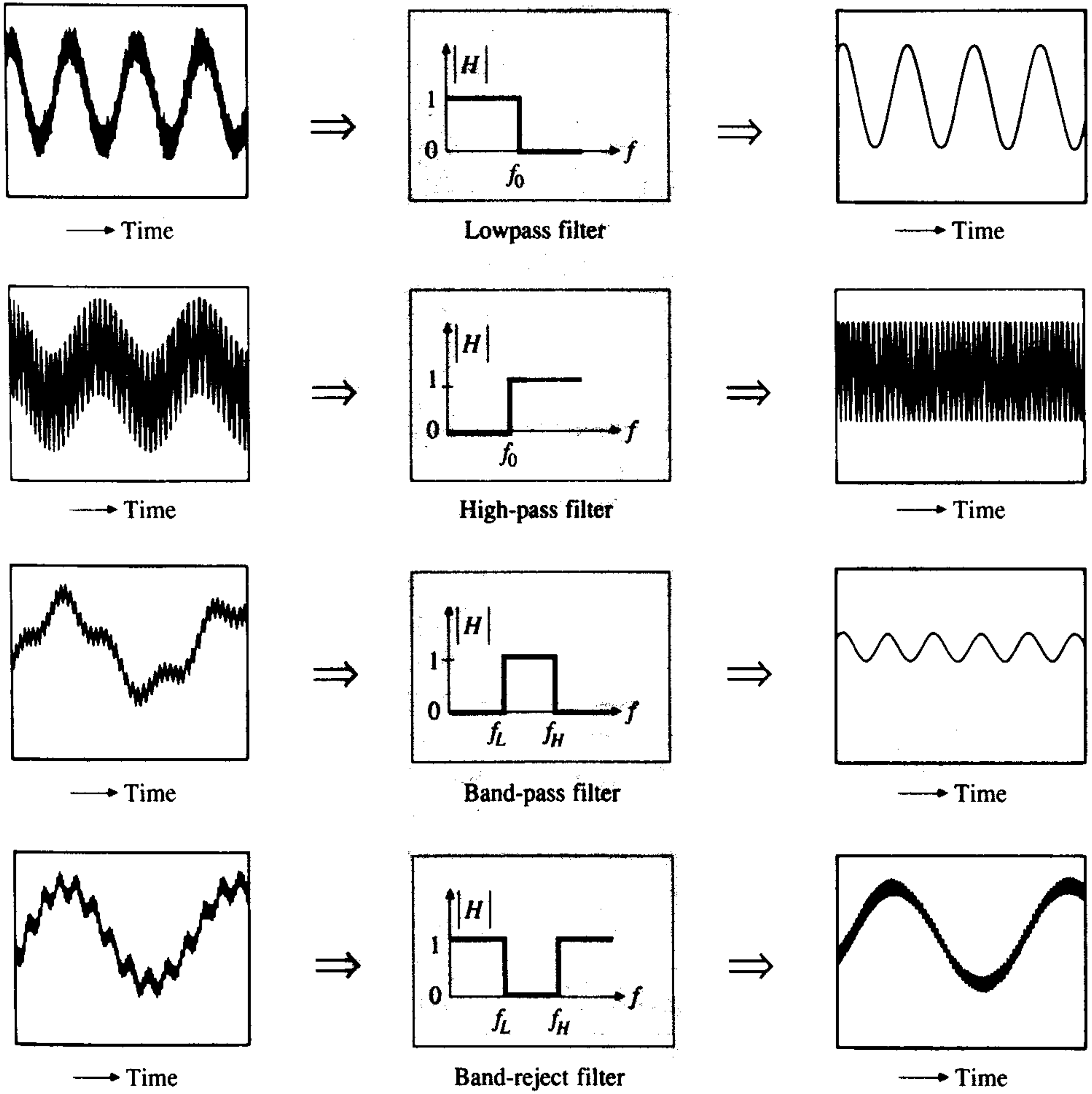
Digital Filters

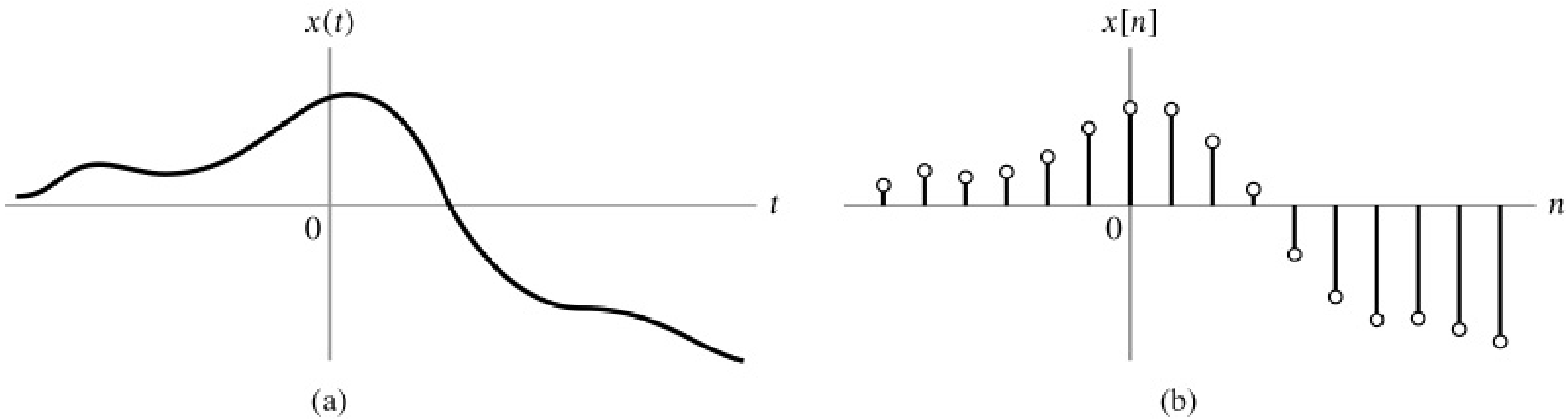
Hsiao-Lung Chan, Ph.D.
Department of Electrical Engineering,
Chang-Gung University, Taiwan
chanhl@mail.cgu.edu.tw

Ideal filters

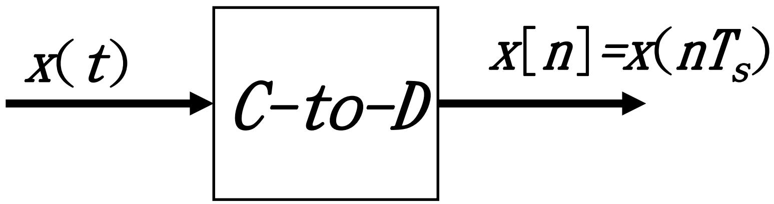


S. Franco, "Design with Operational Amplifiers and Analog Integrated Circuits", Second Edition, 1998.

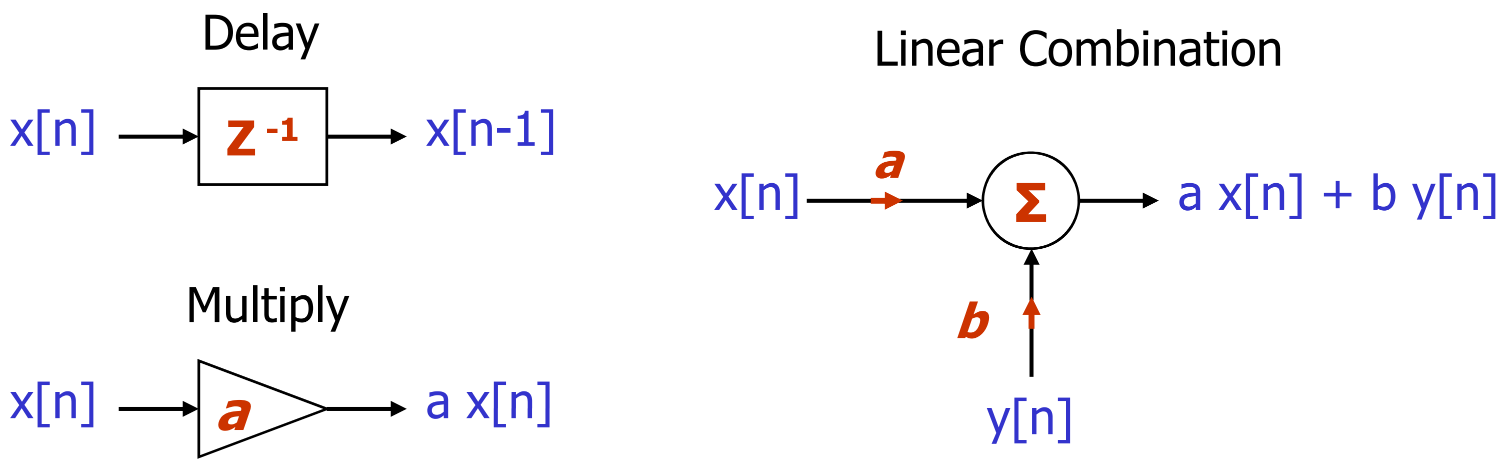
Continuous- and discrete-time signals



- Sampling rate (f_s)
  - $f_s = 1/T_s$ number of samples per second
  - T_s = 125 microsecsec (10^{-3} sec) → f_s = 8000 samples/sec



Z transform



	Digital signal	z transform	Analog signal
Input signal	x[n]	X(z)	x(t)
Delay one sample	x[n-1]	Z^{-1} X(z)	x(t-T)
Multiply	a x[n]	a X(z)	a x(t)
Linear combination	a x[n] + b y[n]	a X(z) + b Y(z)	a x(t) + b y(t)

Z-transform of a signal

- Any signal has a z-Transform:

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

n	$n < -1$	-1	0	1	2	3	4	5	$n > 5$
$x[n]$	0	0	2	4	6	4	2	0	0

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

Example 1: Perform the running average of last six digital sample

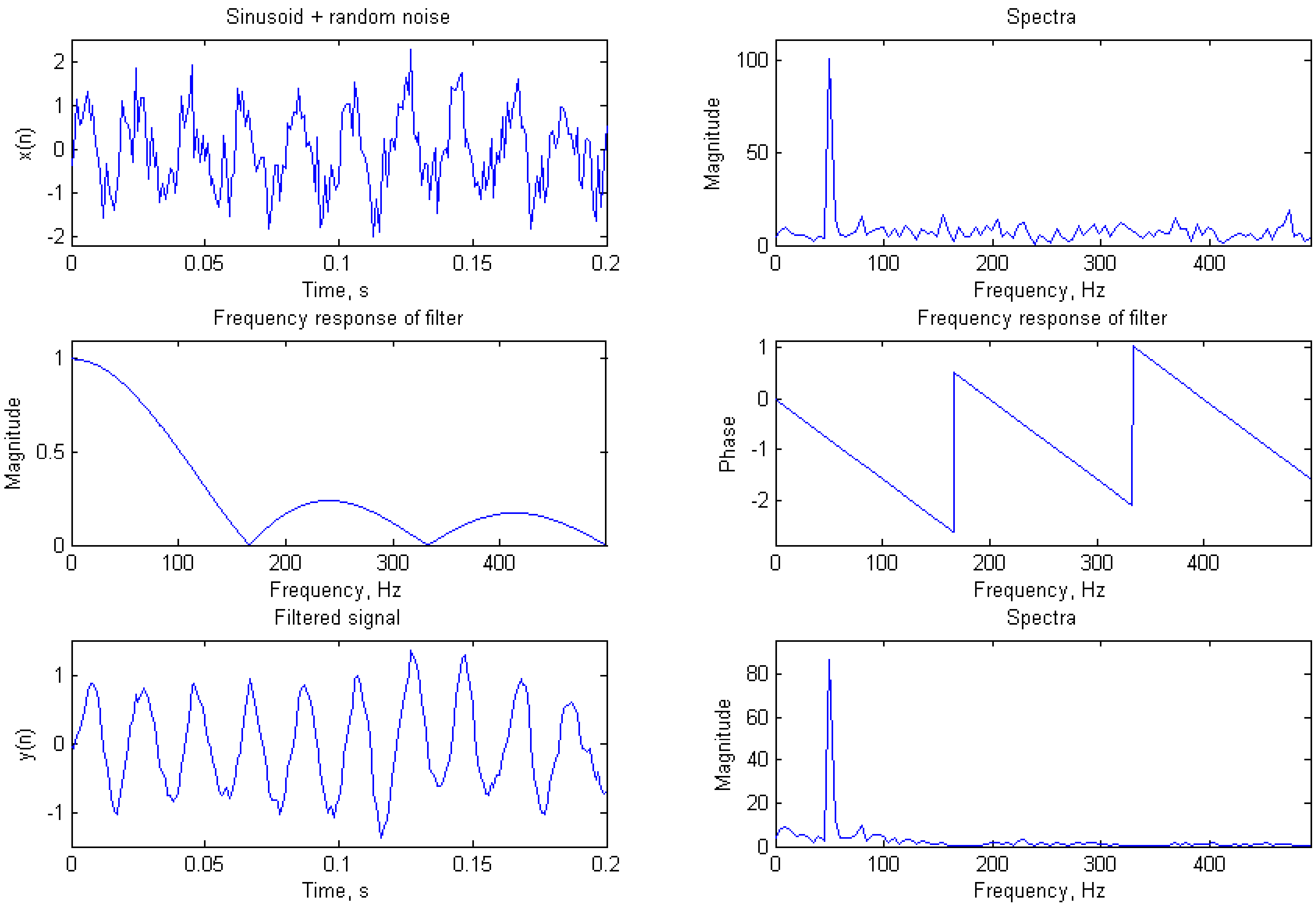
$$y[n] = \frac{x[n] + x[n-1] + x[n-2] + x[n-3] + x[n-4] + x[n-5]}{6}$$

