EPFL

Disal

Signals, Instruments, and Systems – W5

Introduction to Signal Processing – Bode Plots, Z-Transform, Digital Filters, Order and Type of Filters

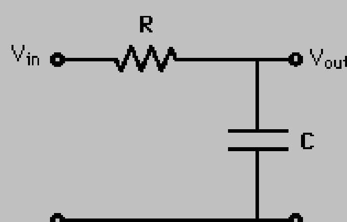
Outline

- Bode plots
- Z-transform and frequency responses/transfer functions of time-discrete systems
- Digital filters in time and frequency domains
- FIR and IIR filters
- Filter order and type

Bode Plots

Transfer Function of an Analog Low-Pass Filter

Circuit



Analog filter

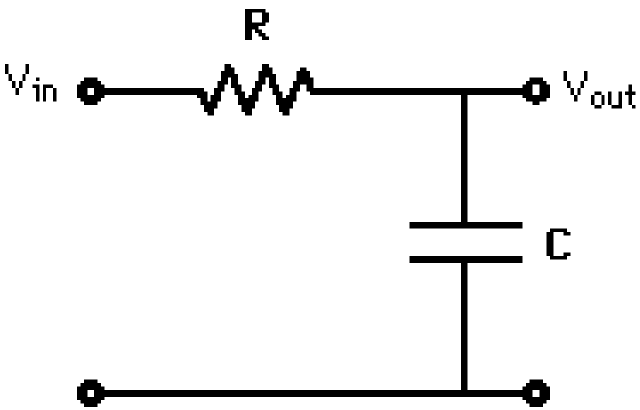
Laplace Tr.

$$H(s) = \frac{V_{out}(s)}{V_{in}(s)} = \frac{\overbrace{1}^{\text{Numerator}}}{\underbrace{1+sRC}_{\text{Denominator}}}$$

See also W4, s. 43

Analog Low-Pass Filter – From Transfer Function to Frequency Response

From W4, s. 44



$$H(s) = \frac{1}{1 + sRC}$$

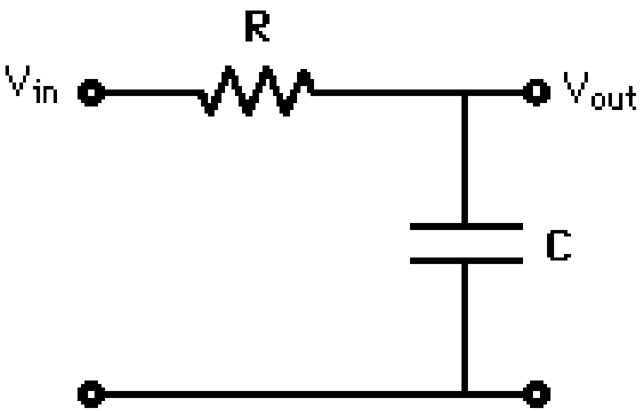
$$H(s = i\omega) = H(\omega) = \frac{1}{1 + i\omega RC}$$

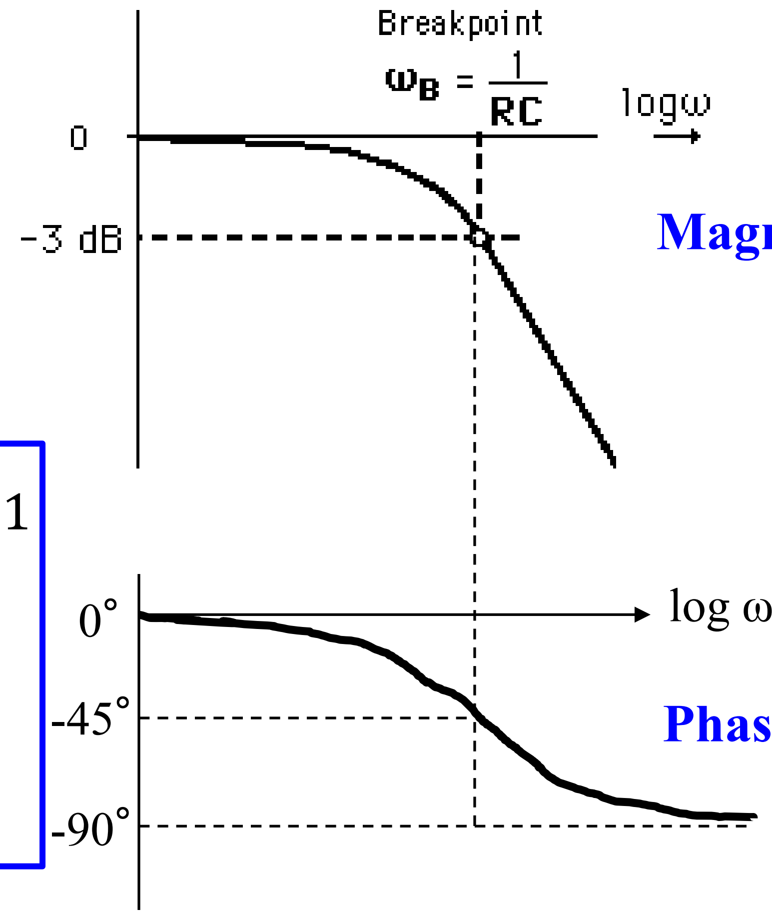
Magnitude: $$|H(\omega)| = \left|\frac{1}{1 + i\omega RC}\right| = \frac{1}{|1 + i\omega RC|} = \frac{1}{\sqrt{1 + \omega^2 R^2 C^2}}$$

Phase: $$\angle H(\omega) = \tan^{-1}\left(\frac{Im[H(\omega)]}{Re[H(\omega)]}\right) = \tan^{-1}(-\omega RC)$$

Analog Low-Pass Filter - Frequency Response (Sketch)

