **passa alta**
 - ω_p – Frequência de corte normalizada na *passband*;
 - ω_s – Frequência de corte normalizada na *stopband*;
 - δ_1 – máximo ripple na *passband* ;
 - δ_2 – mínima atenuação na *stopband* [dB];
 - a_p – máximo ripple na *passband*; e
 - a_s – mínima atenuação na *stopband* [dB].



$$a_p = 20 \log_{10} \left(\frac{1 + \delta_1}{1 - \delta_1} \right)$$

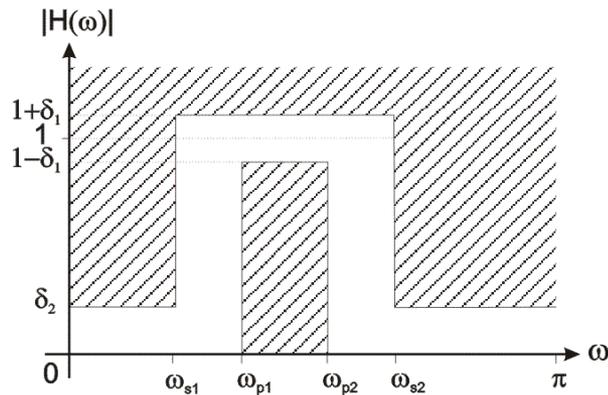
$$\omega = \frac{2\pi f}{f_s}$$

$$a_s = -20 \log_{10} \delta_2$$

f_s é a frequência de amostragem;
 f é a frequência a ser normalizada; e
 ω é a frequência normalizada.

FIR: Métodos de Projeto

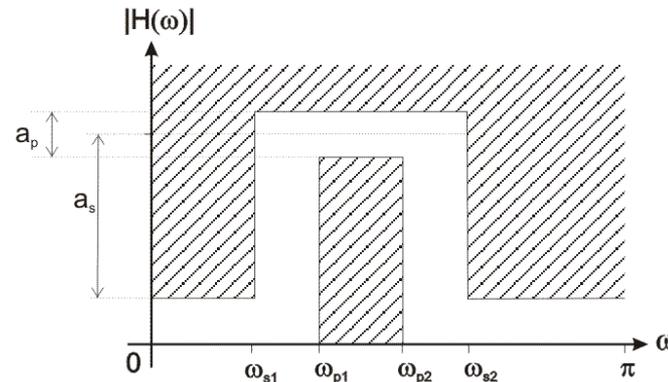
- Especificação do filtro digital **passa banda**
 - ω_p – Frequência de corte normalizada na *passband*;
 - ω_s – Frequência de corte normalizada na *stopband*;
 - δ_1 – máximo ripple na *passband* ;
 - δ_2 – mínima atenuação na *stopband* [dB];
 - a_p – máximo ripple na *passband*; e
 - a_s – mínima atenuação na *stopband* [dB].



$$a_p = 20 \log_{10} \left(\frac{1 + \delta_1}{1 - \delta_1} \right)$$

$$\omega = \frac{2\pi f}{f_s}$$

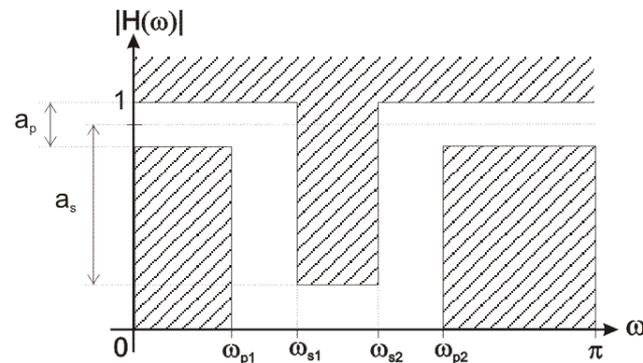
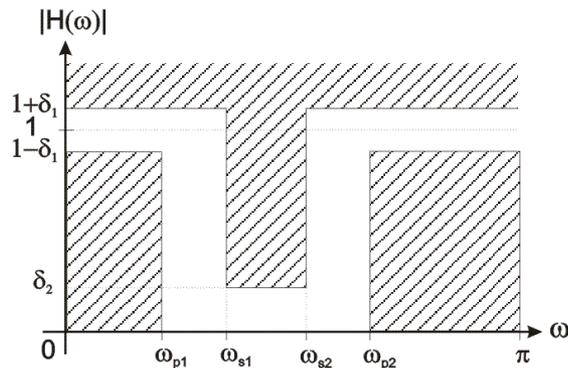
$$a_s = -20 \log_{10} \delta_2$$



f_s é a frequência de amostragem;
 f é a frequência a ser normalizada; e
 ω é a frequência normalizada.

FIR: Métodos de Projeto

- Especificação do filtro digital **bloqueia banda**
 - ω_p – Frequência de corte normalizada na *passband*;
 - ω_s – Frequência de corte normalizada na *stopband*;
 - δ_1 – máximo ripple na *passband* ;
 - δ_2 – mínima atenuação na *stopband* [dB];
 - a_p – máximo ripple na *passband*; e
 - a_s – mínima atenuação na *stopband* [dB].



$$a_p = 20 \log_{10} \left(\frac{1 + \delta_1}{1 - \delta_1} \right)$$

$$\omega = \frac{2\pi f}{f_s}$$

$$a_s = -20 \log_{10} \delta_2$$

f_s é a frequência de amostragem;
 f é a frequência a ser normalizada; e
 ω é a frequência normalizada.