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

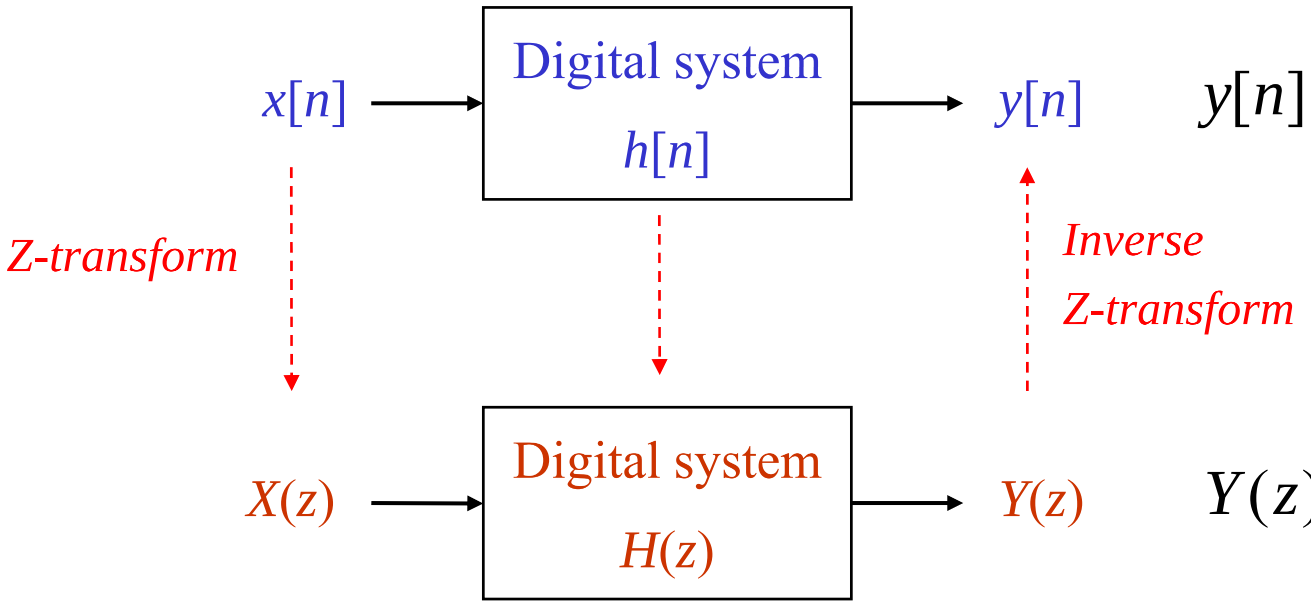
Transfer function

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1 + z^{-1} + z^{-2} + z^{-3} + z^{-4} + z^{-5}}{6}$$

```
fs=1000;
t=0:1/fs:0.2;
x = sin(2*pi*50*t) +
    0.5*randn(size(t));

Xf=fft(x);
resolution=fs/length(Xf);
f=(0:length(Xf)-1)*resolution;
Xf_magnitude = abs(Xf);
subplot(3,2,2)
index=1:length(Xf_magnitude)/2;
plot(f(index),Xf_magnitude(index))

b=[1/6 1/6 1/6 1/6 1/6 1/6];
a=[1];
NFFT=1024;
[h,f] = freqz(b,a,NFFT);
f=f/pi*fs/2;
h_magnitude=abs(h);
h_phase=phase(h);
subplot(3,2,3)
plot(f,h_magnitude);
subplot(3,2,4)
plot(f,h_phase);

y=filter(b,a,x);
t=(0:length(y)-1)/fs;

Yf=fft(y);
resolution=fs/length(Yf);
f=(0:length(Yf)-1)*resolution;
Yf_magnitude = abs(Yf);

subplot(3,2,6)
index=1:length(Yf_magnitude)/2;
plot(f(index),Yf_magnitude(index))
```


Sinusoid + random noise
x(n)
Time, s
Spectra
Magnitude
Frequency, Hz
Frequency response of filter
Magnitude
Frequency, Hz
Frequency response of filter
Phase
Frequency, Hz
Filtered signal
y(n)
Time, s
Spectra
Magnitude
Frequency, Hz

Transfer function



impulse response

convolution

$$y[n] = x[n] * h[n]$$

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

transfer function

Example

- Input is x[n], find y[n]

$$x[n] = \delta[n-1] - \delta[n-2] + \delta[n-3] - \delta[n-4]$$

$$\text{and} \quad h[n] = \delta[n] + 2\delta[n-1] + 3\delta[n-2] + 4\delta[n-3]$$

$$\Big\Downarrow \ \text{Z transform}$$

$$X(z) = 0 + 1z^{-1} - 1z^{-2} + 1z^{-3} - 1z^{-4}$$

$$\text{and} \quad H(z) = 1 + 2z^{-1} + 3z^{-2} + 4z^{-3}$$

Method 1: Multiply the z-TRANSFORMS

$$
\begin{aligned}
Y(z) &= H(z)X(z) \\
&= (1 + 2z^{-1} + 3z^{-2} + 4z^{-3})(z^{-1} - z^{-2} + z^{-3} - z^{-4}) \\
&= z^{-1} + (-1 + 2)z^{-2} + (1 - 2 + 3)z^{-3} + (-1 + 2 - 3 + 4)z^{-4} \\
&\quad + (-2 + 3 - 4)z^{-5} + (-3 + 4)z^{-6} + (-4)z^{-7} \\
&= z^{-1} + z^{-2} + 2z^{-3} + 2z^{-4} - 3z^{-5} + z^{-6} - 4z^{-7}
\end{aligned}
$$

***y*[*n*] = ?**

Method 2: Convolution

x[n], X(z)	0	+1	-1	+1	-1			
h[n], H(z)	1	2	3	4				
	0	+1	-1	+1	-1			
		0	+2	-2	+2	-2		
			0	+3	-3	+3	-3	
				0	+4	-4	+4	-4
y[n], Y(z)	0	+1	+1	+2	+2	-3	+1	-4

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

conv([1 2 3 4],[0 1 -1 1 -1])

filter([1 2 3 4],[1],[0 1 -1 1 -1])

Zeros of H(z)

- Find z, where H(z)=0

$$H(z) = 1 - 2z^{-1} + 2z^{-2} - z^{-3}$$

$$H(z) = (1 - z^{-1})(1 - z^{-1} + z^{-2})$$

$$\text{Roots}: z = 1, \tfrac{1}{2} \pm j\frac{\sqrt{3}}{2}$$

$$\boxed{e^{\pm j\pi/3}}$$

Poles of H(z)

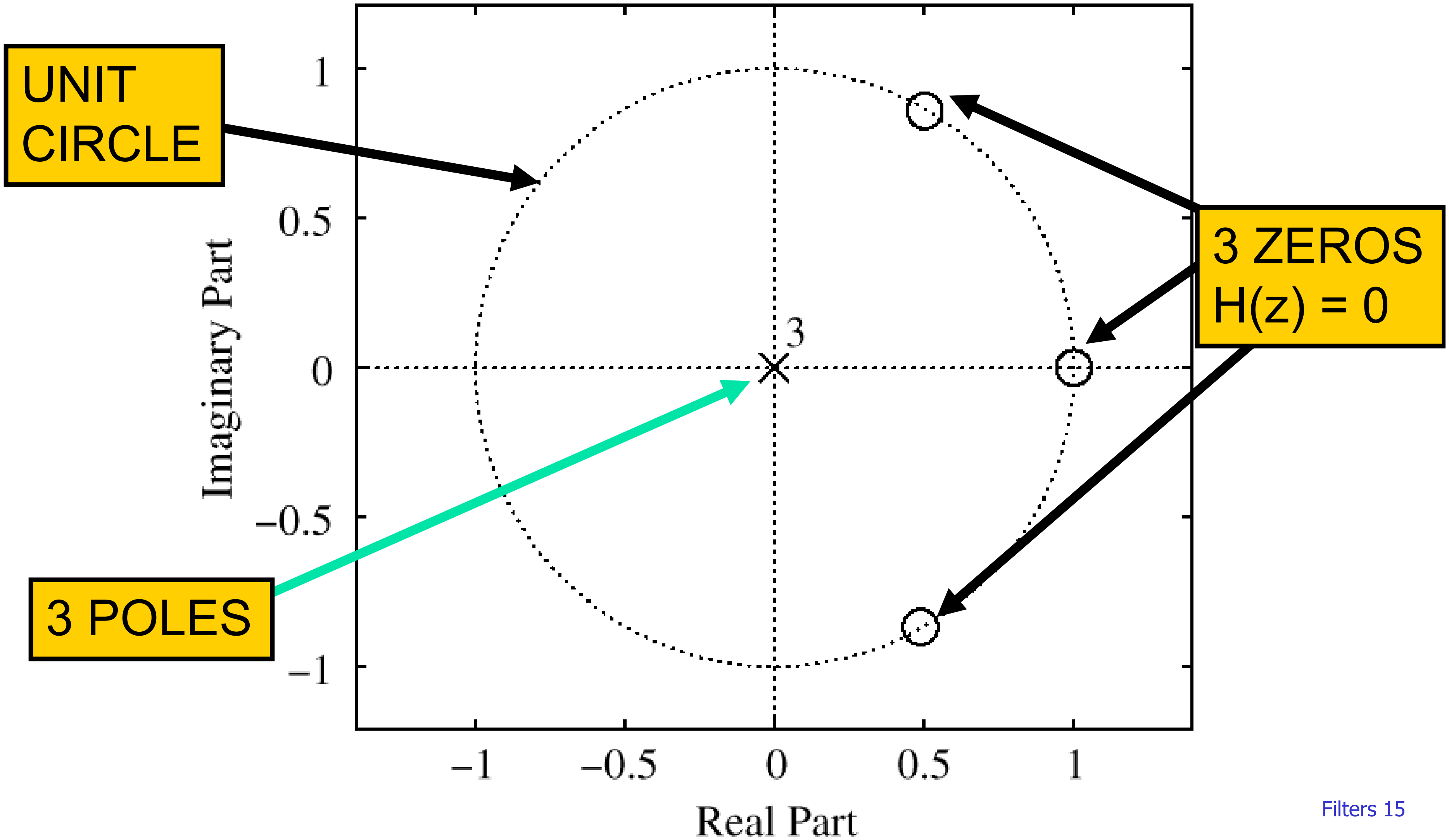
- Find z, where $H(z) \to \infty$

$$H(z) = 1 - 2z^{-1} + 2z^{-2} - z^{-3}$$

$$H(z) = \frac{z^3 - 2z^2 + 2z \quad -1}{z^3}$$

$$\text{Three Poles at} : z = 0$$

Plot poles and zeros in z-domain



Exercise: Poles and zeros of H(z)

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1 + z^{-1} + z^{-2} + z^{-3} + z^{-4} + z^{-5}}{6}$$

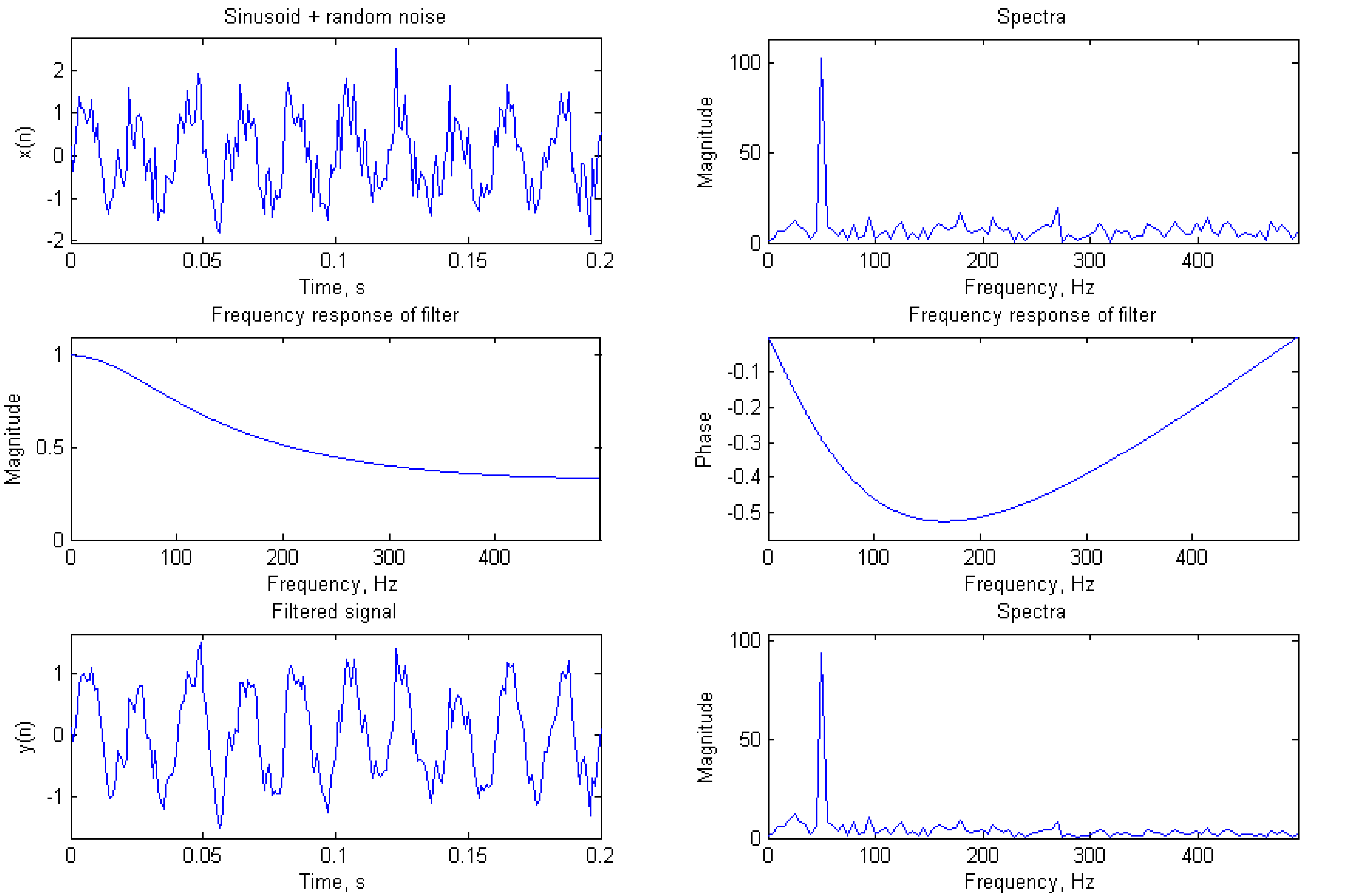
Example 2: Perform the average of current data and last filter output

$$y(n) = \frac{y(n-1) + x(n)}{2}$$

$$Y(z) = \frac{z^{-1}Y(z) + X(z)}{2}$$

Transfer function

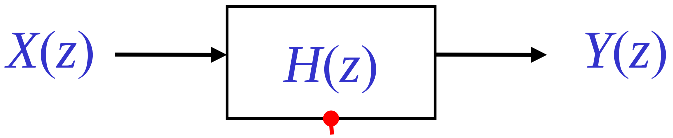
$$H(z) = \frac{Y(z)}{X(z)} = \frac{1}{2 - z^{-1}}$$


Sinusoid + random noise
x(n)
Time, s
Spectra
Magnitude
Frequency, Hz
Frequency response of filter
Magnitude
Frequency, Hz
Frequency response of filter
Phase
Frequency, Hz
Filtered signal
y(n)
Time, s
Spectra
Magnitude
Frequency, Hz

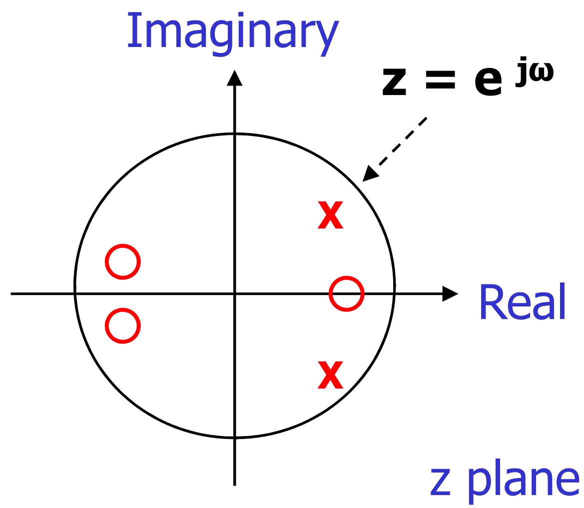
Exercise: Poles and zeros of H(z)

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1}{2 - z^{-1}}$$

Frequency response of transfer function



If the poles of H(z) are located with unit circle



Frequency Response of H(z)

$$H(j\omega) = H(z)\Big|_{z=e^{j\omega}}$$

Frequency response (cont.)