From W4, s. 45

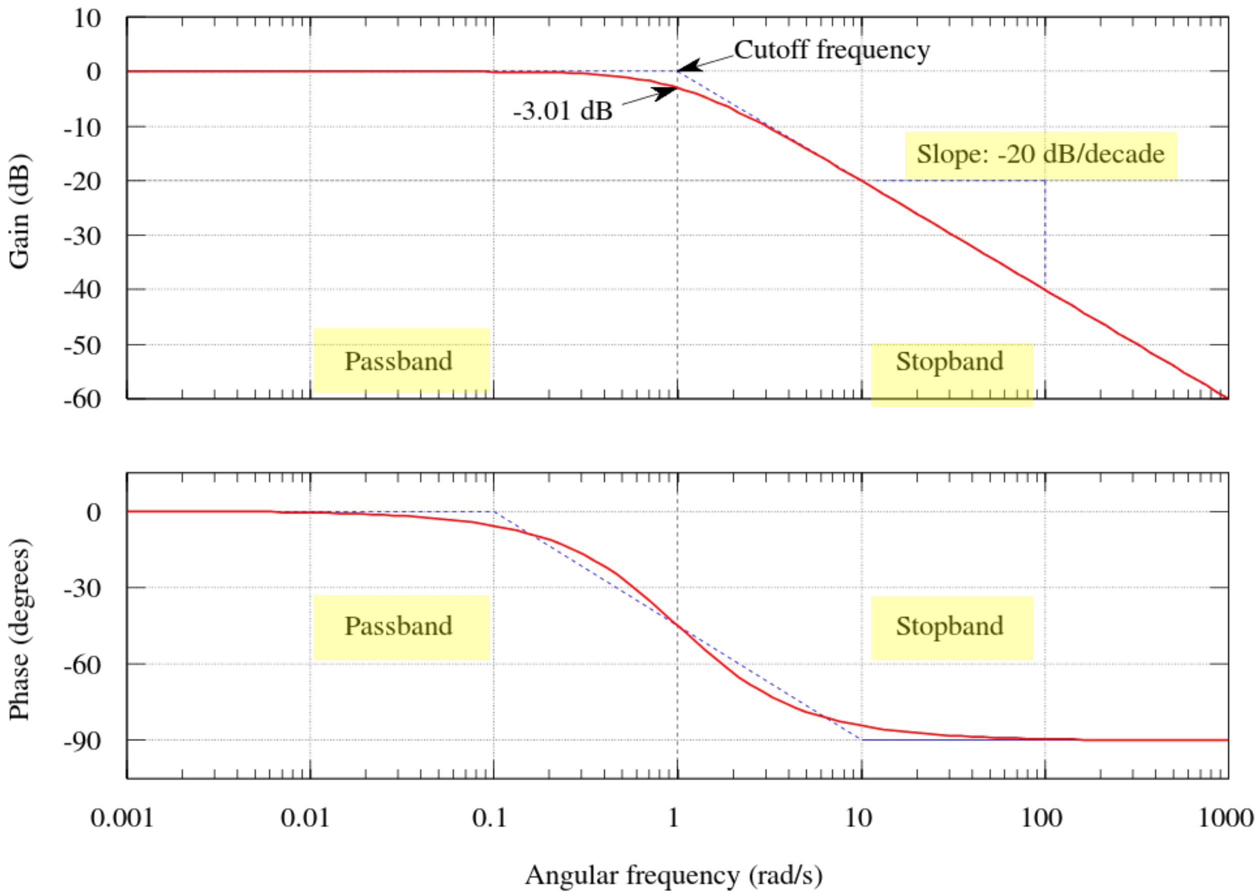




$$|H(\omega_B)| = \frac{1}{\sqrt{1+\omega_B^2 R^2 C^2}} = \frac{1}{\sqrt{2}} = 0.7071$$

i.e. -3 dB = about 71% of the undamped magnitude

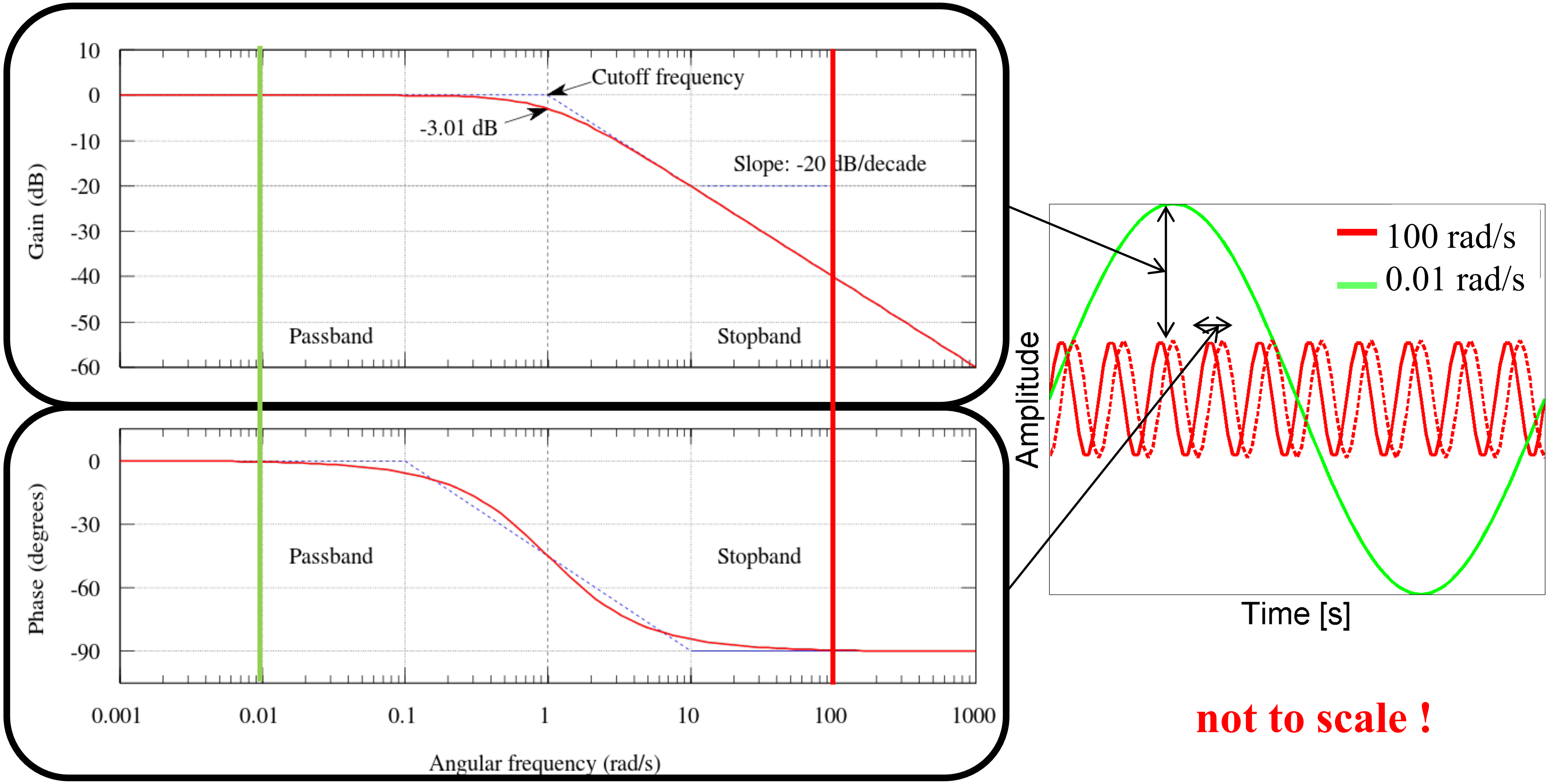
Analog Low-Pass Filter – Frequency Response (Bode Plot)



Notes:

- Cut-off frequency = breakpoint
- Magnitude or gain: log-log plot
- Phase: semi-log plot

Bode Plot: The Example of an Analog Low-Pass Filter



not to scale !