Transformada Z

- A transformada Z é operada sobre sinais discretos no tempo. Ela converte um sinal discreto no domínio do tempo em uma representação complexa no domínio da frequência.
- É muito adequada para análise de sinais e sistemas discretos no tempo, sendo derivada da transformada de Fourier discreta no tempo e é considerada a operação básica no processo de projeto de um filtro digital. A transformada Z é definida como:

$$X(z) = \sum_{n=-\infty}^{\infty} x(n)z^{-n}$$

onde z é um número complexo.

Transformada Z

- Exemplo:

– Considere que as amostras de um sinal discreto no tempo $x(n)$ são conhecidas. Obter a transformada deste sinal através da transformada z e da transformada de Fourier.

$$x(n) = \{1, 2, 3, 4, 5, 4, 3, 2, 1\} ; 0 \leq n \leq 8$$

A transformada z é definida como: $X(z) = \sum_{n=-\infty}^{\infty} x(n)z^{-n}$

$$X(z) = 1 + 2z^{-1} + 3z^{-2} + 4z^{-3} + 5z^{-4} + 4z^{-5} + 3z^{-6} + 2z^{-7} + z^{-8}$$

O que leva a:

$$X(z) = \frac{z^8 + 2z^7 + 3z^6 + 4z^5 + 5z^4 + 4z^3 + 3z^2 + 2z + 1}{z^8}$$

Transformada Z

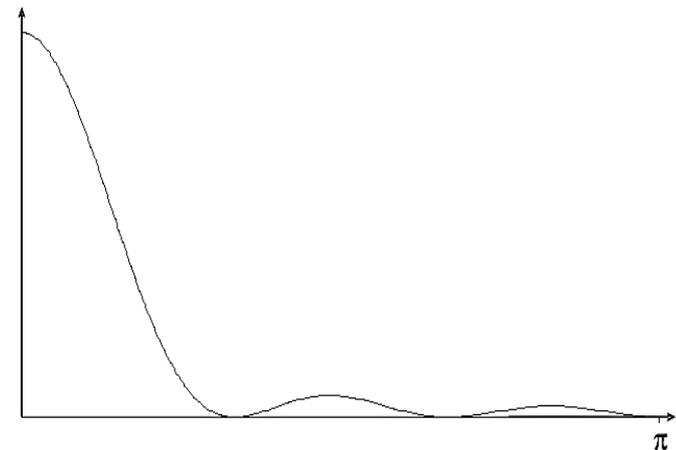
$$X(z) = 1 + 2z^{-1} + 3z^{-2} + 4z^{-3} + 5z^{-4} + 4z^{-5} + 3z^{-6} + 2z^{-7} + z^{-8}$$

- Exemplo (continuação):

$$X(z) = \frac{z^8 + 2z^7 + 3z^6 + 4z^5 + 5z^4 + 4z^3 + 3z^2 + 2z + 1}{z^8}$$

- A transformada de Fourier pode ser obtida reescrevendo-se a expressão anterior em termos de z , fazendo $z=e^{j\omega}$. O que leva a:

$$X(e^{j\omega}) = \frac{e^{j8\omega} + 2e^{j7\omega} + 3e^{j6\omega} + 4e^{j5\omega} + 5e^{j4\omega} + 4e^{j3\omega} + 3e^{j2\omega} + 2e^{j\omega} + 1}{e^{j8\omega}}$$

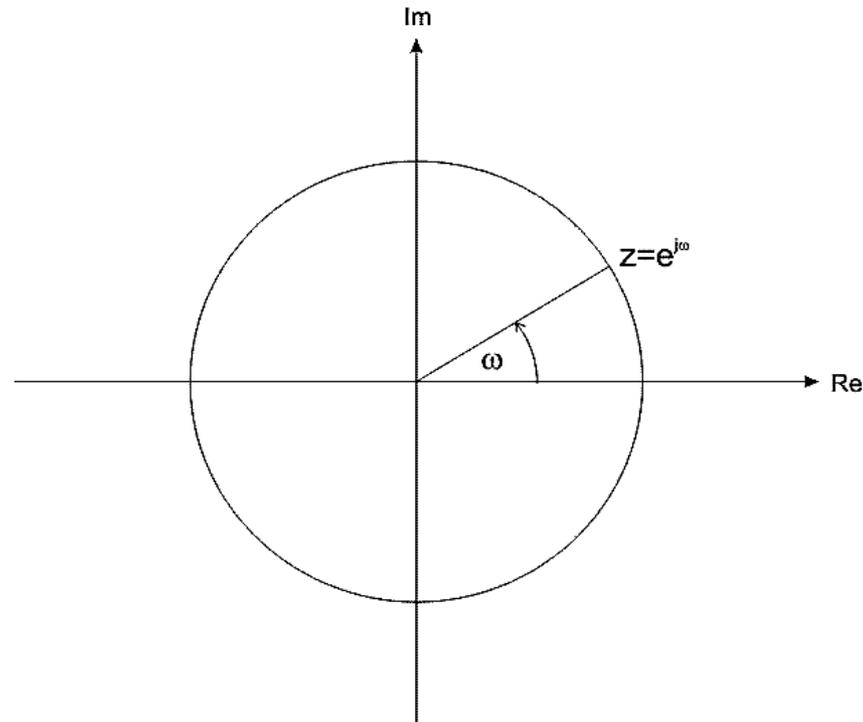


$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n} = X(z = e^{j\omega}) \quad z = re^{j\omega}$$

Estas duas últimas expressões nos levam a concluir que a transformada de Fourier é apenas um caso especial da transformada z para $r=1$.