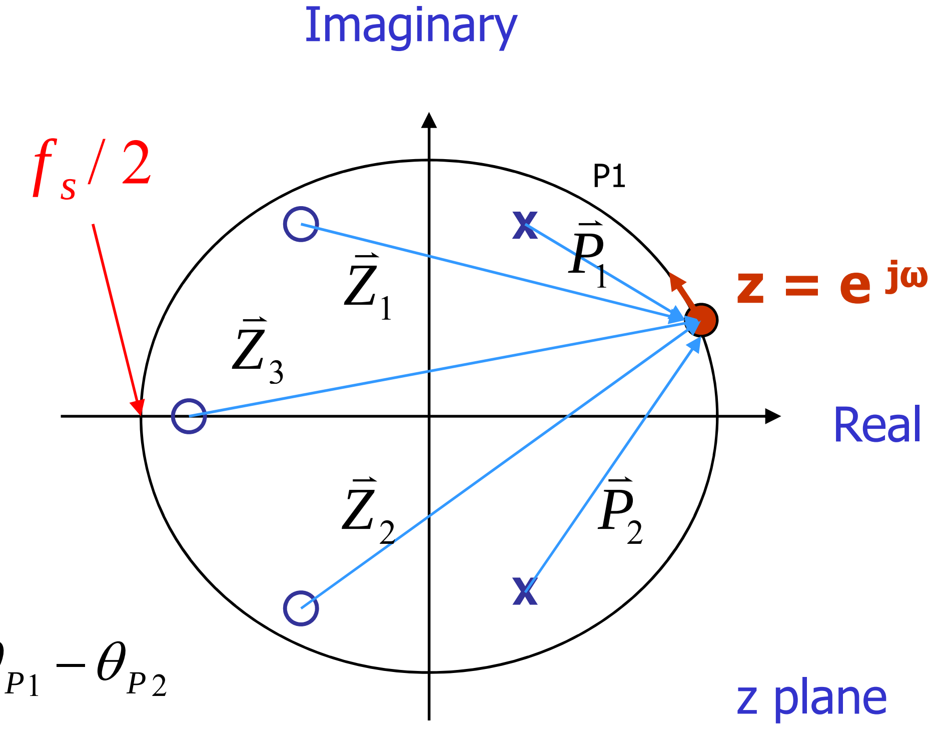
$$H(j\omega) = \frac{\vec{Z}_1 \cdot \vec{Z}_2 \cdot \vec{Z}_3}{\vec{P}_1 \cdot \vec{P}_2}$$

Magnitude

$$|H(j\omega)| = \frac{|\vec{Z}_1| \cdot |\vec{Z}_2| \cdot |\vec{Z}_3|}{|\vec{P}_1| \cdot |\vec{P}_2|}$$

Phase

$$\angle H(j\omega) = \theta_{Z1} + \theta_{Z2} + \theta_{Z2} - \theta_{P1} - \theta_{P2}$$



Exercise: Frequency response of H(z)

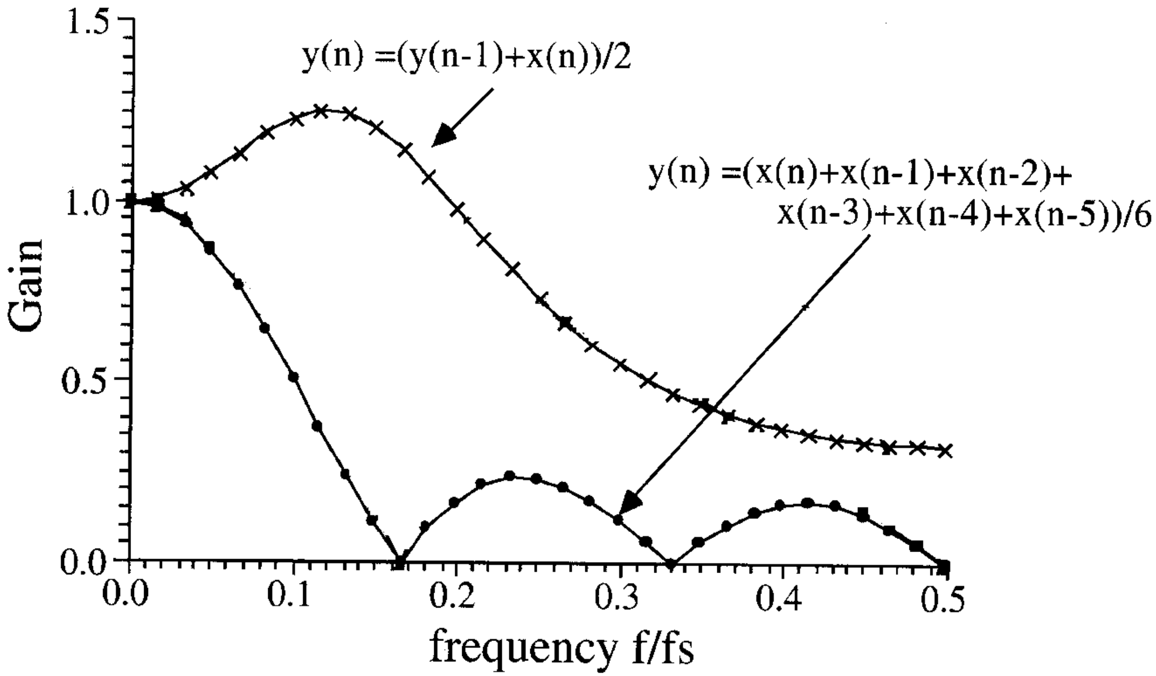
$$H(z) = \frac{Y(z)}{X(z)} = \frac{1 + z^{-1} + z^{-2} + z^{-3} + z^{-4} + z^{-5}}{6}$$

Exercise: Frequency response of H(z)

$$H(z) = \frac{Y(z)}{X(z)} = \frac{1}{2 - z^{-1}}$$

Frequency response of example 1 and 2

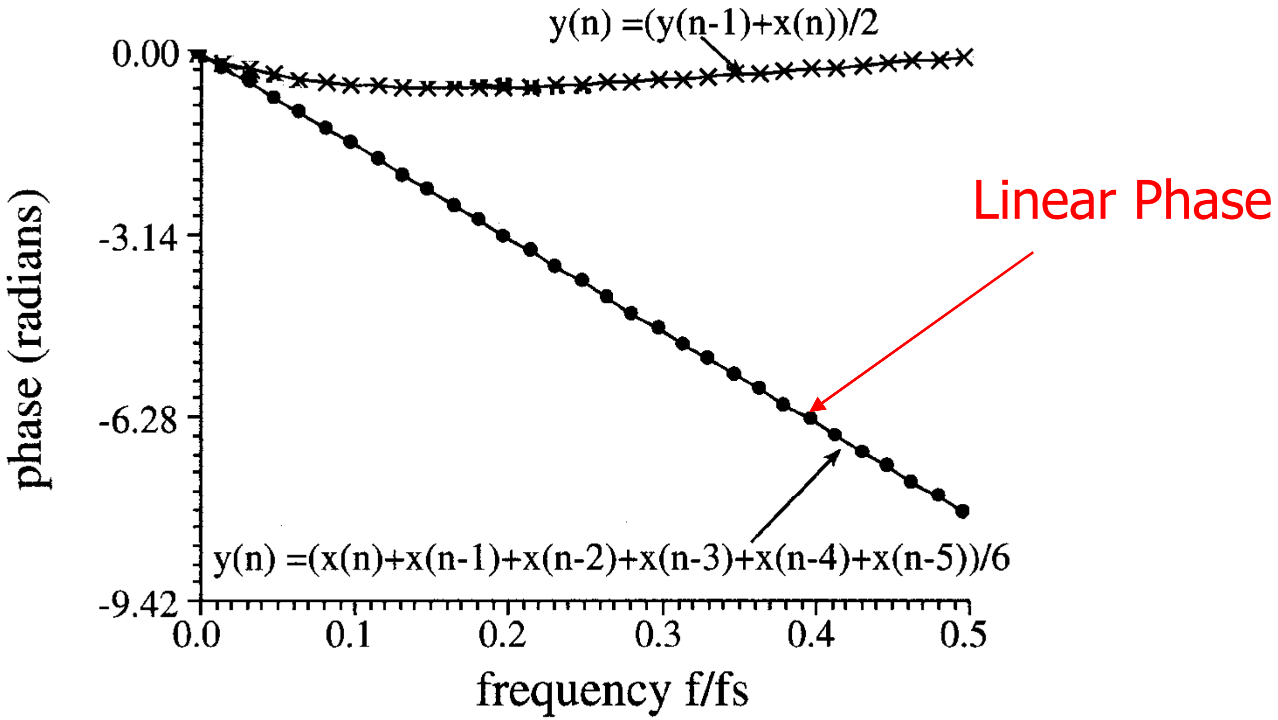
Magnitude



From Jonathan W. Valvano, Embedded Microcomputer Systems, real time interfacing, Brooks/Cole, 2000.