From Transfer Functions to Bode Plots

Generalized transfer function for LTI systems in continuous time:

$$H(s) = \frac{Num(s)}{Den(s)} = A\prod\frac{(s-x_n)^{a_n}}{(s-y_n)^{b_n}}$$

where x_n, y_n constants, a_n, $b_n > 0$

Frequency response: $s = i\omega$

See also W4, s. 22, s. 44, s. 46

Bode **magnitude**: $|H(s = i\omega)| = |H(i\omega)| = |H(\omega)|$

Bode **phase**: $\angle H(s = i\omega) = \angle H(i\omega) = \angle H(\omega)$

Zero (numerator = 0): every value of s where $\omega = |x_n|$

Pole (denominator = 0): every value of s where $\omega = |y_n|$

Bode Plots – Why handy?

$$X(\omega) \longrightarrow \boxed{H(\omega)} \longrightarrow Y(\omega)$$

Frequency domain

$$Y(\omega) = H(\omega)X(\omega)$$

$$|Y(\omega)| = |H(\omega)||X(\omega)|$$ Magnitude

$$\angle Y(\omega) = \angle H(\omega) + \angle X(\omega)$$ Phase

$$\log|Y(\omega)| = \log|H(\omega)| + \log|X(\omega)|$$ Magnitude Bode

Bode Plot – Why only positive and log scale for ω as well?

– If impulse response h(t) real, then $|H(\omega)|$ **even** function of ω and $\angle H(\omega)$ **odd** function of ω

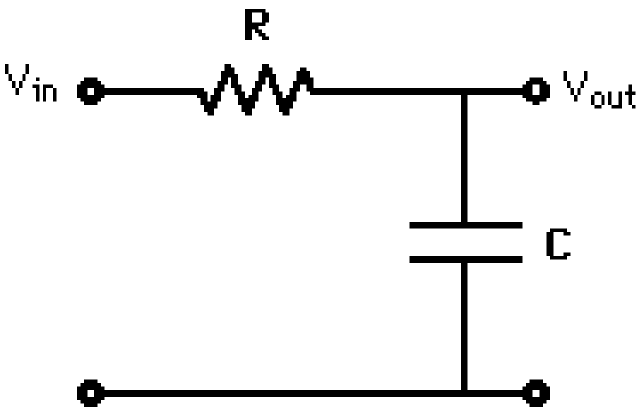
– Therefore plots for negative ω can be straightforwardly obtained from those of positive ω, so disregarded

– Log scale for frequency allows for covering a wider range of possible input frequencies on the same plot

Bode Plot – Generalized Rules

- Zero (numerator = 0)
  - Magnitude: +20 dB/decade
  - Phase: +90º; +45º/decade, starting 1 decade before zero
- Pole (denominator = 0)
  - Magnitude: -20 dB/decade
  - Phase: -90º; -45º/decade, starting 1 decade before pole

Example of Analog Low-Pass Filter – Frequency Response



$$H(s) = \frac{1}{1 + sRC}$$

From s.9:

$$\text{Num(s)} = 1 \neq 0 \ \forall s$$

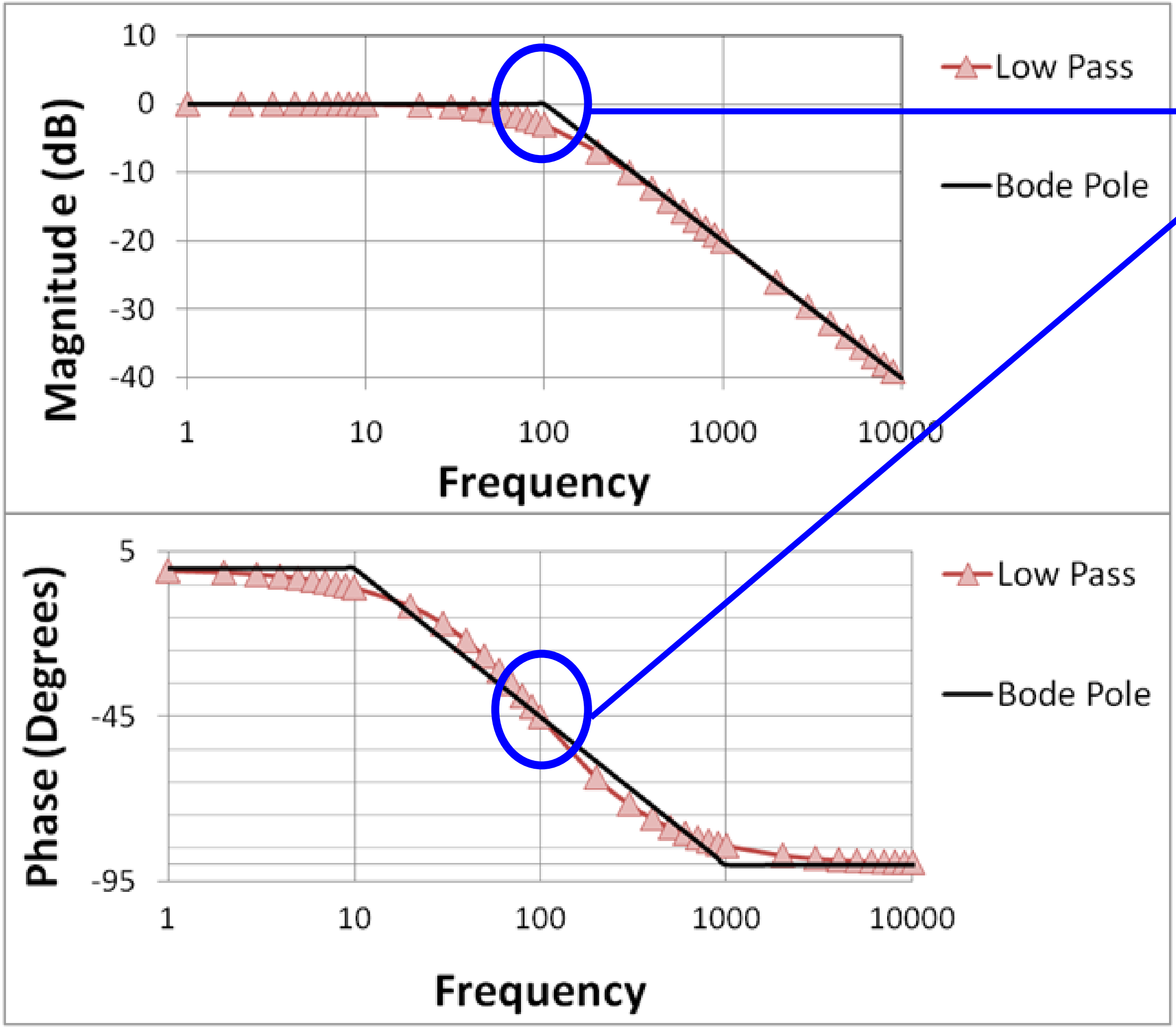
$$\text{Den(s)} = 1 + \text{sRC} = 0$$

$$\text{s} = \frac{-1}{RC} \Longrightarrow \omega = \left|\frac{-1}{RC}\right| = \frac{1}{RC}$$

No zeros

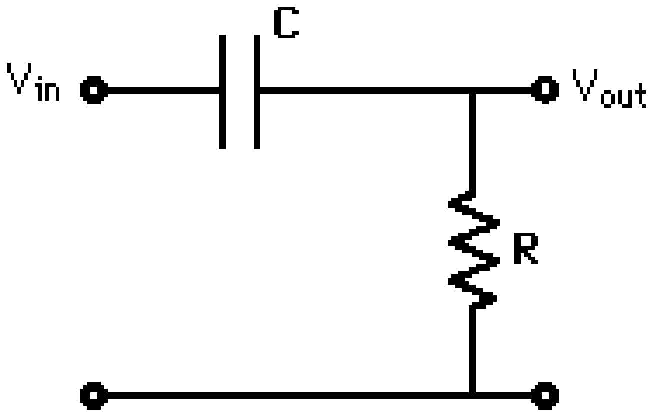
1 pole at $\omega_c = \frac{1}{RC}$

Example of Analog Low-Pass Filter – Bode Plot



Pole (at the cutoff frequency) $\omega_c = \frac{1}{RC} =$ 100 rad/s

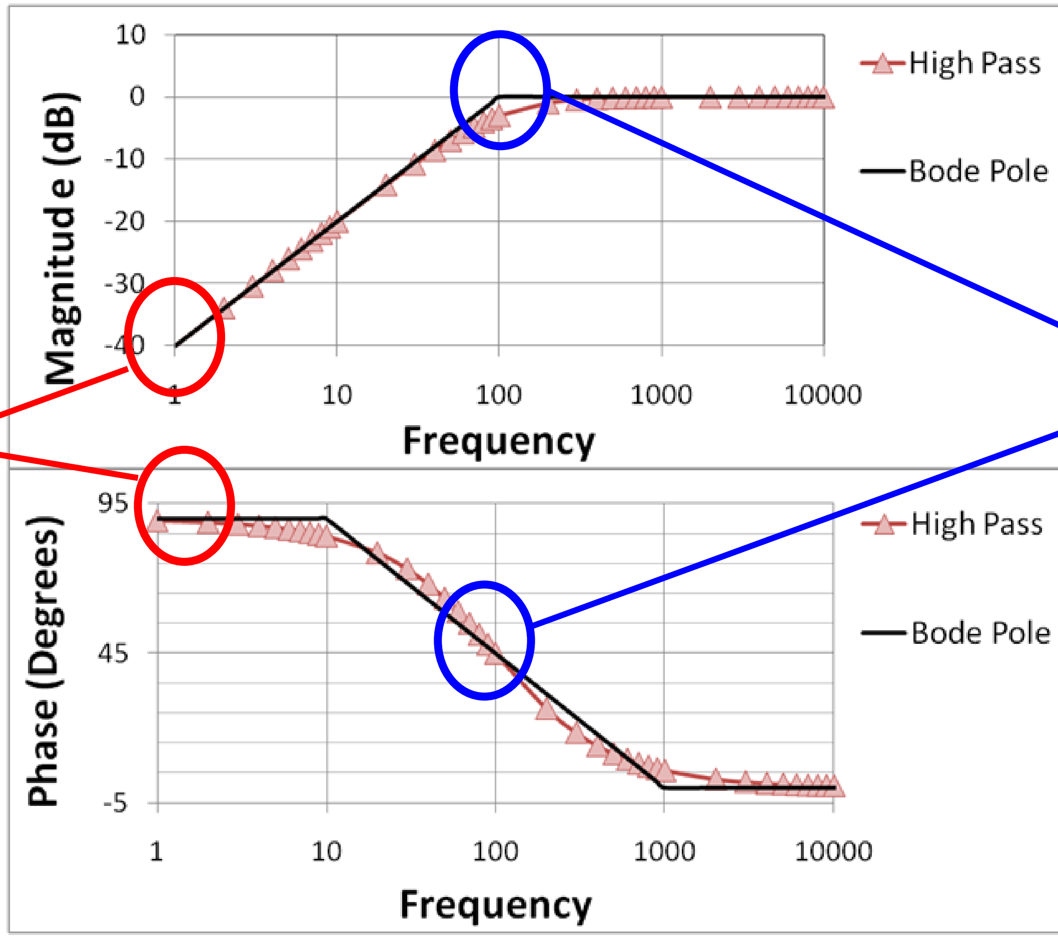
Example of Analog High-Pass Filter – Frequency Response



$$H(s) = \frac{sRC}{1 + sRC}$$

From s. 9: $\boxed{\begin{array}{l} 1 \text{ zero at } \omega_0 = 0 \\ 1 \text{ pole at } \omega_c = \frac{1}{RC} \end{array}}$

Example of Analog High-Pass Filter – Bode Plot



Zero at $\omega_0 = 0$

Pole at $\omega_c = \frac{1}{RC} = 100$ rad/s

A First-Order Filter with a Zero and a Pole

$$X(\omega) \longrightarrow \boxed{H(\omega)} \longrightarrow Y(\omega)$$

$\omega_{c1} = 100\ \mathrm{rad/s}$

$\omega_{c2} = 1000\ \mathrm{rad/s}$

$$H(s) = \frac{s - \omega_{c2}}{s - \omega_{c1}} = \frac{1}{s - \omega_{c1}}(s - \omega_{c2}) = H_1(s)H_2(s)$$