Transformada Z

- No plano z , a transformada de Fourier é representada como um círculo unitário



Função de transferência de sistemas discretos no tempo

- A transformada Z é primariamente usada para encontrar a função de transferência de sistemas lineares discretos no tempo. Quando a função de transferência é obtida, é necessário considerar seus zeros e polos no plano z. A função de transferência de sistemas discretos no tempo é definida como:

$$H(z) = \frac{\sum_{i=0}^{M-1} b_i z^{-i}}{\sum_{j=0}^{N-1} a_j z^{-j}} = H_0 \frac{\prod_{i=0}^{M-1} (1 - q_i z^{-i})}{\prod_{j=0}^{N-1} (1 - p_j z^{-j})}$$

onde:

b_i são os coeficientes *feedforward* do filtro (parte não recursiva);
 a_j são os coeficientes de realimentação (*feedback*) (parte recursiva);
 H_0 é uma constante;
 q_i são os zeros da função de transferência; e
 p_j são os polos da função de transferência.

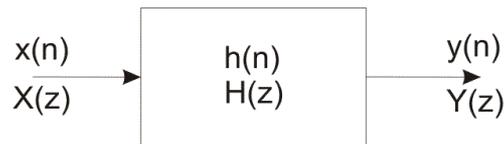
- A parte recursiva da função de transferência é, na verdade, uma realimentação do sistema discreto no tempo. Como mencionado anteriormente, filtros FIR não têm esta parte recursiva da função de transferência, assim a expressão acima pode ser simplificada como segue:

$$H(z) = \sum_{i=0}^{M-1} b_i z^{-i} = H_0 \prod_{i=0}^{M-1} (1 - q_i z^{-i})$$

Função de transferência de sistemas discretos no tempo

- A resposta impulsiva de um sistema discreto no tempo é obtida a partir da transformada inversa z da função de transferência, i.e. a função de transferência do sistema discreto no tempo é, de fato, a transformada Z da resposta impulsiva:

$$H(z) = \sum_{n=-\infty}^{\infty} h(n)z^{-n} \quad \text{onde } h(n) \text{ é a resposta impulsiva do sistema discreto no tempo}$$

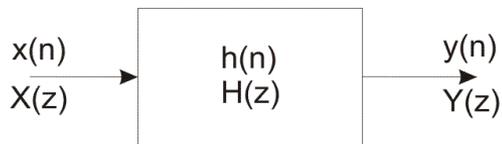


- No domínio do tempo, o sistema discreto no tempo, mostrado na figura, pode ser expresso através da convolução do sinal de entrada $\mathbf{x(n)}$ e da resposta impulsiva $\mathbf{h(n)}$ do sistema:

$$y(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k)$$

Função de transferência de sistemas discretos no tempo

- No domínio da frequência, o sistema discreto no tempo, mostrado na figura, pode ser expresso como a multiplicação da transformada Z do sinal de entrada $X(z)$ e da função de transferência $H(z)$ do sistema:



$$Y(z) = X(z)H(z)$$

- O que leva a:

$$H(z) = \frac{Y(z)}{X(z)}$$

- A primeira expressão para representar os sistemas discretos no tempo (slide anterior) é mais adequada para implementações via software, enquanto a segunda é melhor para análise, implementação via hardware e síntese, i.e. projeto de um sistema discreto no tempo.

Exemplo

- A resposta impulsiva de um filtro FIR de ordem 10 projetado usando a janela de Hamming (discussão mais adiante) é:

$$h(n) = \{0, -0.0127, -0.0248, 0.0638, 0.2761, 0.4, 0.2761, 0.0638, -0.0248, -0.0127, 0\}$$

- A função de transferência deste filtro é obtida pela transformada z da resposta impulsiva:

$$\begin{aligned} H(z) &= \sum_{n=0}^{10} h(n)z^{-n} = 0 - 0.0127z^{-1} - 0.0248z^{-2} + 0.0638z^{-3} + 0.2761z^{-4} + 0.4z^{-5} \\ &\quad + 0.2761z^{-6} + 0.0638z^{-7} - 0.0248z^{-8} - 0.0127z^{-9} + 0z^{-10} \\ &= -0.0127z^{-1} - 0.0248z^{-2} + 0.0638z^{-3} + 0.2761z^{-4} + 0.4z^{-5} \\ &\quad + 0.2761z^{-6} + 0.0638z^{-7} - 0.0248z^{-8} - 0.0127z^{-9} \end{aligned}$$

Exemplo (cont.)

- Usando a seguinte expressão:

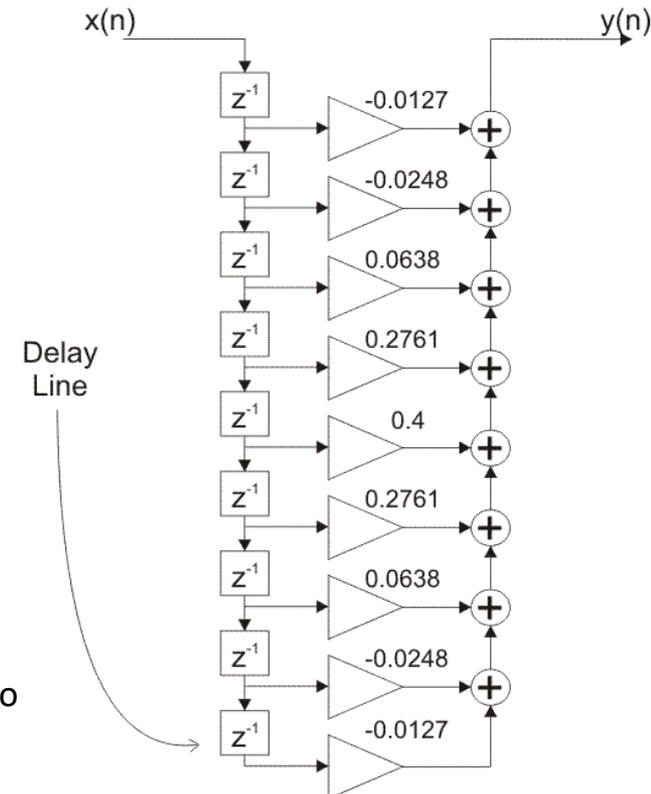
$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n} = X(z = e^{j\omega})$$