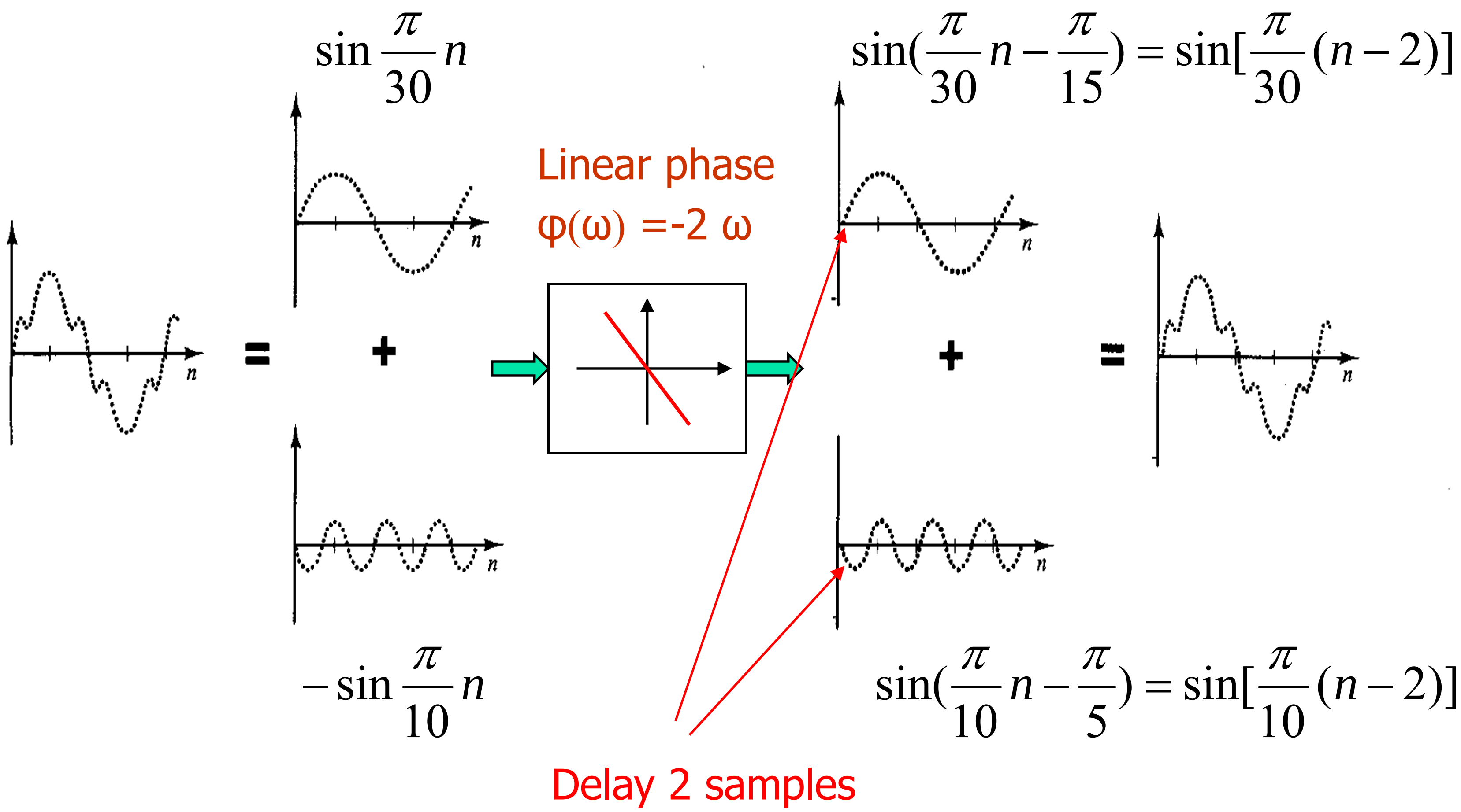
Frequency Response of Example 1 and 2

Phase



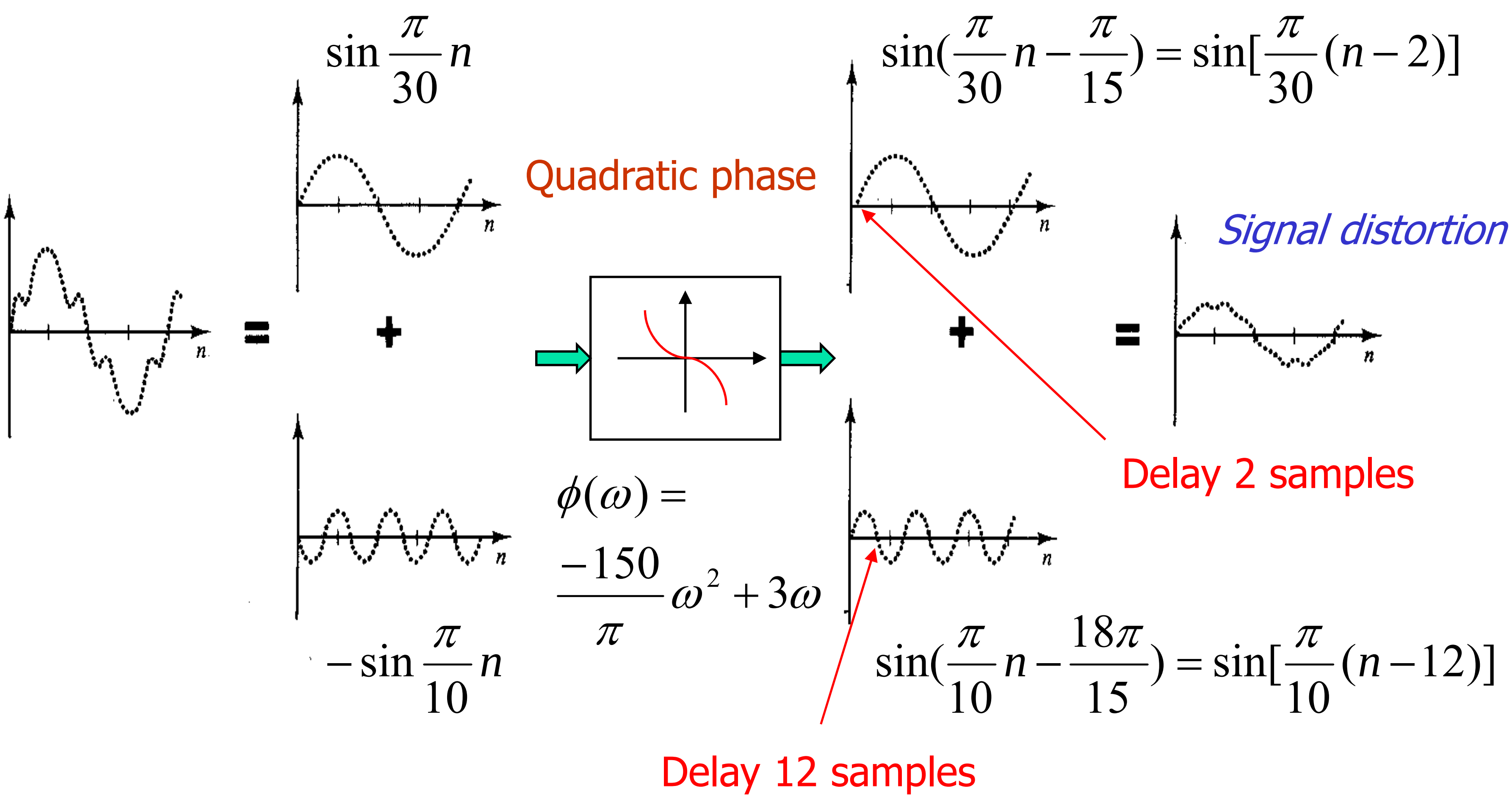
From Jonathan W. Valvano, Embedded Microcomputer Systems, real time interfacing, Brooks/Cole, 2000.

Linear phase



Modified from L.Ludeman, Fundamentals of digital signal processing,Harper & Row, 1986.

Nonlinear phase



Modified from L.Ludeman, Fundamentals of digital signal processing,Harper & Row, 1986.

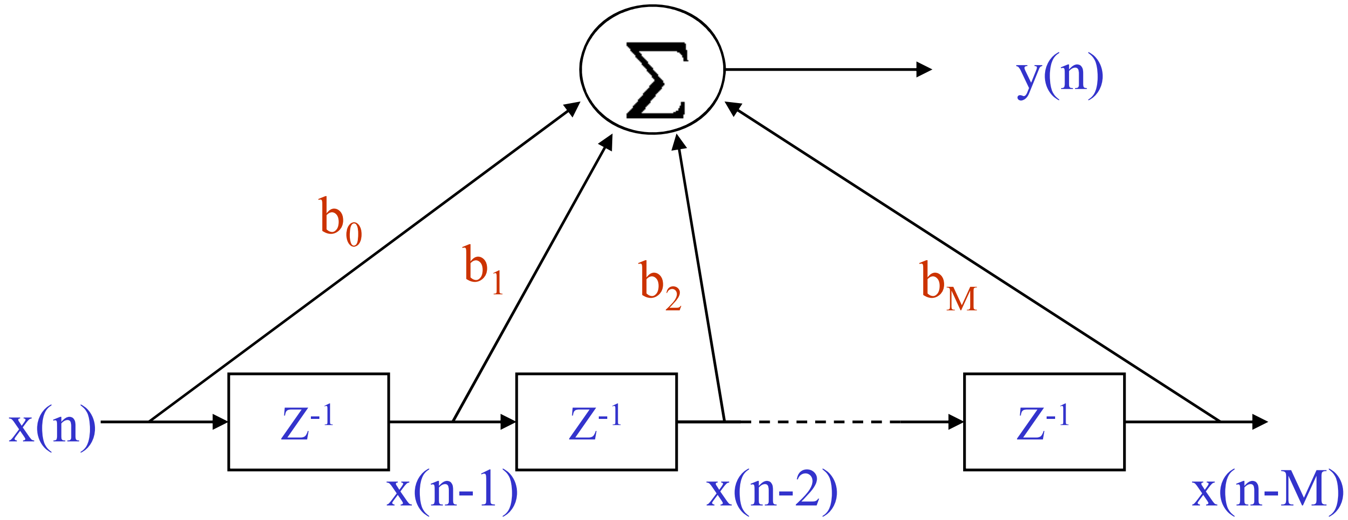
Group delay

$$\tau = -\frac{d\phi(\omega)}{d\omega}$$

- Linear phase yield constant group delay

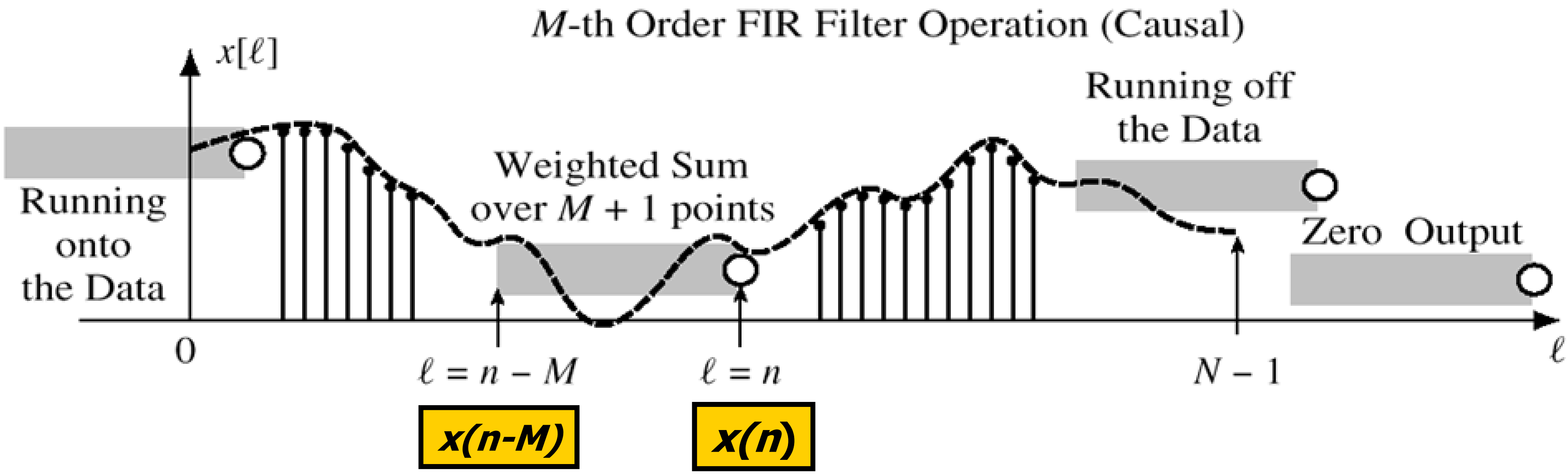
Finite impulse response (FIR) filter

$$y(n)=\sum_{k=0}^{M} b_k x(n-k) \qquad H(z)=\frac{Y(z)}{X(z)}=b_0+b_1 z^{-1}+b_2 z^{-2}\cdots+b_N z^{-M}$$

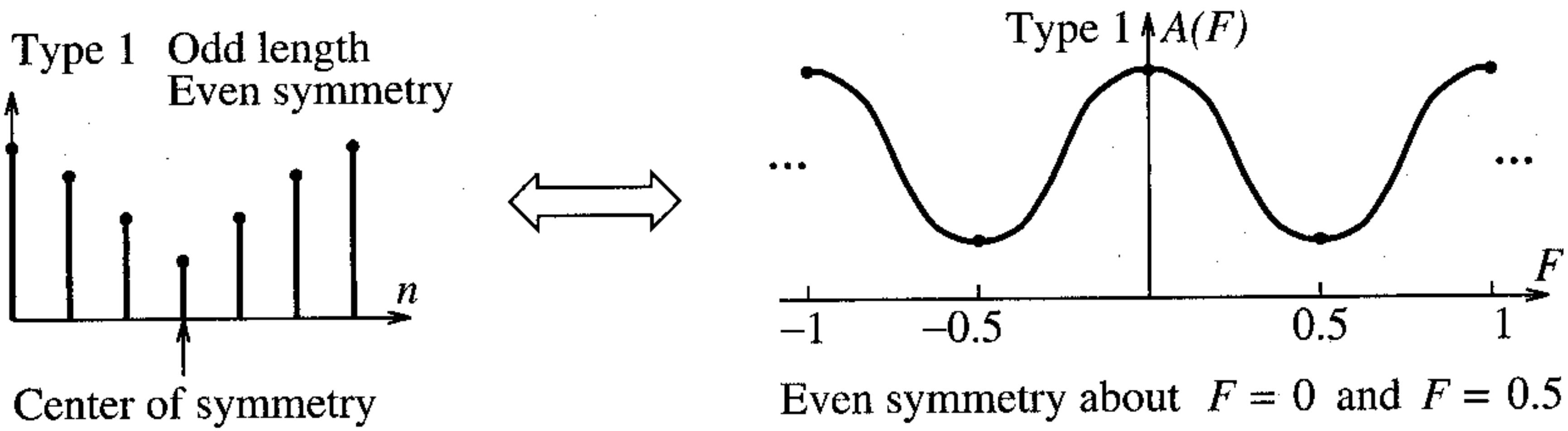


- FIR posses linear-phase property if filter coefficients are symmetry or anti-symmetry around the center

FIR: sliding a length-L window over x(n)



FIR filter: odd length, even symmetry



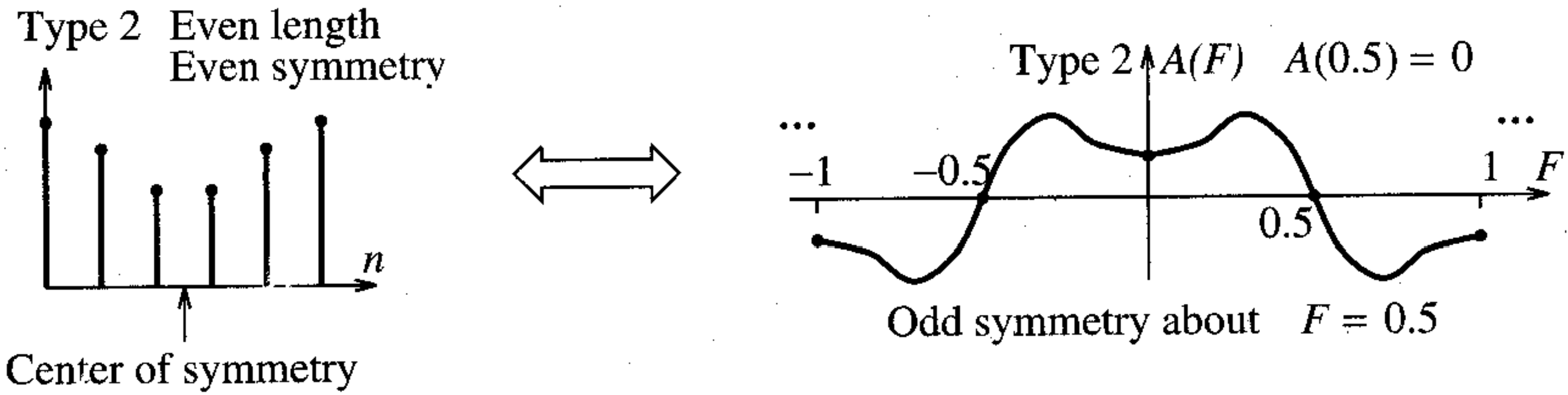
Features of a type 1 symmetric sequence

Linear phase = 2πMF

$$H_1(F) = \sum_{n=0}^{N-1} h[n] e^{-j2\pi MF} = \underbrace{\left[h[M] + 2\sum_{k=0}^{M-1} h[k]\cos[(M-k)2\pi F] \right]}_{A(F)} e^{-j2\pi MF}$$

From A.Ambardar, Analog and Digital Signal Processing, 2nd Edition, Brook/Cole, 1999.

FIR filter: even length, even symmetry



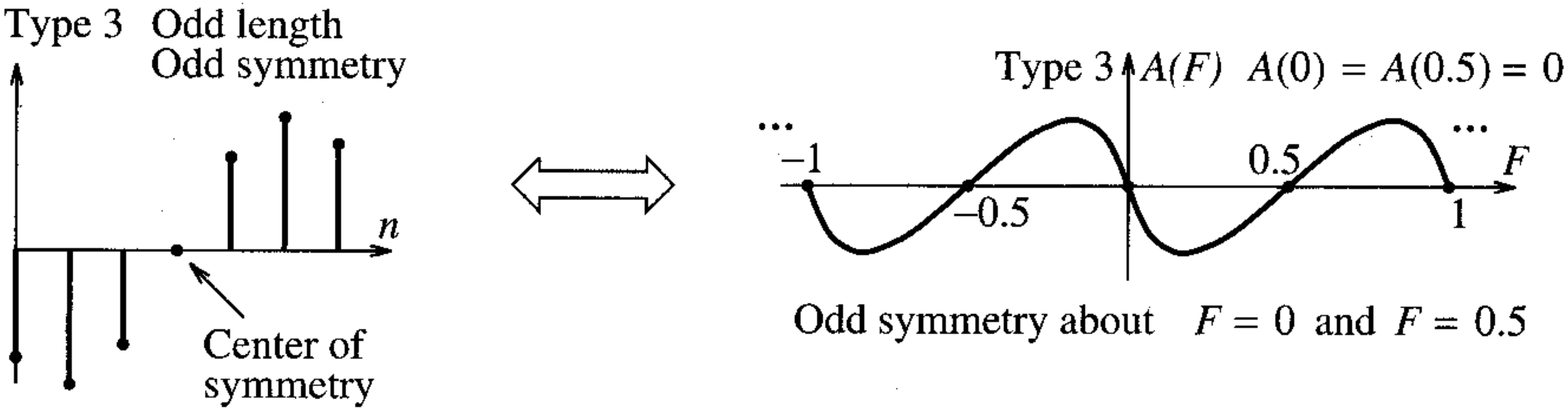
Features of a type 2 symmetric sequence

Linear phase = 2πMF

$$H_1(F) = \sum_{n=0}^{N-1} h[n]e^{-j2\pi MF} = \underbrace{\left[2\sum_{k=0}^{M-1/2} h[k]\cos[(M-k)2\pi F]\right]}_{\text{A(F)}} e^{-j2\pi MF}$$

From A.Ambardar, Analog and Digital Signal Processing, 2nd Edition, Brook/Cole, 1999.