Transform $H(s)$ in $H(\omega)$ as before

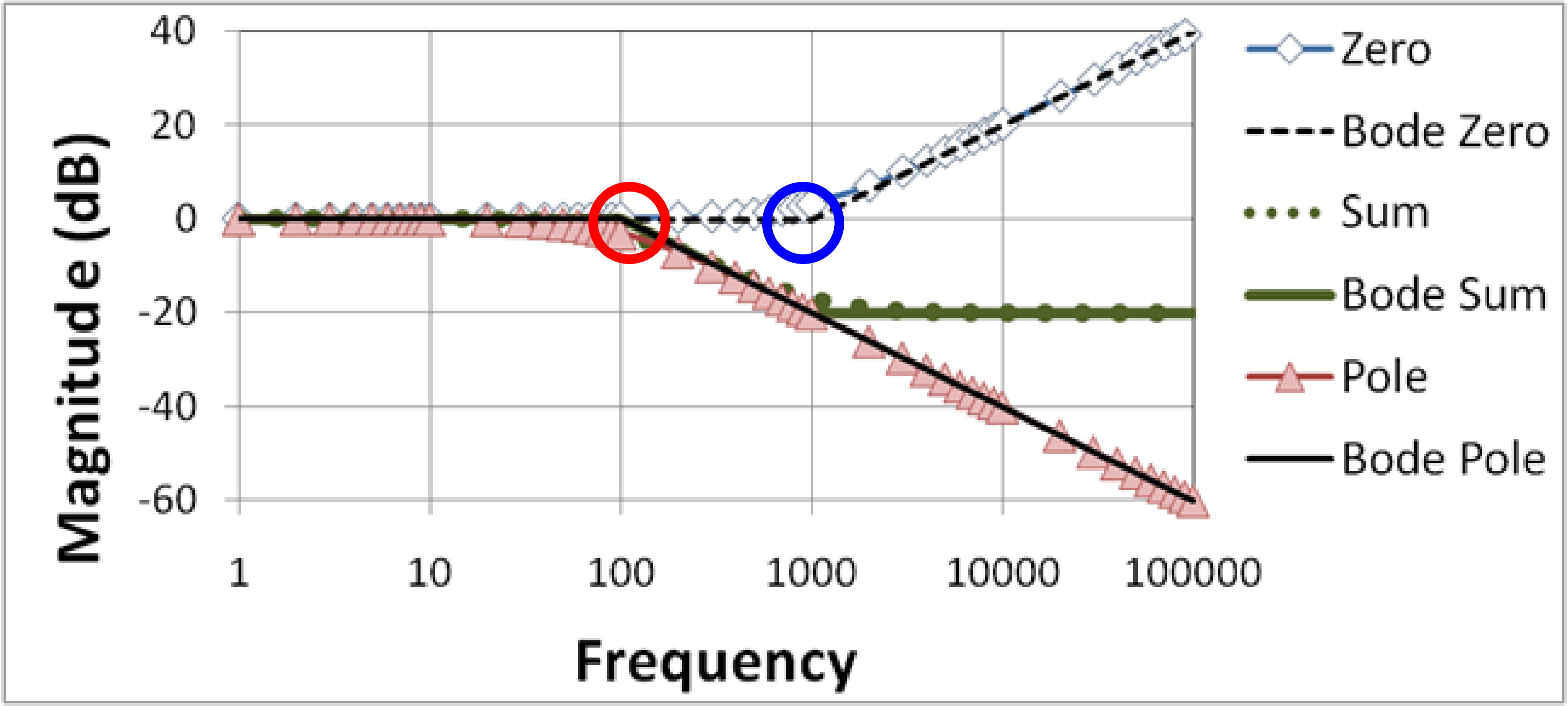
$$H(\omega) = H_1(\omega)H_2(\omega)$$

$|H(\omega)| = |H_1(\omega)||H_2(\omega)|$ Magnitude

$\angle H(\omega) = \angle H_1(\omega) + \angle H_2(\omega)$ Phase

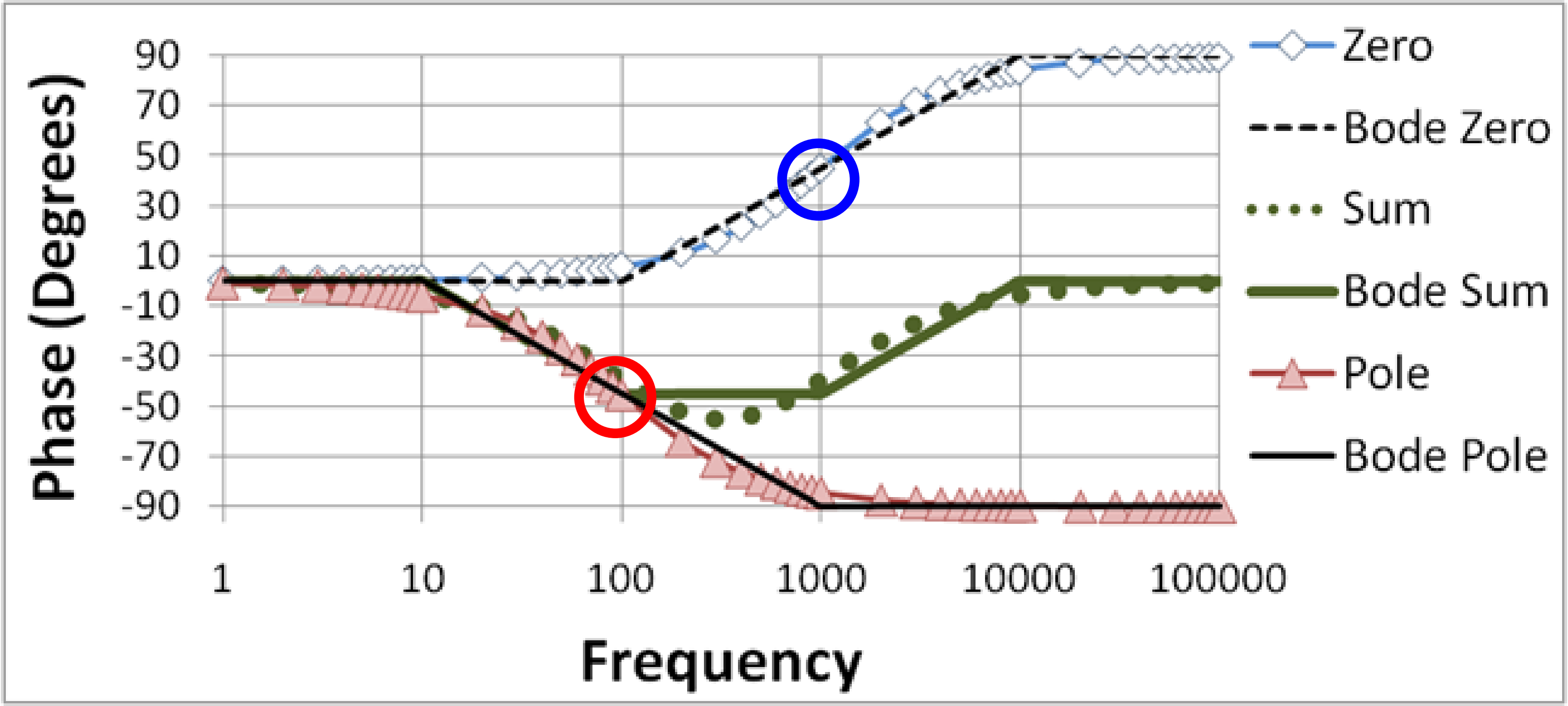
$\log|H(\omega)| = \log|H_1(\omega)| + \log|H_2(\omega)|$ Magnitude Bode