- É possível fornecer a função de transferência da frequência normalizada fixa. Se, por exemplo, $\omega = 0.2\pi$ então:

$$H(e^{j0.2\pi}) = -0.0127e^{-j0.2\pi} - 0.0248z^{-j0.4\pi} + 0.0638z^{-j0.6\pi} + 0.2761z^{-0.8\pi} + 0.4z^{-j\pi} \\ + 0.2761z^{-j1.2\pi} + 0.0638z^{-j1.4\pi} - 0.0248z^{-j1.6\pi} - 0.0127z^{-j1.8\pi}$$

$$|H(e^{j0.2\pi})| = (-0.0127 \cos(-0.2\pi) - 0.0248 \cos(-0.4\pi) + 0.0638 \cos(-0.6\pi) + 0.2761 \cos(-0.8\pi) \\ + 0.4 \cos(-\pi) \\ + 0.2761 \cos(-1.2\pi) + 0.0638 \cos(-1.4\pi) - 0.0248 \cos(-1.6\pi) - 0.0127 \cos(-1.8\pi))^2 + \\ (-0.0127 \sin(-0.2\pi) - 0.0248 \sin(-0.4\pi) + 0.0638 \sin(-0.6\pi) + 0.2761 \sin(-0.8\pi) \\ + 0.4 \sin(-\pi) \\ + 0.2761 \sin(-1.2\pi) + 0.0638 \sin(-1.4\pi) - 0.0248 \sin(-1.6\pi) - 0.0127 \sin(-1.8\pi))^2 \\ = (-0.92205)^2 + 0^2 = 0.92205$$

A realização em software requer um buffer com comprimento mínimo de 9. Buffers são normalmente circulares e seu comprimento pode ser expresso como 2^n , onde neste caso significa que o buffer circular é de comprimento $16=2^4$.

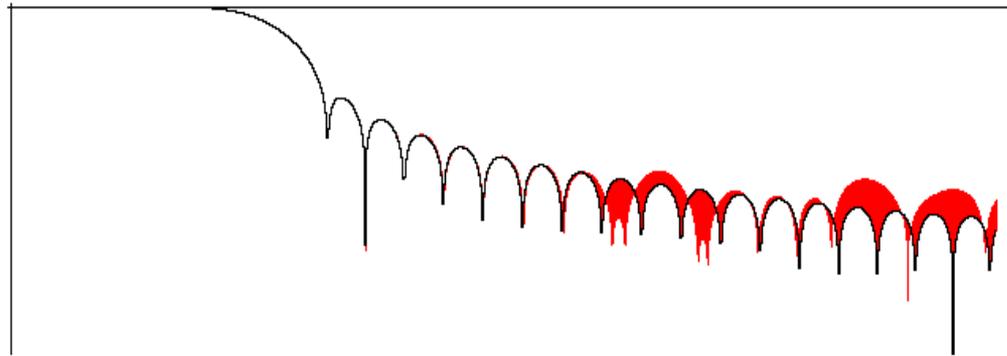


Efeitos dos polos e zeros da função de transferência

- A localização dos zeros e polos da função de transferência é muito importante para análise e síntese de sistemas discretos no tempo. De acordo com sua posição é possível testar a estabilidade de um sistema discreto no tempo, detectar erros de round-off que ocorrem devido à implementação do software bem como erros nos coeficientes encontrados durante a implementação em hardware do filtro.
- A fim de que o sistema discreto no tempo seja estável, todos os polos da função de transferência devem ser calculados dentro do círculo unitário, como mostrado na Figura 2-2-6. Se este requerimento não for satisfeito, o sistema torna-se instável, o que é muito arriscado. A localização dos zeros não afeta a estabilidade dos sistemas discretos no tempo. Lembramos que filtros FIR não têm realimentação, o que significa que a função de transferência não tem polos. Isso faz com que o filtro FIR seja sempre estável. A estabilidade do filtro será discutida mais detalhadamente no estudo dos filtros IIR.
- Um erro na representação do coeficiente é sempre produzido devido à implementação em software e hardware. Em software, um erro é causado pelo efeito do comprimento finito da palavra, enquanto em hardware, ocorre devido à impossibilidade de representação dos coeficientes com precisão absoluta. O resultado, em ambos os casos, é que os valores dos coeficientes diferem dos valores obtidos no processo de projeto. Tais erros causam desvios de frequência no sistema discreto no tempo projetado.
- O desvio de frequência depende do espaçamento entre os zeros da função de transferência do filtro FIR. O erro dos coeficientes afeta mais a característica espectral à medida que o espaçamento entre os zeros da função de transferência diminui. Esta propriedade é particularmente típica de filtros de alta ordem, porque seus zeros são próximos entre si. Erros pequenos na representação dos coeficientes podem causar grandes desvios de frequência.