FIR filter: odd length, odd symmetry



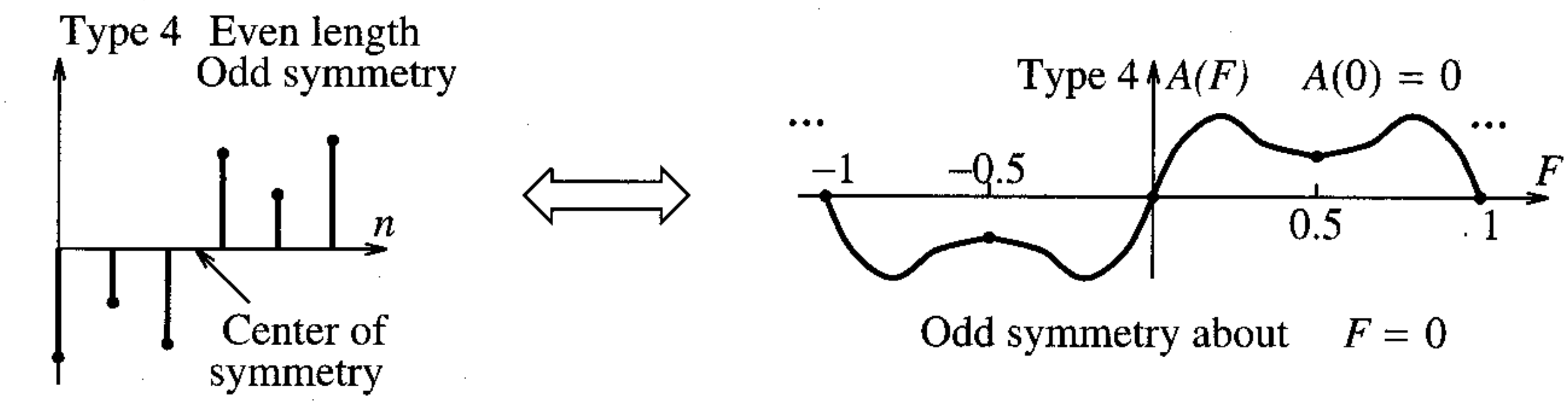
Features of a type 3 symmetric sequence

Linear phase = $2\pi MF + 90^0$

$$H_1(F)=\sum_{n=0}^{N-1} h[n]e^{-j2\pi MF} = j\underbrace{\left[2\sum_{k=0}^{M-1/2} h[k]\sin[(M-k)2\pi F]\right]}_{A(F)}e^{-j2\pi MF}$$

From A.Ambardar, Analog and Digital Signal Processing, 2nd Edition, Brook/Cole, 1999.