Bode Plot (Magnitude)



Pole at $\omega_{c1} = 100$ rad/s, magnitude -20 dB/decade
Zero at $\omega_{c2} = 1000$ rad/s, magnitude +20 dB/decade

Bode Plot (Phase)



Pole at $\omega_{c1} = 100\ \mathrm{rad/s}$, -90°; -45°/decade, starting 1 decade before pole

Zero at $\omega_{c2} = 1000\ \mathrm{rad/s}$, +90°; 45°/decade, starting 1 decade before zero

Types and Order of Analog Filters

Analog Filter Type and Order

- Several filters types exist and are defined by the polynomials at the numerator/denominator (Bessel, Butterworth, Tschebishev, etc.) of the transfer function
- Filter with a high order are closer to the ideal filter (rectangular function) but require more components to be implemented and consume more power

Higher Order Filters with Zeros and Poles

$$H(s) = \frac{Num(s)}{Den(s)} = \mathrm{A} \prod \frac{(s - x_n)^{a_n}}{(s - y_n)^{b_n}}$$

$$X(\omega) \longrightarrow \boxed{H(\omega)} \longrightarrow Y(\omega)$$

where x_n, y_n constants, a_n, $b_n > 0$

Filter order = highest order between *Num(s)* and *Den(s)* polynomials

Examples

$$H(s) = \frac{1}{(s - y_1)^3}$$

Third order with 3 poles in y_1

$$H(s) = \frac{(s - x_1)^2}{(s - y_1)^3}$$

Third order filter with 3 poles in y_1 and 2 zeros in x_1

$$H(s) = \frac{(s - x_1)(s - x_2)}{(s - y_1)^2}$$

Second order filter with 2 poles in y_1 and 1 zeros in x_1
1 zeros in x_2

Notes:

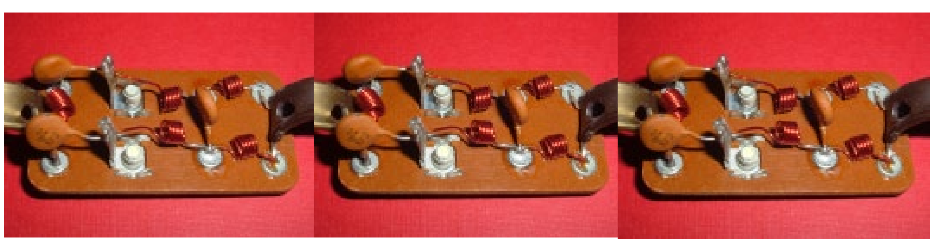
- poles and zeros of H(s) can be represented in the zero-pole plot (complex s-plane) for further analysis (e.g., stability).
- this is valid also for cascaded blocks of LTI systems (e.g., cascaded filters)

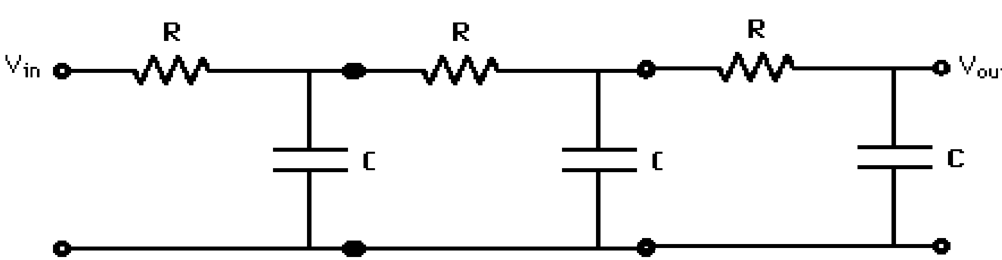
Bode Plots of Higher Filter Order

- 1st order is equivalent to ±20dB per decade
- Each successive order adds ± 20dB per decade
- Example: 3rd order analog low-pass filter

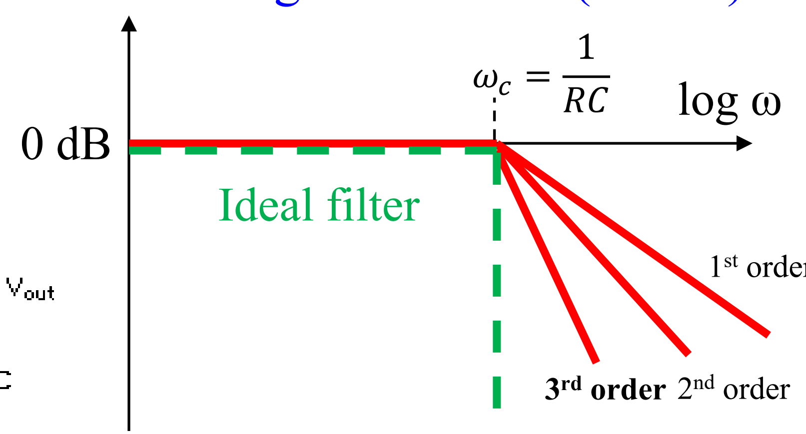
$$H(s) = \frac{1}{(1+sRC)^3} \Rightarrow |H(\omega)| = \frac{1}{(1+\omega^2R^2C^2)^{\frac{3}{2}}}$$

See s. 5 for first order





Magnitude Bode (sketch)



More Transforms for Time-Discrete Systems

Discrete-Time Fourier Transform

- Corresponds to the Fourier Transform for discrete-time signals (different from the Discrete Fourier Transform, a finite, bounded approximation of the Fourier Transform for digital devices)
- Transform discrete-time signals from time-domain to frequency domain (continuous spectrum)

$$X(\omega) = \sum_{n=-\infty}^{\infty} x[n] \cdot e^{-i\omega n}$$

Note: compare with DFT notation on W2, s. 35:

$$X[k] = \sum_{n=0}^{N-1} x[n] \cdot e^{\frac{-2\pi i}{N}kn}$$

$$k = 0, \cdots, N-1$$

Z-Transform

- Transform signals from time-domain to a complex frequency-domain representation (z-plane)
- Corresponds to Laplace transform for time-discrete signals
- The Discrete-Time Fourier Transform is a special case of the Z-Transform with $z = e^{i\omega}$