Efeitos dos polos e zeros da função de transferência

- A figura ilustra a característica de frequência requerida e obtida para um filtro FIR. O efeito do comprimento finito da palavra na função de transferência do filtro é claramente marcado. Assumiu-se um filtro passa baixa de ordem 50 com frequência de corte normalizada de 0.25 Hz projetado usando-se janela de Hann.
- O desvio de frequência mostrado na figura é relativamente pequeno, embora seja significativo em alguns pontos. A mínima atenuação e a região da janela de transição do filtro resultante permanecem inalteradas, de modo que este desvio é aceitável. Entretanto, como nem sempre isso acontece, é necessário que se tenha cuidado no projeto de filtros de alta ordem pois os zeros da função de transferência se aproximam e afetam mais significativamente a característica espectral.

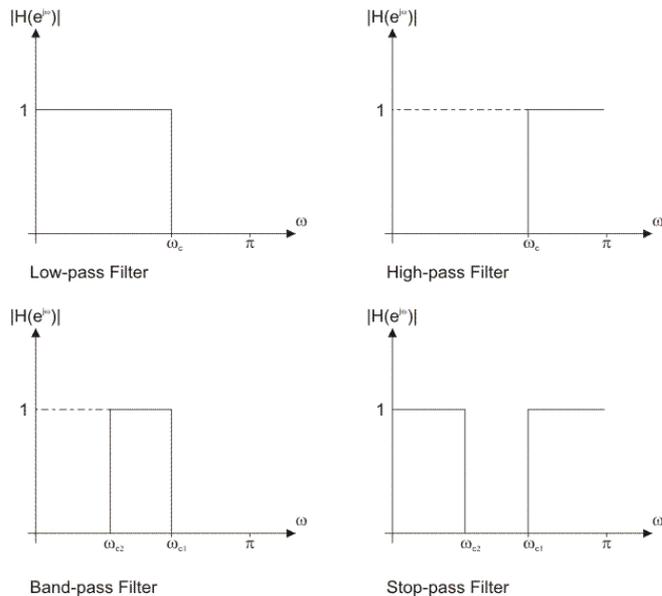


Aproximação por filtro ideal

- A resposta de frequência do filtro ideal é usada no projeto de filtros FIR usando funções de janelas.
 - O objetivo é computar as amostras do filtro ideal.
- Filtros FIR têm resposta impulsiva finita, o que significa que a amostragem de frequência do filtro ideal pode ser realizada num número finito de pontos.
 - Como a resposta de frequência do filtro ideal é finita, é fácil produzir erros de amostragem e o erro é menor quando a ordem do filtro aumenta.

Aproximação por filtro ideal

- A figura ilustra as funções de transferência de quatro filtros ideais padrão.
- A resposta em frequência do filtro ideal pode ser computada pela transformada inversa de Fourier. Para os quatro filtros padrão, as respostas são mostradas na tabela.



Type of filter	Frequency response $h_d[n]$
low-pass filter	$h_d[n] = \begin{cases} \frac{\sin[\omega_c(n-M)]}{\pi(n-M)}; & n \neq M \\ \frac{\omega_c}{\pi}; & n = M \end{cases}$
high-pass filter	$h_d[n] = \begin{cases} 1 - \frac{\omega_c}{\pi}; & n \neq M \\ -\frac{\sin(\omega_c(n-M))}{\pi(n-M)}; & n = M \end{cases}$
band-pass filter	$h_d[n] = \begin{cases} \frac{\sin(\omega_{c2}(n-M))}{\pi(n-M)} - \frac{\sin(\omega_{c1}(n-M))}{\pi(n-M)}; & n \neq M \\ \frac{\omega_{c2} - \omega_{c1}}{\pi}; & n = M \end{cases}$
band-stop filter	$h_d[n] = \begin{cases} \frac{\sin(\omega_{c1}(n-M))}{\pi(n-M)} - \frac{\sin(\omega_{c2}(n-M))}{\pi(n-M)}; & n \neq M \\ 1 - \frac{\omega_{c2} - \omega_{c1}}{\pi}; & n = M \end{cases}$

Aproximação por filtro ideal

- Na tabela (slide anterior), o valor da variável n varia entre 0 e N , onde N é a ordem do filtro. A constante M pode ser expressa como $M = N / 2$. Equivalentemente, N pode ser expressa como $N = 2M$.
- A constante M é um inteiro se a ordem do filtro é par, o que não acontece para filtros com ordem ímpar, Se M é um inteiro (filtro de ordem par), a resposta em frequência do filtro ideal é simétrica em torno da M -ésima amostra, o que é encontrado através da expressão mostrada na 2-2-1. Se M não é um inteiro, a resposta do filtro ideal continua simétrica, mas não em torno de uma amostra da resposta em frequência.
- Como a variável n varia entre 0 e N , a resposta do filtro ideal tem $N+1$ amostras.
- Se for necessário encontrar a resposta em frequência de um filtro ideal não padrão, a expressão seguinte, para a transformada inversa de Fourier deve ser usada:

$$h_d[n] = \frac{1}{\pi} \int_0^{\pi} e^{j\omega(n-M)} d\omega$$

Obs.: Filtros não padrão são raramente usados. Entretanto, se houver necessidade, a integral deve ser computada por métodos numéricos.