FIR filter: even length, odd symmetry

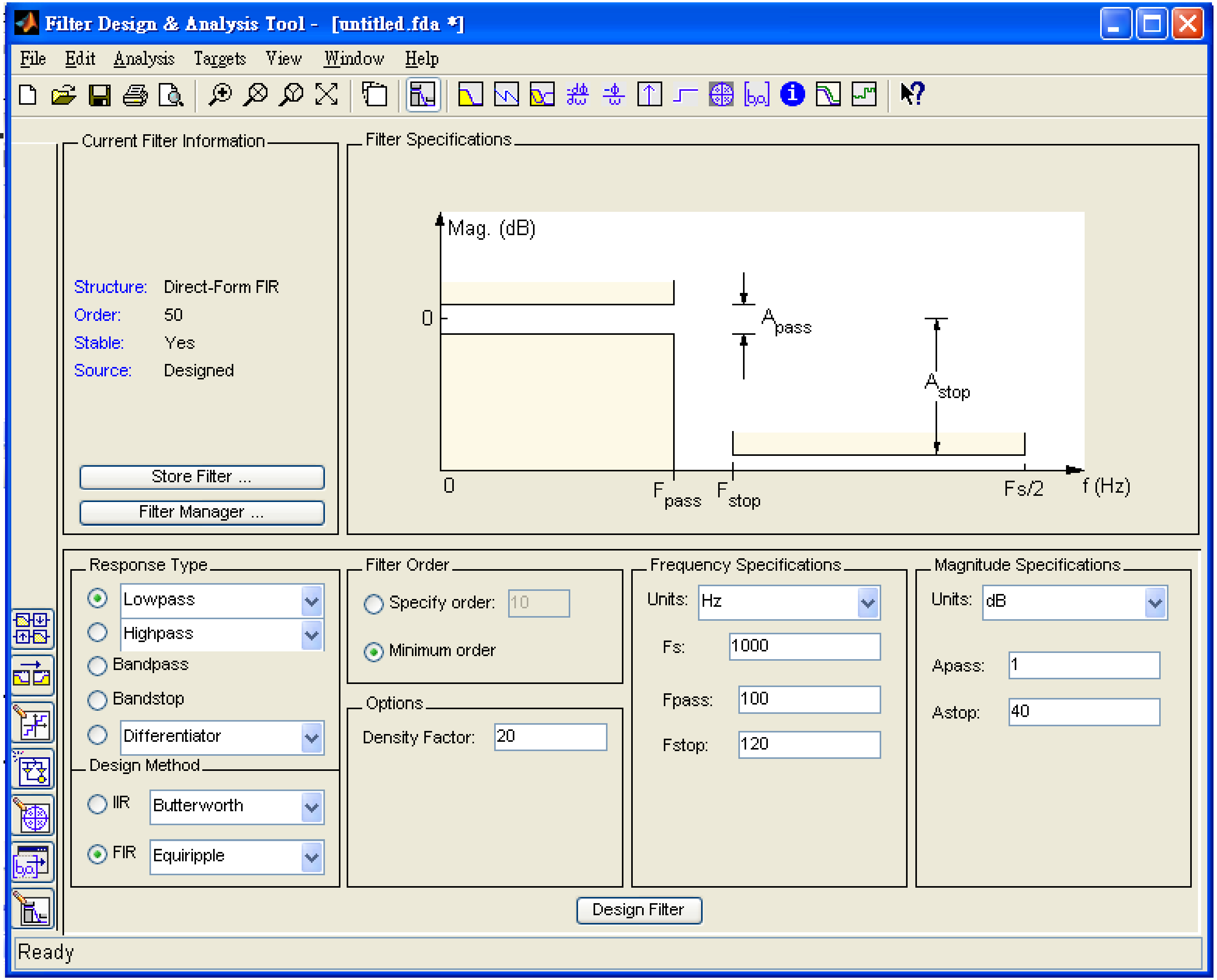


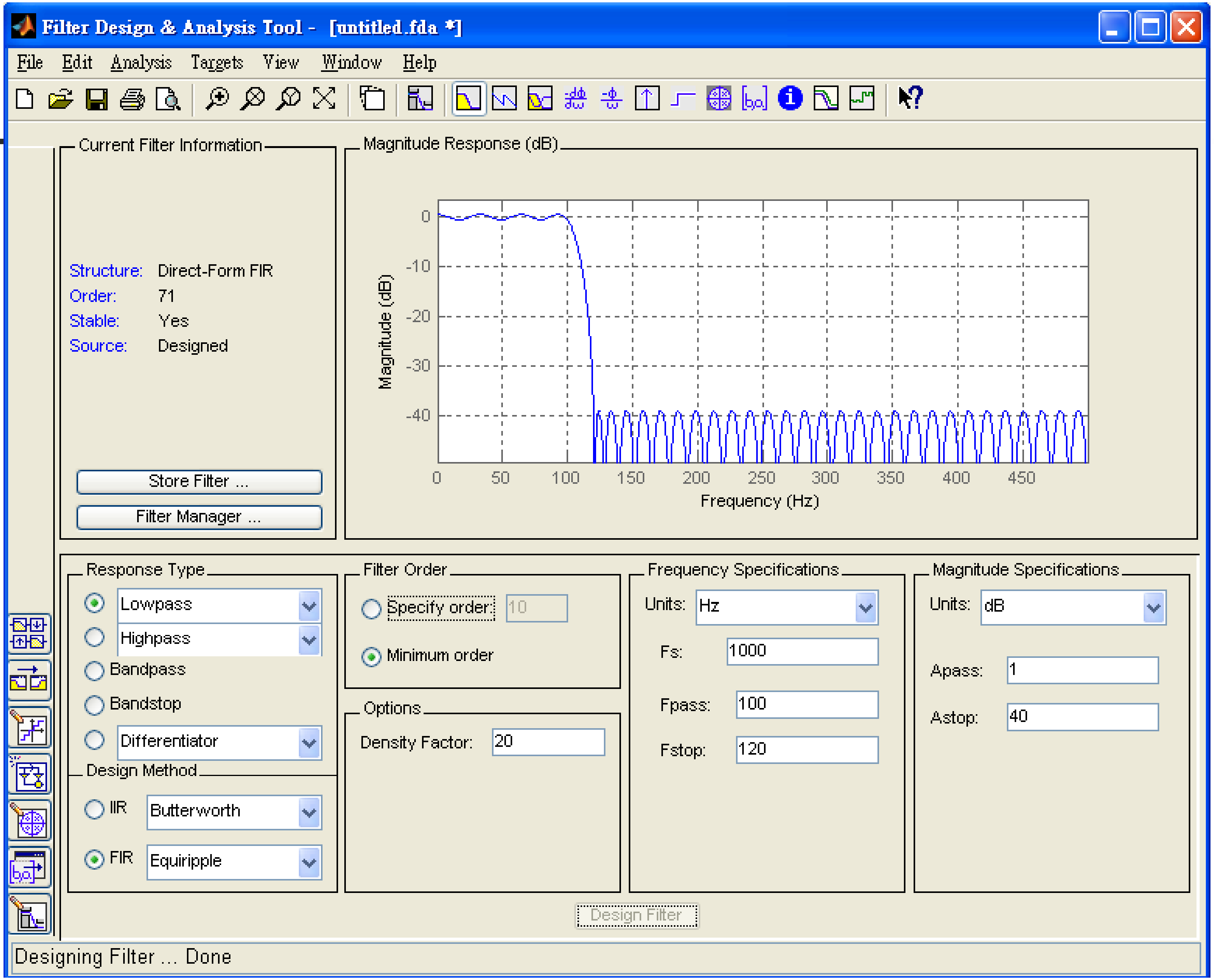
Features of a type 4 symmetric sequence

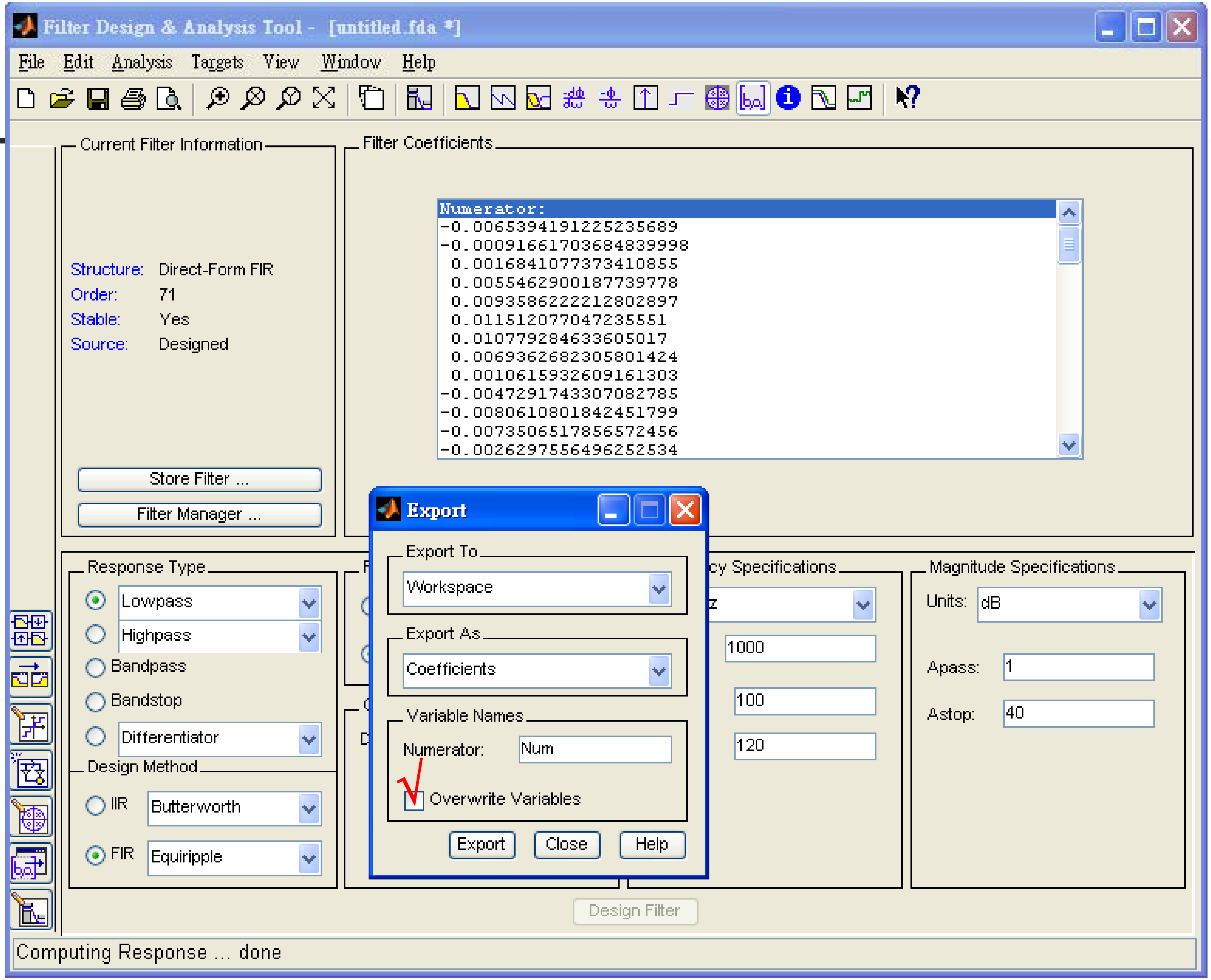
Linear phase = 2πMF+90⁰

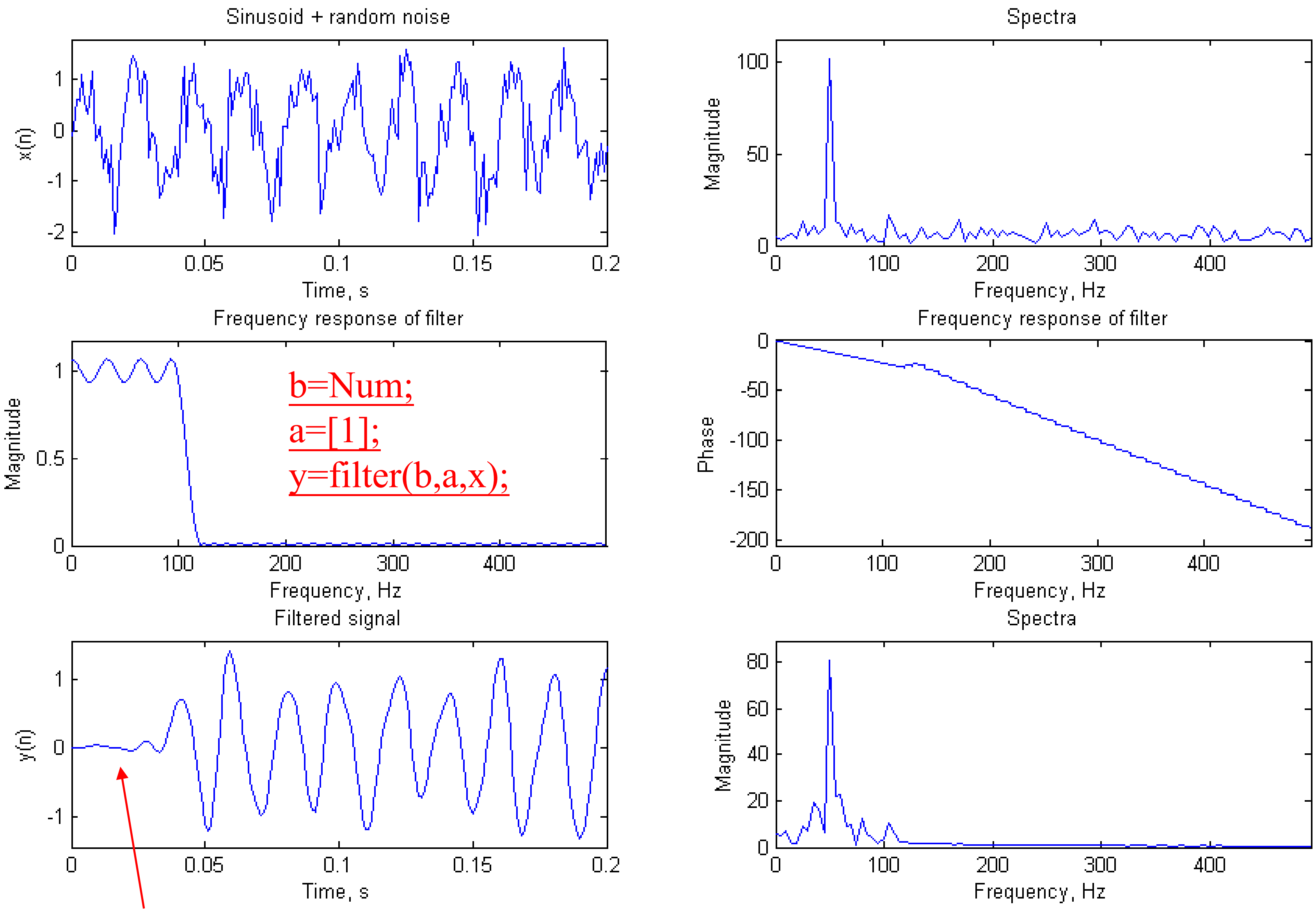
$$H_1(F) = \sum_{n=0}^{N-1} h[n]e^{-j2\pi MF} = \underbrace{j\left[2\sum_{k=0}^{M-1/2} h[k]\sin[(M-k)2\pi F]\right]}_{A(F)} e^{-j2\pi MF}$$

From A.Ambardar, Analog and Digital Signal Processing, 2nd Edition, Brook/Cole, 1999.


Filter Design & Analysis Tool - [untitled.fda *]
File Edit Analysis Targets View Window Help
Current Filter Information
Structure: Direct-Form FIR
Order: 50
Stable: Yes
Source: Designed
Store Filter ...
Filter Manager ...
Filter Specifications
Mag. (dB)
0
A pass
A stop
0
F pass
F stop
Fs/2
f (Hz)
Response Type
Lowpass
Highpass
Bandpass
Bandstop
Differentiator
Design Method
IIR
Butterworth
FIR
Equiripple
Filter Order
Specify order: 10
Minimum order
Options
Density Factor: 20
Frequency Specifications
Units: Hz
Fs: 1000
Fpass: 100
Fstop: 120
Magnitude Specifications
Units: dB
Apass: 1
Astop: 40
Design Filter
Ready


Filter Design & Analysis Tool - [untitled.fda *]
File Edit Analysis Targets View Window Help
Current Filter Information
Structure: Direct-Form FIR
Order: 71
Stable: Yes
Source: Designed
Store Filter ...
Filter Manager ...
Magnitude Response (dB)
Magnitude (dB)
0
-10
-20
-30
-40
0 50 100 150 200 250 300 350 400 450
Frequency (Hz)
Response Type
Lowpass
Highpass
Bandpass
Bandstop
Differentiator
Design Method
IIR Butterworth
FIR Equiripple
Filter Order
Specify order: 10
Minimum order
Options
Density Factor: 20
Frequency Specifications
Units: Hz
Fs: 1000
Fpass: 100
Fstop: 120
Magnitude Specifications
Units: dB
Apass: 1
Astop: 40
Design Filter
Designing Filter ... Done

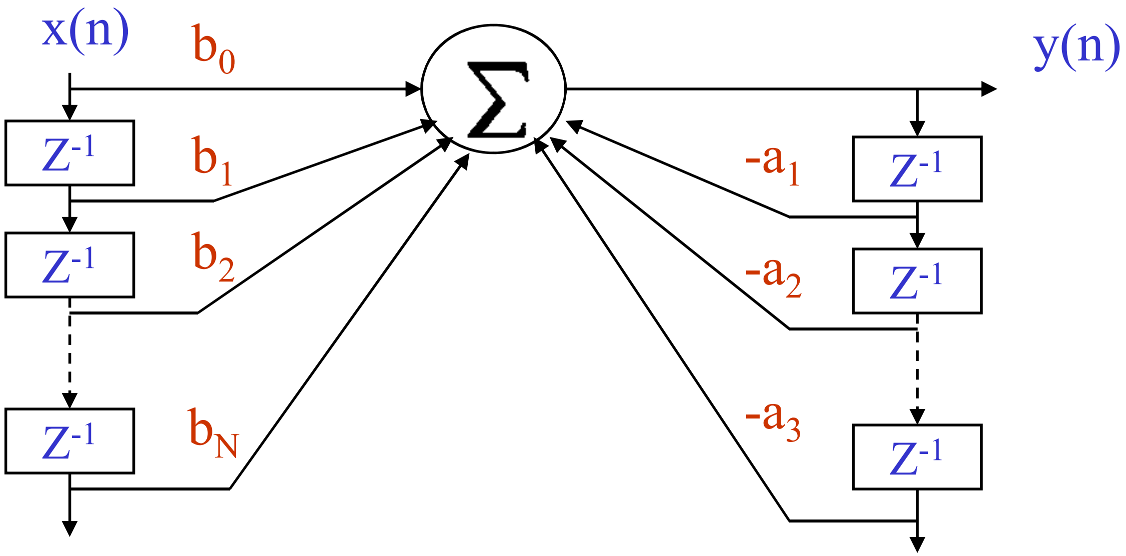

Filter Design & Analysis Tool - [untitled.fda *]
File Edit Analysis Targets View Window Help
Current Filter Information
Structure: Direct-Form FIR
Order: 71
Stable: Yes
Source: Designed
Store Filter ...
Filter Manager ...
Filter Coefficients
Numerator:
-0.0065394191225235689
-0.00091661703684839998
0.0016841077373410855
0.0055462900187739778
0.0093586222212802897
0.011512077047235551
0.010779284633605017
0.0069362682305801424
0.0010615932609161303
-0.0047291743307082785
-0.0080610801842451799
-0.0073506517856572456
-0.0026297556496252534
Export
Export To
Workspace
Export As
Coefficients
Variable Names
Numerator: Num
Overwrite Variables
Export Close Help
Response Type
Lowpass
Highpass
Bandpass
Bandstop
Differentiator
Design Method
IIR Butterworth
FIR Equiripple
Magnitude Specifications
Units: dB
Apass: 1
Astop: 40
1000
100
120
Design Filter
Computing Response ... done

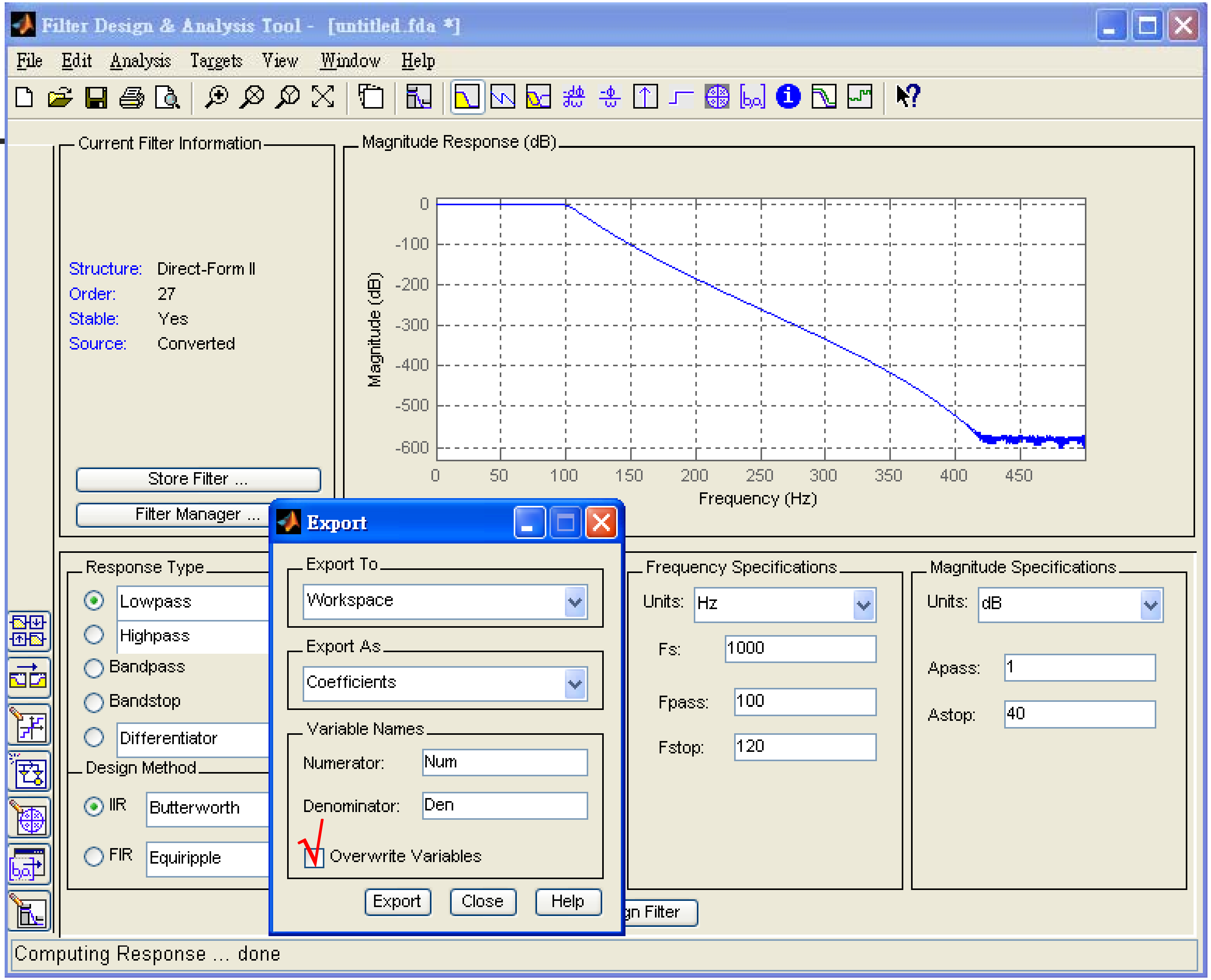


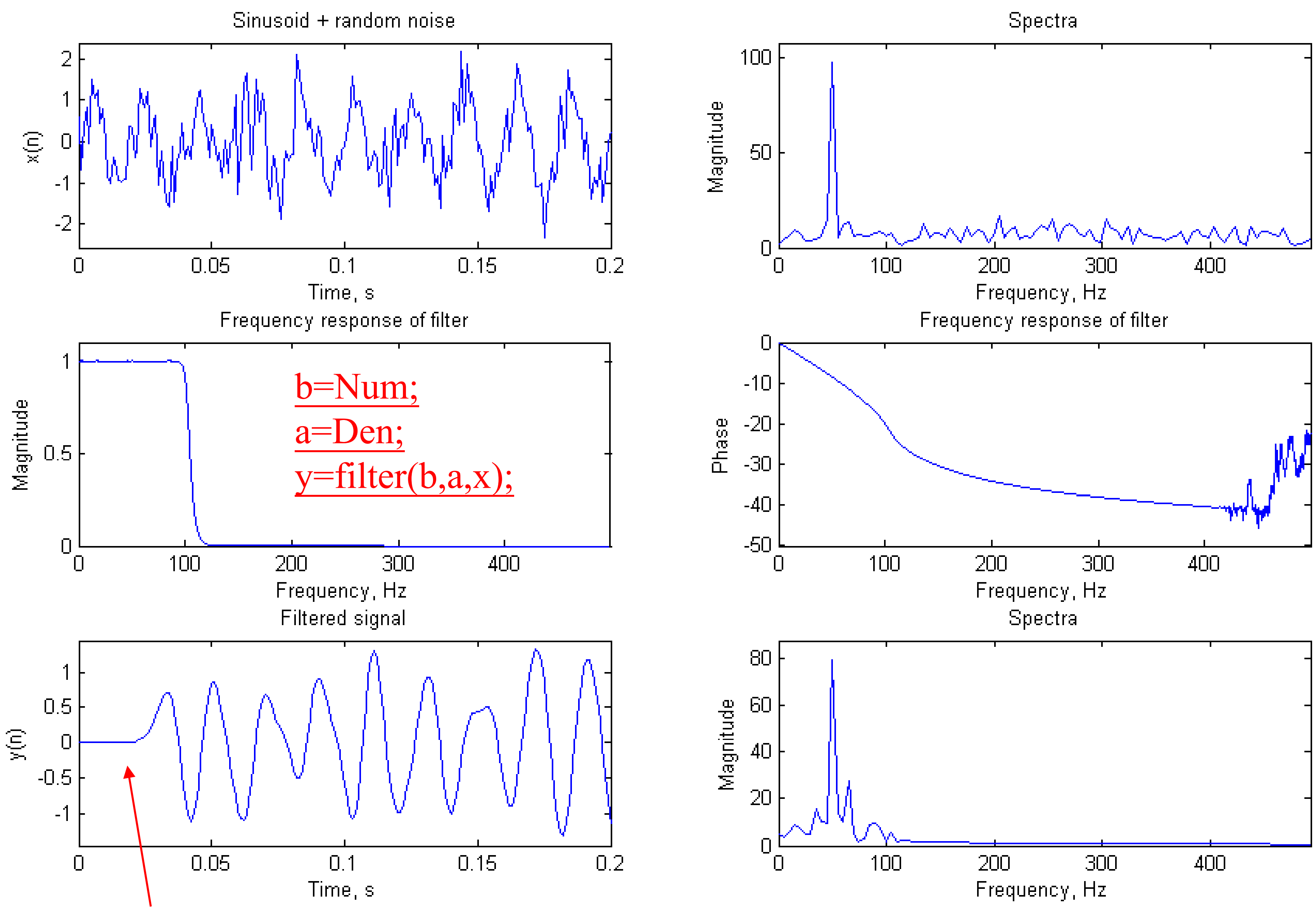
Transient. Group delay = filter length / 2