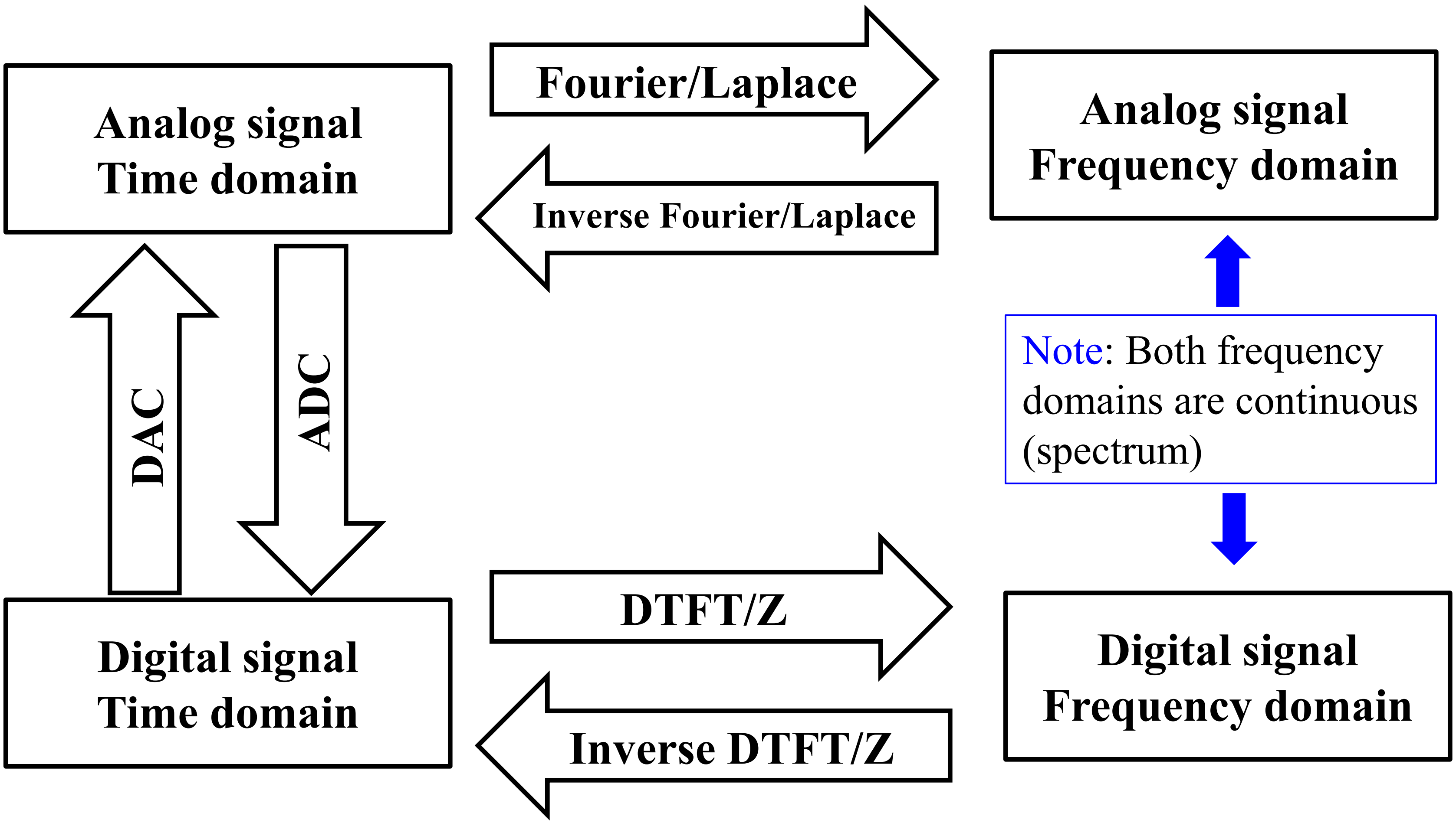
$$X(z) = Z\{x[n]\} = \sum_{n=-\infty}^{\infty} x[n] z^{-n}$$

$$z = Ae^{i\phi} \text{ or } z = A(\cos\phi + i\sin\phi)$$

Some Properties of Z-Transform

Property	Signal	z-Transform
	$x[n]$	$X(z)$
	$x_1[n]$	$X_1(z)$
	$x_2[n]$	$X_2(z)$
Linearity	$ax_1[n] + bx_2[n]$	$aX_1(z) + bX_2(z)$
Time shifting	$x[n - n_0]$	$z^{-n_0} X(z)$
Scaling in the z-domain	$e^{j\omega_0 n} x[n]$	$X(e^{-j\omega_0} z)$
	$z_0^n x[n]$	$X\left(\frac{z}{z_0}\right)$
	$a^n x[n]$	$X(a^{-1} z)$
Time reversal	$x[-n]$	$X(z^{-1})$
Time expansion	$x_{(k)}[n] = \begin{cases} x[r], & n = rk \\ 0, & n \neq rk \end{cases}$ for some integer r	$X(z^k)$
Conjugation	$x^*[n]$	$X^*(z^*)$
Convolution	$x_1[n] * x_2[n]$	$X_1(z)X_2(z)$
First difference	$x[n] - x[n-1]$	$(1 - z^{-1})X(z)$
Accumulation	$\sum_{k=-\infty}^{n} x[k]$	$\frac{1}{1 - z^{-1}} X(z)$
Differentiation in the z-domain	$nx[n]$	$-z \frac{dX(z)}{dz}$

Transform Overview



Frequency Responses and Transfer Functions for Time-Discrete LTI Systems

Frequency Response

From W4, s. 28

Continuous time

$$x(t) = \sum_{k=-\infty}^{\infty} a_k e^{ik\omega_0 t} \longrightarrow \boxed{h(t)} \longrightarrow y(t) = \sum_{k=-\infty}^{\infty} H(k\omega_0) a_k e^{ik\omega_0 t}$$

$$a_k \rightarrow H(k\omega_0) a_k$$

$$H(k\omega_0) = \underbrace{|H(k\omega_0)|}_{\textbf{Magnitude or Amplitude or Gain}} e^{\overbrace{i\angle H(k\omega_0)}^{\textbf{Phase}}}$$

By linearity a sum of frequencies go out of the LTI filter only with different amplitude and phase.