Projeto de filtro FIR usando funções de janela

- O processo pode ser dividido em vários passos:
 1. Definindo as especificações do filtro;
 2. Especificando as funções de janela, de acordo com as especificações do filtro;
 3. Computando a ordem do filtro requerida para um dado conjunto de especificações;
 4. Computando os coeficientes das funções de janela;
 5. Computando os coeficientes do filtro ideal de acordo com a ordem do filtro;
 6. Computando os coeficientes do filtro FIR de acordo com a função de janela obtida e os coeficientes de filtro ideal;
 7. Se o filtro resultante tiver uma região de transição muito larga ou muito estreita, é necessário mudar a ordem do filtro (aumentando-a ou diminuindo-a) de acordo com a necessidade, e fazer iterações dos passos 4, 5 e 6 tantas vezes quantas forem necessárias.
- O objetivo final de definir as especificações do filtro é encontrar as frequências normalizadas (ω_c , ω_{c1} , ω_{c2}), a largura da transição e a atenuação da banda cortada. A função da janela e a ordem do filtro são ambas especificadas de acordo com estes parâmetros.
- A função janela selecionada deve satisfazer as especificações dadas.
- Quando a função da janela é conhecida, podemos computar a ordem do filtro requerida para um dado conjunto de especificações. Uma das técnicas para este cálculo será explicada mais adiante.

Projeto de filtro FIR usando funções de janela

- Após estimar os coeficientes da função da janela, é necessário encontrar as amostras de frequência do filtro ideal.
 - O objetivo final deste passo é obter os coeficientes $h_d[n]$. Duas sequências $w[n]$ e $h_d[n]$ têm o mesmo número de elementos.
- O próximo passo é computar a resposta em frequência do filtro projetado $h[n]$ usando a seguinte expressão:

$$h[n] = w[n] \cdot h_d[n]$$

- Mais tarde, a função de transferência do filtro projetado será obtida pela transformada da resposta impulsiva via transformada de Fourier:

$$H(e^{j\omega}) = \sum_{n=0}^N h[n] \cdot e^{-jn\omega}$$

- Ou transformada Z:

$$H(z) = \sum_{n=0}^N h[n] z^{-n}$$

- Se a região de transição do filtro projetado for mais larga do que o necessário, é preciso aumentar a ordem do filtro, reestimar os coeficientes da função de janela e as amostras de frequência do filtro ideal, multiplica-las a fim de se obter a resposta em frequência do filtro projetado e reestimar a função de transferência. Se a transição for mais estreita que o necessário, a ordem do filtro pode ser diminuída. Também é necessário reestimar os coeficientes de frequência, após isso.

Realização do filtro FIR

- A função de transferência do filtro FIR pode ser expressa como:

$$H(z) = \frac{Y(z)}{X(z)} = \sum_{n=0}^{N-1} h[n] \cdot z^{-n}$$

- A resposta em frequência no domínio do tempo é de maior interesse para a realização do filtro FIR (ambas, hardware e software). A função de transferência pode ser obtida pela transformada Z da resposta em frequência. As amostras da saída do filtro podem ser computadas usando a seguinte expressão:

$$y[n] = \sum_{k=0}^{N-1} h[k] \cdot x[n-k]$$

– onde:

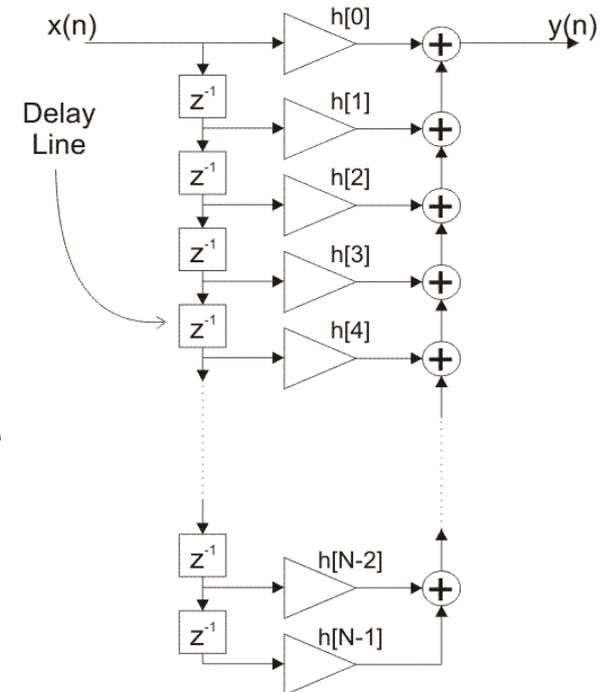
- **x[k]** são as amostras da entrada do filtro FIR;
- **h[k]** são os coeficientes da resposta em frequência do filtro FIR; e
- **y[n]** são as amostras na saída do filtro FIR.

Realização tipo *direta*

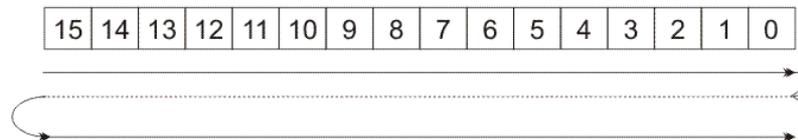
- A realização direta de um filtro FIR (direta também é conhecida como filtro transversal) é baseada na implementação direta da expressão:

$$y[n] = \sum_{k=0}^{N-1} h[k] \cdot x[n-k]$$

- Para a estrutura da realização direta, as constantes de multiplicação são os coeficientes da função de transferência, i.e. os coeficientes da resposta em frequência do filtro FIR.
- Para a realização em software, é necessário prover um buffer para o mínimo de N amostras, onde N é o número de coeficientes do filtro FIR. Para simplicidade e velocidade de processamento, o buffer mais usado é chamado de buffer circular com comprimento expresso como 2^k . O valor da constante k é o mínimo valor para o qual $N \leq 2^k$ é válido. Assim: $k = \lceil 1 + \log_2 N \rceil$



- Onde o operador $\lfloor \rfloor$ representa arredondamento para baixo para um valor menor.