Infinite impulse response (IIR) filter

$$\sum_{p=0}^{N} a_p y(n-p) = \sum_{q=0}^{M} b_q x(n-q) \qquad H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{q=0}^{M} b_q z^{-q}}{\sum_{p=0}^{N} a_p z^{-p}}$$

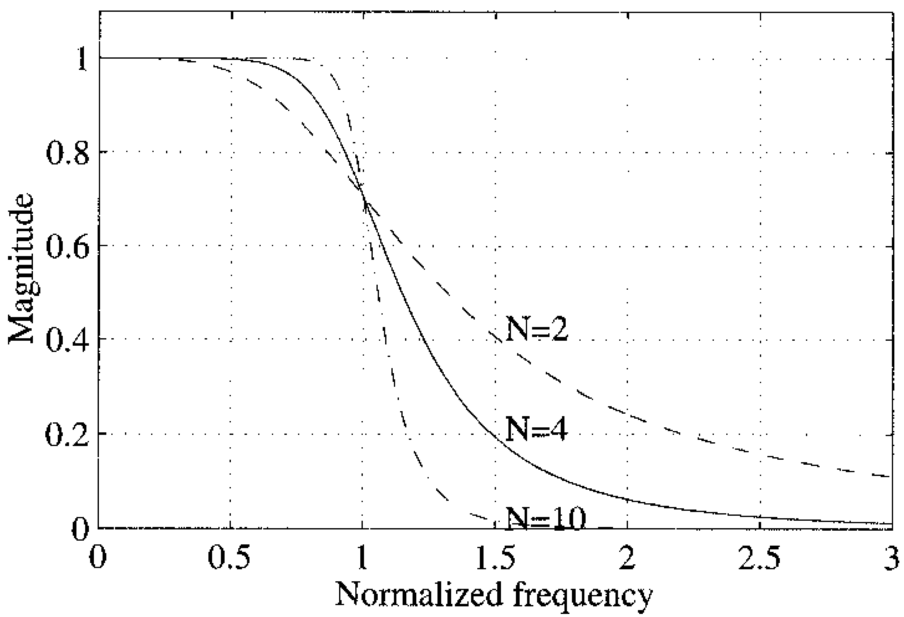



Filter Design & Analysis Tool - [untitled.fda *]
File Edit Analysis Targets View Window Help
Current Filter Information
Structure: Direct-Form II
Order: 27
Stable: Yes
Source: Converted
Store Filter ...
Filter Manager ...
Magnitude Response (dB)
Magnitude (dB)
0
-100
-200
-300
-400
-500
-600
0 50 100 150 200 250 300 350 400 450
Frequency (Hz)
Export
Export To
Workspace
Export As
Coefficients
Variable Names
Numerator: Num
Denominator: Den
Overwrite Variables
Export Close Help
Response Type
Lowpass
Highpass
Bandpass
Bandstop
Differentiator
Design Method
IIR Butterworth
FIR Equiripple
Frequency Specifications
Units: Hz
Fs: 1000
Fpass: 100
Fstop: 120
Magnitude Specifications
Units: dB
Apass: 1
Astop: 40
Computing Response ... done



Transient. Group delay is not constant, changed as frequency

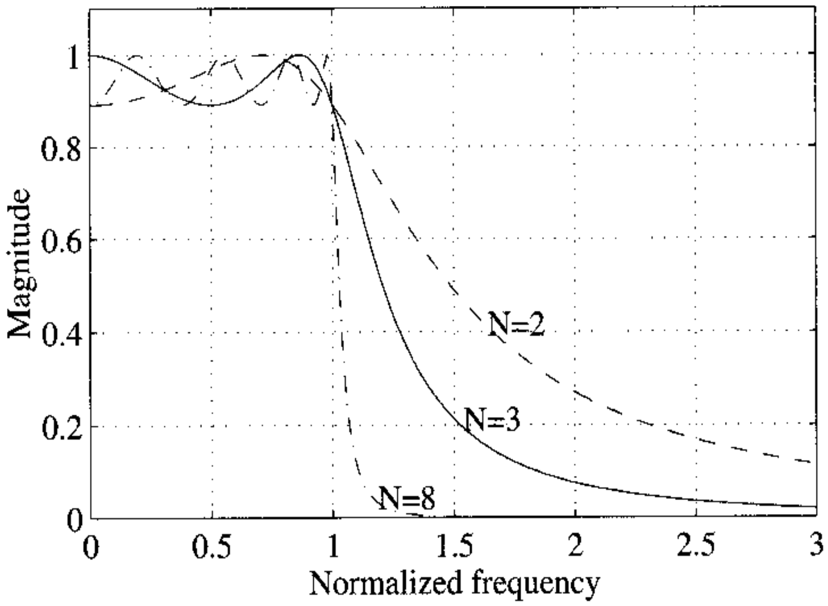
Butterworth approximation



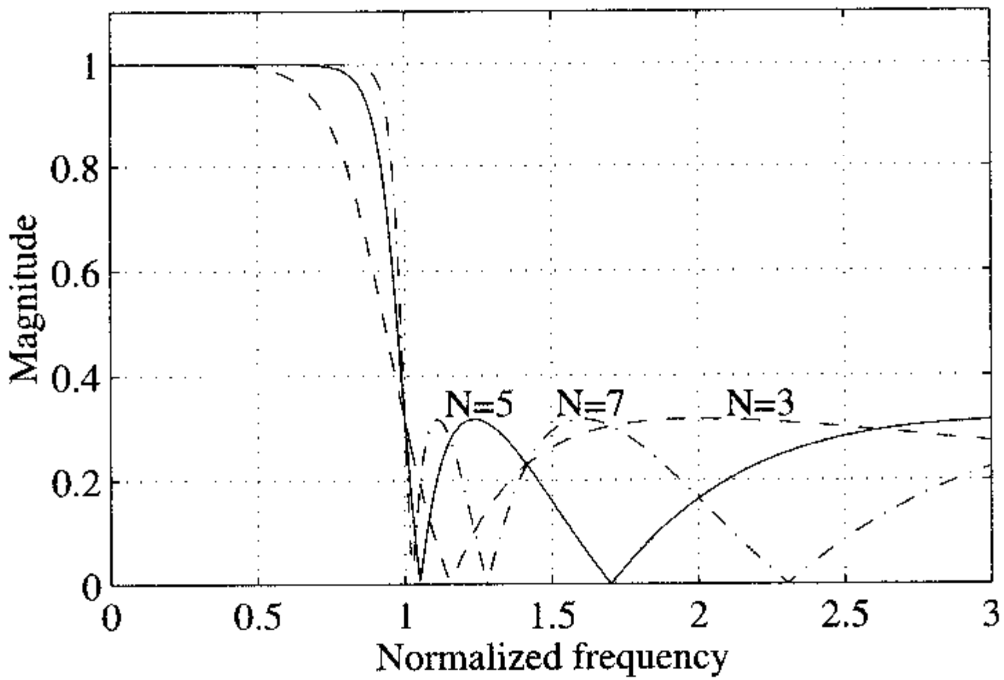
ω / ω_c

$$|H(j\omega)|^2 = \frac{1}{1 + (\omega / \omega_c)^{2N}}$$

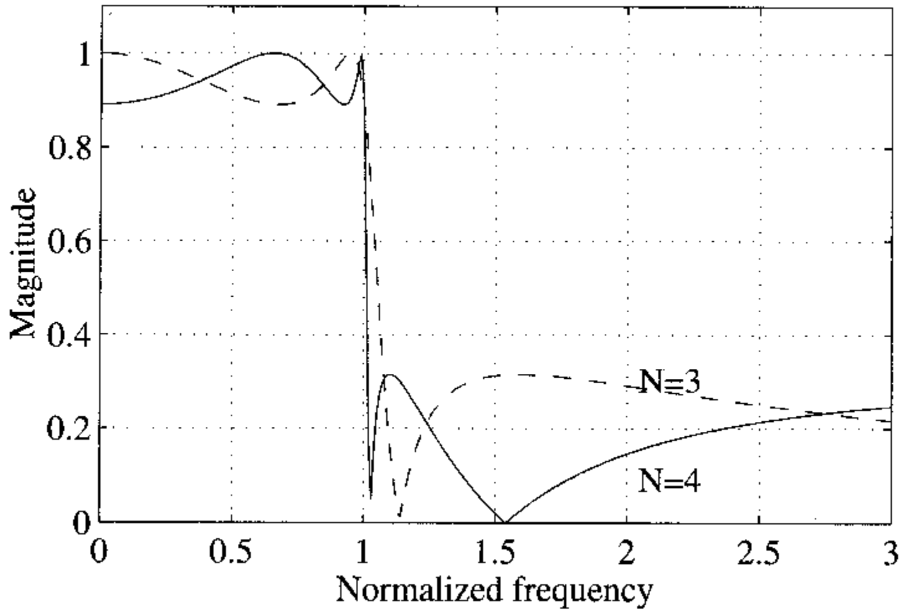
Type I Chebyshev approximation



Type II Chebyshev approximation

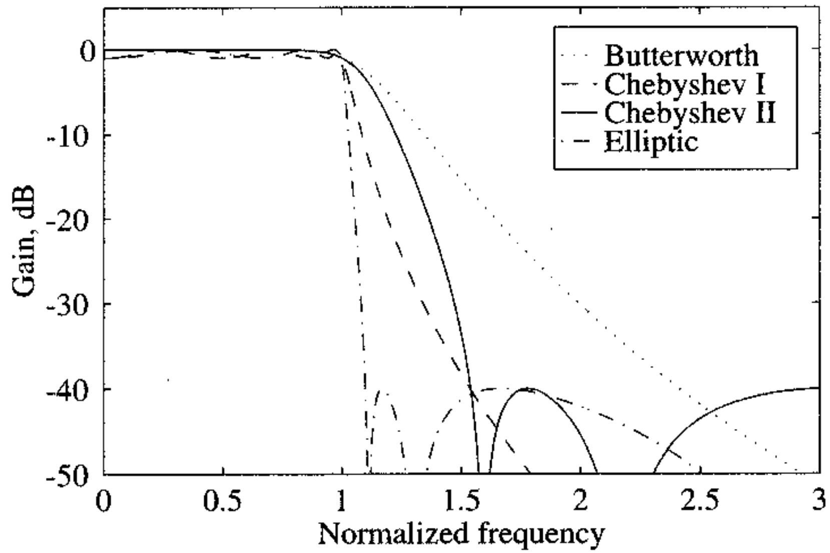


Elliptic approximation

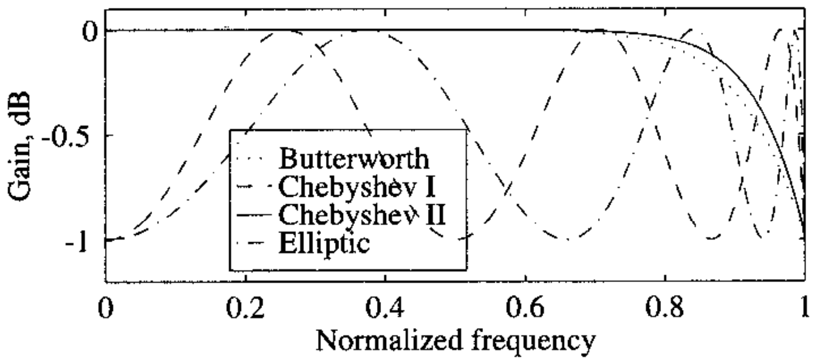


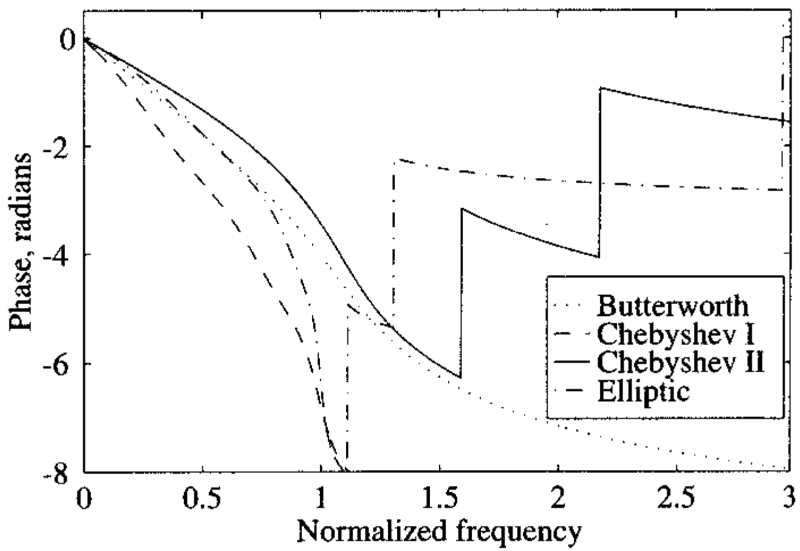
- Equiripple in both passband and stopband
- Meet filter requirements with the lowest order

Comparison



- Filter order = 6
- Maximum passband deviation = 1 dB
- Mimimum stopband attenuation = 40 dB
- Passband edge frequency = 1