➔ Buffer circular buffer de comprimento $16 = 2^4$

Realização tipo *direta*

- O algoritmo usado para a realização em software de um filtro FIR consiste de:
 - Ler as amostras de um sinal sendo filtrado;
 - Armazenar uma nova amostra na primeira localização disponível; e
 - Realizar a convolução com os coeficientes do filtro (coeficientes da resposta em frequência), resultando na amostra da saída do filtro FIR.
- Como o buffer tem o tamanho de 16 bits, o endereçamento circular é realizado usando o modo de endereçamento 16:

$$(5+1)_{\text{mode } 16}=6$$

$$(15+1)_{\text{mode } 16}=0$$

Exemplo

- Seja um filtro FIR de quinta ordem, projete-o usando a realização direta com buffer circular. O comprimento do buffer precisa ser 2^k .

– O comprimento do buffer é obtido da seguinte maneira:

$$k = \lceil 1 + \log_{(2)} N \rceil = \lceil 1 + \log_{(2)} 5 \rceil = \lceil 3.322 \rceil = 3$$

- Isso significa que o mínimo comprimento do buffer circular é $2^3 = 8$.
- O conteúdo do buffer, depois do recebimento de 10 amostras, é mostrado na tabela. As amostras de entrada são denotadas por $x[n]$, enquanto as células sombreadas denotam as localizações do buffer que foram alteradas.

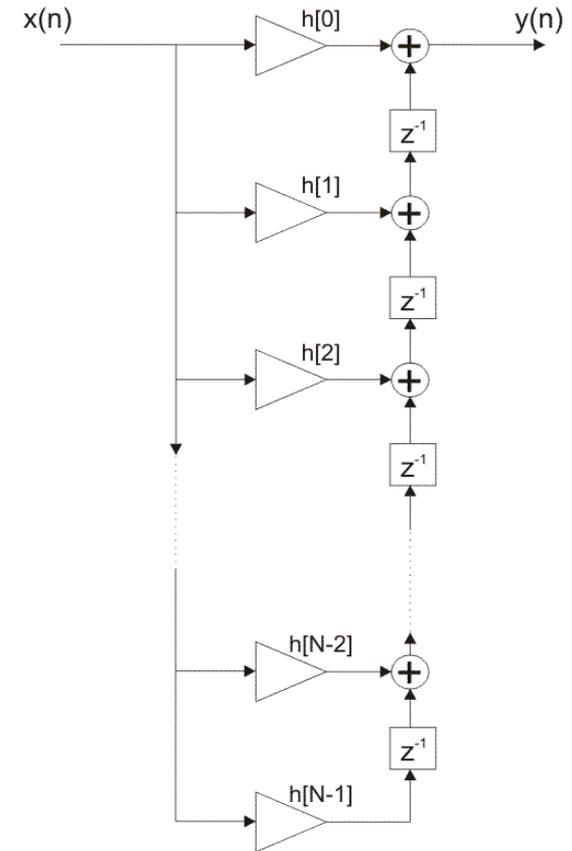
PASSO	ADDRESS 7	ADDR. 6	ADDR. 5	ADDR. 4	ADDR. 3	ADDR. 2	ADDR. 1	ADDR. 0
0								
1								x[0]
2							x[1]	x[0]
3						x[2]	x[1]	x[0]
4					x[3]	x[2]	x[1]	x[0]
5				x[4]	x[3]	x[2]	x[1]	x[0]
6			x[5]	x[4]	x[3]	x[2]	x[1]	x[0]
7		x[6]	x[5]	x[4]	x[3]	x[2]	x[1]	x[0]
8	x[7]	x[6]	x[5]	x[4]	x[3]	x[2]	x[1]	x[0]
9	x[7]	x[6]	x[5]	x[4]	x[3]	x[2]	x[1]	x[8]
10	x[7]	x[6]	x[5]	x[4]	x[3]	x[2]	x[9]	x[8]

Para a realização em software, a filtragem das amostras de entrada é realizada segundo a expressão:

$$y[n] = \sum_{k=0}^{N-1} h[k] \cdot x[n - k]$$

Realização direta *transposta*

- É similar à realização direta em muitos aspectos. Em termos de implementação em software, a realização transposta direta também deve ter um buffer com comprimento mínimo N , onde N é o número de coeficientes do filtro FIR.
- Não existem diferenças significativas entre as realizações direta e direta transposta. Ambas as estruturas têm o mesmo número de elementos de atraso, o mesmo número de multiplicadores e os mesmos coeficientes para realizar a multiplicação para cima.



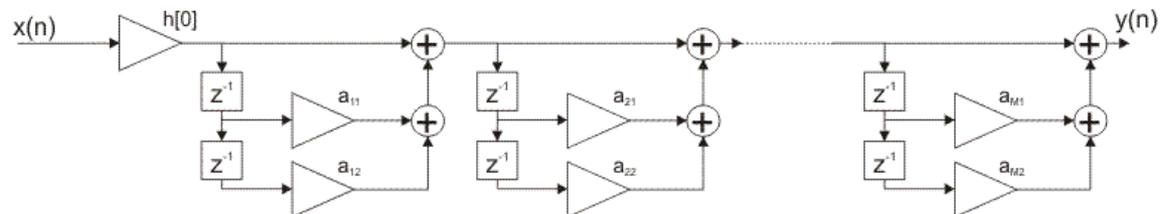
Realização *em cascata*

- Sendo bastante conveniente por causa de sua estrutura modular, é comumente usada para realização em hardware de filtro FIR. Neste caso, o filtro é dividido em várias seções de ordem inferior, sendo as seções de segunda ordem as mais usadas. Seções individuais são em geral realizadas na forma direta, embora também possa ser usada a direta transposta. A realização em cascata é normalmente empregada para filtros de ordens mais altas.
- A função de transferência de uma realização em cascata se apresenta como:

$$H(z) = h[0] \cdot \prod_{k=1}^M (1 + a_{k1} \cdot z^{-1} + a_{k2} \cdot z^{-2})$$

– onde:

- M é o número de seções; e
 - a_{k1} , a_{k2} são os coeficientes de multiplicação da seção k.
- A figura ilustra o diagrama de blocos que descreve esta realização. O número de multiplicadores, somadores e atrasos é o mesmo do da realização direta. A maior vantagem da cascata é sua modularidade, o que a torna conveniente para implementação em hardware. A cascata de seções de segunda ordem é importante para a realização de filtros de ordem arbitrária.



Realização *otimizada*

- A realização otimizada usa menos multiplicadores, mas estes são mais rigorosos. Esta realização é mais usada para implementação em software porque a redução do número de multiplicadores aumenta o processo de convolução (processo de filtragem das amostras).
- A realização otimizada utiliza a simetria dos coeficientes na resposta em frequência. Existem também filtros FIR anti-simétricos mas são mais raros e, para eles, a realização otimizada também por ser usada.
- A simetria dos coeficientes pode ser expressa pela equação:

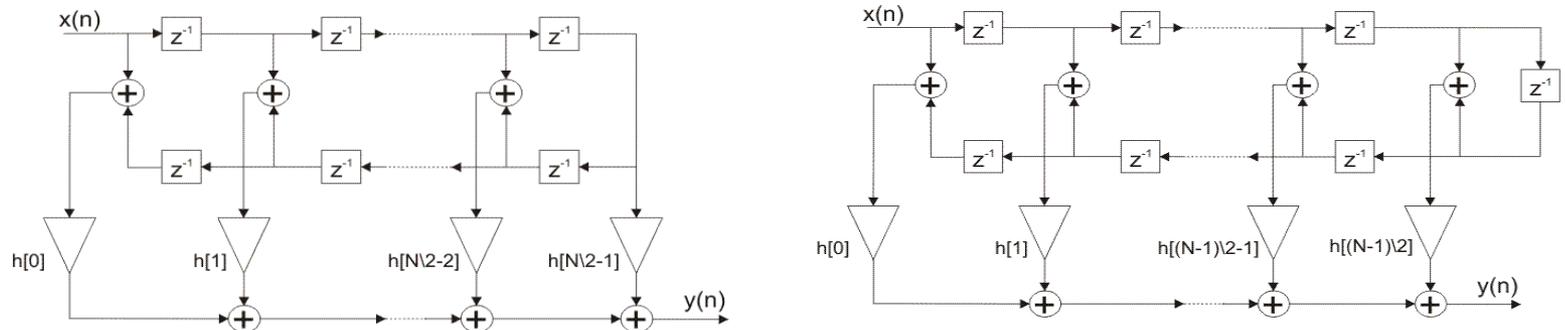
$$h[k] = h[N - 1 - k]$$

- A simetria torna possível que se expresse a função de transferência como:

$$H(z) = h[0] \cdot (1 + z^{-N+1}) + h[1] \cdot (z^{-1} + z^{-N+2}) + \dots + h[N \setminus 2 - 1] \cdot (z^{-N/2+1} + z^{-N/2}) \quad \text{for even } N$$

$$H(z) = h[0] \cdot (1 + z^{-N+1}) + h[1] \cdot (z^{-1} + z^{-N+2}) + \dots + h[(N-1) \setminus 2] \cdot z^{-(N-1)/2} \quad \text{for odd } N$$

- A figura ilustra o diagrama de bloco da realização otimizada para N ímpar (esquerda), e N par (direita).



Funções Janela

- O método da janela é o mais usado no projeto de filtros FIR, sendo popular por ser simples.
- Uma janela é um conjunto finito que consiste de coeficientes selecionados que satisfazem a certos requisitos desejáveis.
- Ao se projetar filtros digitais FIR usando funções de janela é necessário que se especifique:
 - Uma função para janela a ser usada; e
 - A ordem do filtro de acordo com as especificações requeridas (seletividade e atenuação na banda rejeitada).
- Estes dois requisitos estão inter-relacionadas e cada função estabelece um tipo de compromisso entre os dois:
- Quanto maior a seletividade, i.e. quanto mais estreita a região de transição; e
- Quanto mais alta a supressão do espectro indesejado, i.e. quanto maior a atenuação da banda rejeitada.
- A Tabela contém as funções de janela, que serão mencionadas no documento em anexo a esta aula, e compara brevemente sua seletividade e atenuação da banda rejeitada.

FUNÇÃO JANELA	COMPRIMENTO NORMALIZADO DO LÓBULO PRINCIPAL PARA N=20	REGIÃO DE TRANSIÇÃO PARA N=20	MÍNIMA ATENUAÇÃO DA BANDA REJEITADA PARA FUNÇÃO DA JANELA	MÍNIMA ATENUAÇÃO PARA A BANDA REJEITADA NO PROJETO DO FILTRO
Retangular	0.1n	0.041n	13 dB	21 dB
Triangular (Bartlett)	0.2n	0.11n	26 dB	26 dB
Hann	0.21n	0.12n	31 dB	44 dB
Bartlett-Hanning	0.21n	0.13n	36 dB	39 dB
Hamming	0.23n	0.14n	41 dB	53 dB
Bohman	0.31n	0.2n	46 dB	51 dB
Blackman	0.32n	0.2n	58 dB	75 dB
Blackman-Harris	0.43n	0.32n	91 dB	109 dB