Notch filter

$$H(z) = \frac{1 + b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}$$

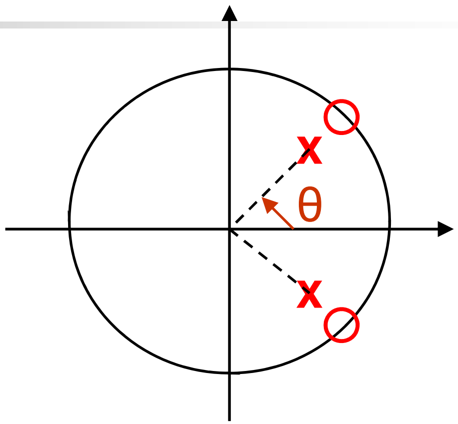
$$Let\ \ b_1 = -2\cos\theta\ \ and\ \ b_2 = 1$$

$$zeros = e^{\pm j\theta}$$

$$Let\ \ a_1 = -2r\cos\theta\ \ and\ \ a_2 = r^2$$

$$poles = r\,e^{\pm j\theta}$$

$$y(n) = x(n) + b_1 \cdot x(n-1) + b_2 \cdot x(n-2) - a_1 \cdot y(n-1) - a_2 y(n-2)$$



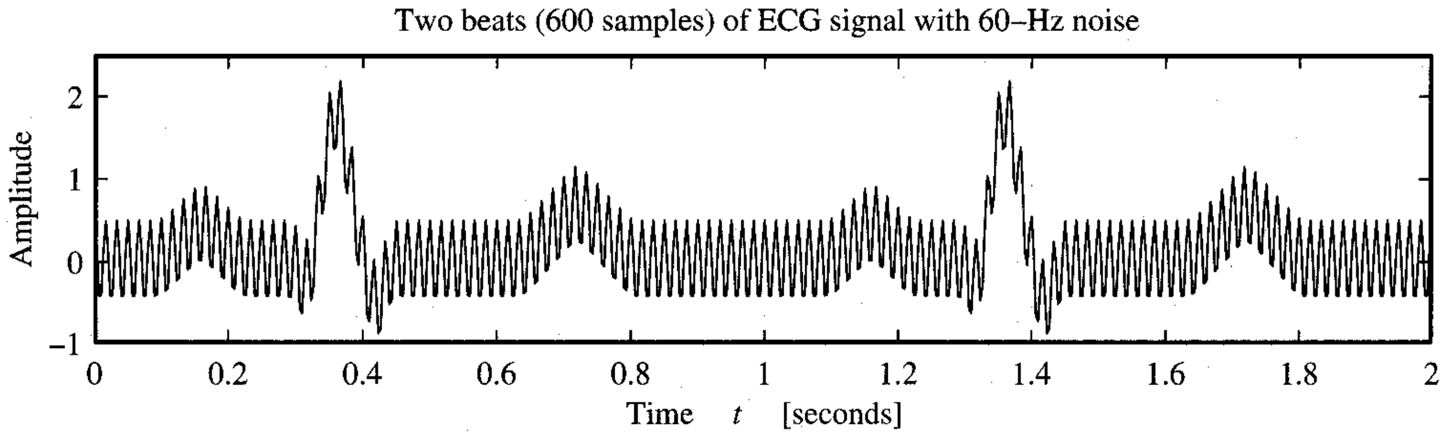
Selection of r and θ

$$\theta = \frac{f_c}{f_s} \times 2\pi$$

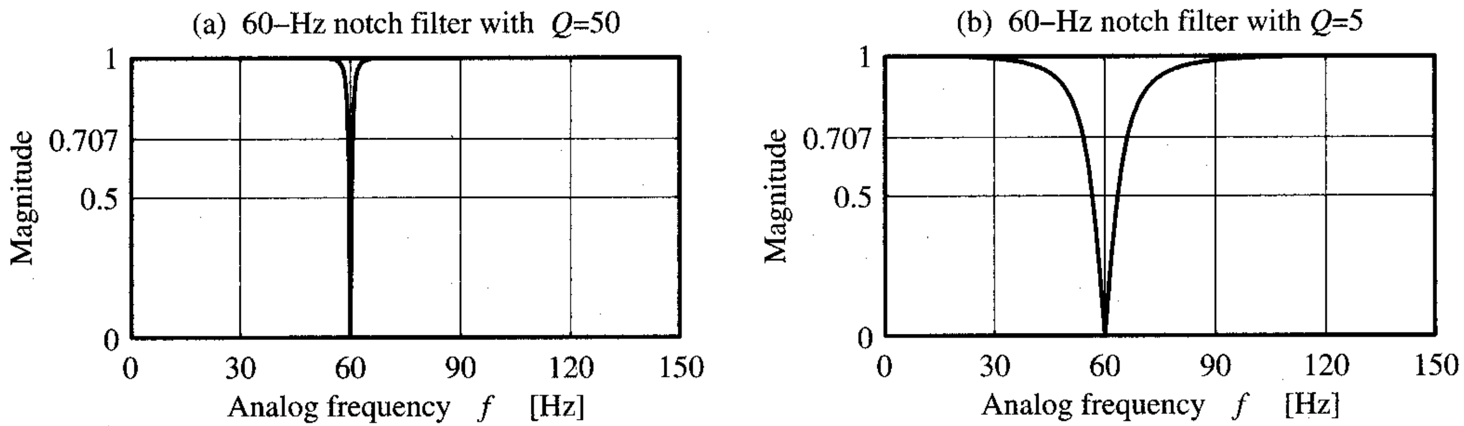
$$0 < r < 1$$

$$Q = \frac{\omega_0}{\Delta\omega} \quad where\ \Delta\omega \approx 2\frac{|1-r|}{\sqrt{r}}$$

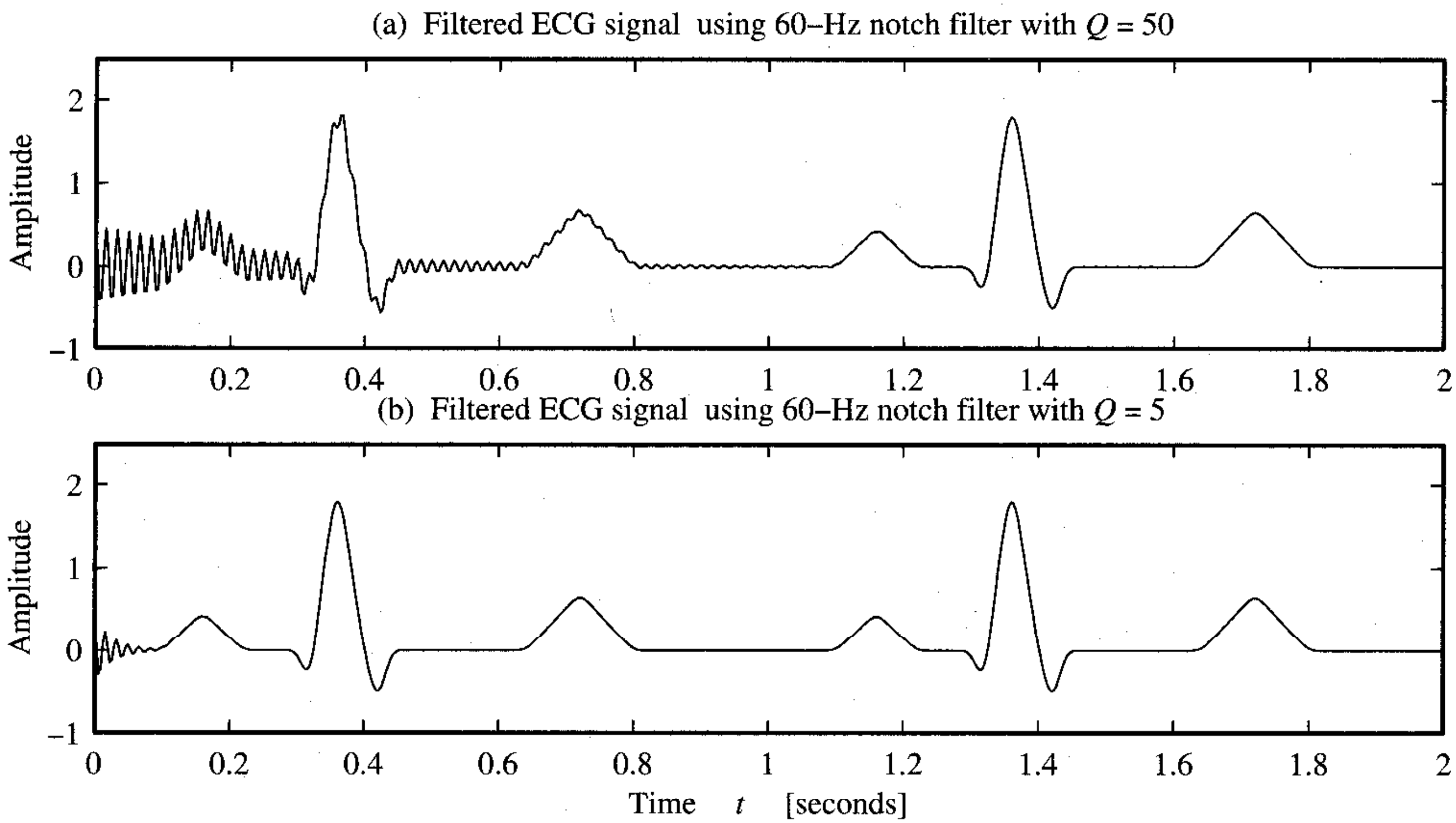
Example: Digital Notch Filter



Simulated ECG signal with 60-Hz interference



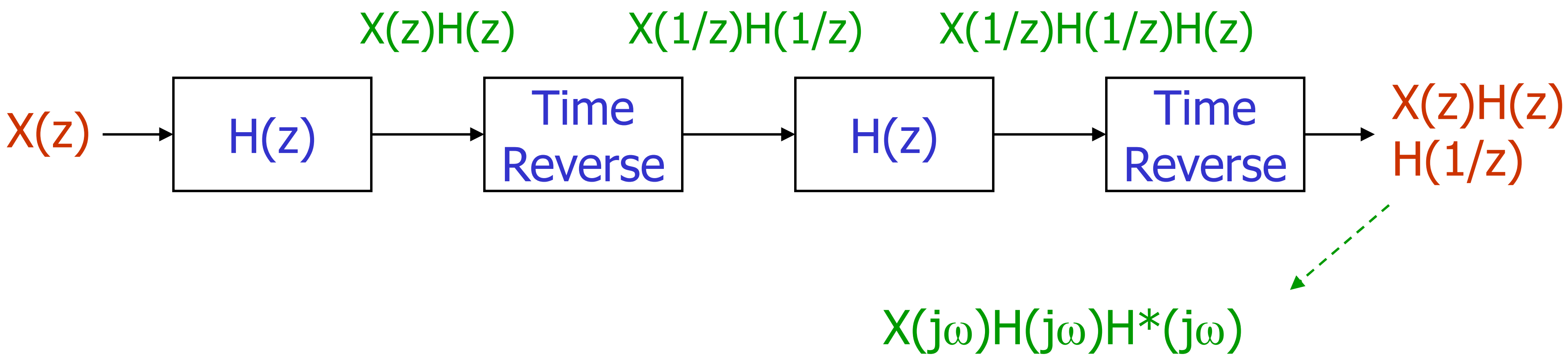
From A Ambardar, "Analog and digital signal processing, 2nd edition", Brooks/Cole, 1999.



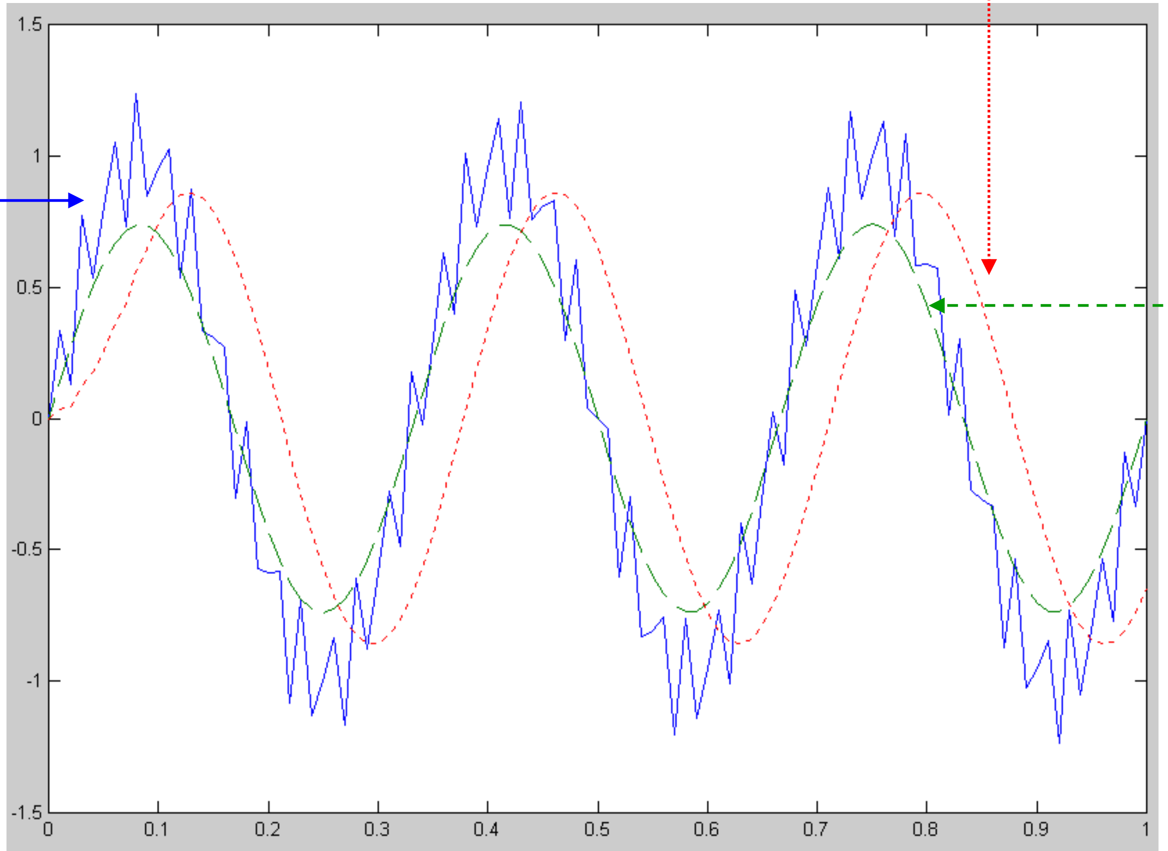
From A Ambardar, "Analog and digital signal processing, 2nd edition", Brooks/Cole, 1999.

Anticausal IIR Filter

- Zero phase
  - Zero group delay
- Zero transient if properly selected initial data
- Not easily implemented in real-time programming
- MATLAB: y= filtfilt(b, a, x)



```
fs = 100;
t = 0:1/fs:1;
x = sin(2*pi*t*3)+.25*sin(2*pi*t*40);
b = [1/2];    a=[1 –1/2];            % recursive averaging filter
y = filtfilt(b,a,x);                 % non–causal filtering
yy = filter(b,a,x);                  % causal filtering
plot(t,x,t,y,'--',t,yy,':')
```



Reference

- J.H. McClellan, R.W. Schafer, M.A. Yoder, Signal Processing First, Prentice Hall, 2003.