Frequency Response

Discrete time

$$x[n] = \sum_{k=-\infty}^{\infty} a_k e^{ik\omega_0 n} \longrightarrow \boxed{h[n]} \longrightarrow y[n] = \sum_{k=-\infty}^{\infty} H(k\omega_0) a_k e^{ik\omega_0 n}$$

$$a_k \rightarrow H(k\omega_0) a_k$$

$$H(k\omega_0) = \underbrace{|H(k\omega_0)|}_{\textbf{Magnitude or Amplitude or Gain}} e^{\overbrace{i\angle H(k\omega_0)}^{\textbf{Phase}}}$$

By linearity a sum of frequencies go out of the LTI filter
only with different amplitude and phase.

Time-Continuous Transfer Function

h(t) : impulse response (CT)
H(s) : transfer function (stationary and transient frequency response) of the filter/system, i.e Laplace transform of h(t)

From W4, s. 29

Time domain

$$f(t) \longrightarrow \boxed{h(t)} \longrightarrow (f * h)(t)$$

Complex frequency domain (s-plane)

$$F(s) \longrightarrow \boxed{H(s)} \longrightarrow F(s)H(s)$$

Reminder: a convolution in the time domain is a multiplication in the frequency domain

Time-Discrete Transfer Function

h[n] : impulse response (DT)
H(z) : transfer function (stationary and transient frequency response) of the filter/system, i.e. Z-transform of h[n]

Time domain

$$f[n] \longrightarrow \boxed{h[n]} \longrightarrow (f * h)[n]$$

Complex frequency domain (z-plane)

$$F(z) \longrightarrow \boxed{H(z)} \longrightarrow F(z)H(z)$$

Reminder: a convolution in the time domain is a multiplication in the frequency domain

From Transfer Functions to Frequency Reponses

- Typical configuration when digital filtering is applied to analog signals:

- Frequency response useful when input x[n] = time-discrete sum of sinusoids, then output y[n] also time-discrete sum of sinusoids
- Focus on the spectrum

From Transfer Functions to Frequency Reponses

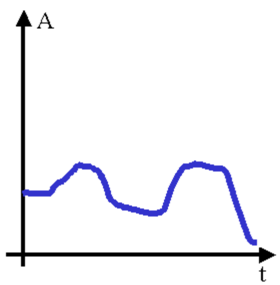
- Bode plots have been defined for continuous-time systems (e.g., analog filters)
- With z = $e^{i\omega}$ (see s. 26), the Z-transform degrades to a DTFT (as the Laplace transform was degrading to the FT with s = $i\omega$)
- We can therefore calculate the frequency response of the corresponding discrete-time system (e.g., a digital filter) with the transfer function $H(e^{i\omega})$
- A FFT is then typically used to calculate numerically the frequency response on computers
- Matlab has a dedicated function for this: freqz

Note: response in normalized angular frequency

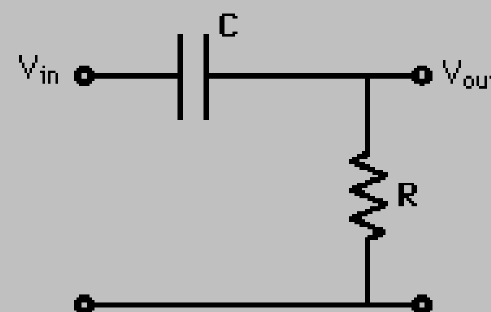
Digital Filters

Filters as System Examples

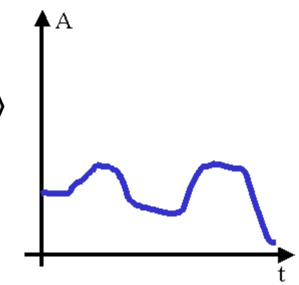
Analog



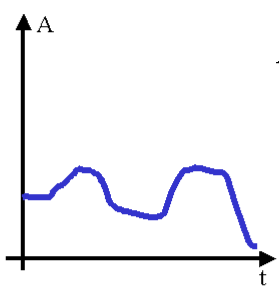
Circuit



Analog filter

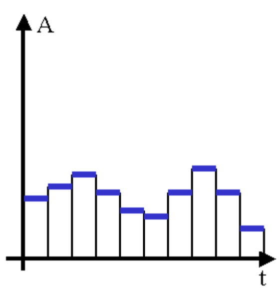


Analog



A/D

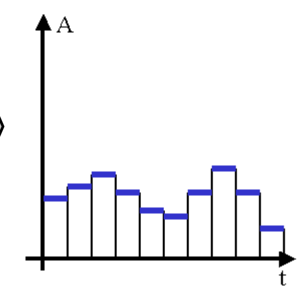
Digital



Function/algorithm

$$y_1 \ldots y_n = f(x_1 \ldots x_n)$$

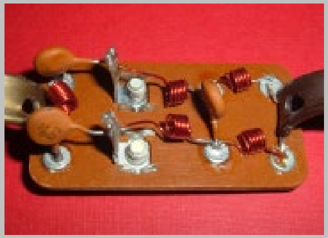
Digital filter



Transfer Functions of Filters

Analog

Circuit



Laplace Transf.

$$H(s) = A\prod \frac{\overbrace{(s-x_n)^{a_n}}^{\text{Numerator}}}{\underbrace{(s-y_n)^{b_n}}_{\text{Denominator}}}$$

Digital

Function/algorithm

$$y_1 \ldots y_n = f(x_1 \ldots x_n)$$

Z Transf.

$$H(z) = \frac{\overbrace{b_0 + b_1 z^{-1} + \ldots + b_N z^{-N}}^{\text{Numerator}}}{\underbrace{1 + a_1 z^{-1} + \ldots + b_M z^{-M}}_{\text{Denominator}}}$$

General Representation (Causal Filters)

Difference equation:

$$y[n] = -\sum_{k=1}^{N} a_k y[n-k] + \sum_{k=0}^{M} b_k x[n-k]$$

$$x[n] \longrightarrow \boxed{h[n]} \longrightarrow \mathrm{y}[n]$$

Z-transform:

$$Y(z) + \sum_{k=1}^{N} a_k Y(z) z^{-k} = \sum_{k=0}^{M} b_k X(z) z^{-k}$$

From s. 27, Z-transform properties:
$Z\{x[n-n_0]\} = z^{-n_0}\mathrm{X}(\mathrm{z})$

Transfer function:

$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{k=0}^{M} b_k z^{-k}}{1 + \sum_{k=1}^{N} a_k z^{-k}}$$

$$\mathrm{X}(z) \longrightarrow \boxed{\mathrm{H}(z)} \longrightarrow \mathrm{Y}(z)$$

An Example

Difference equation:

$$y[n] = x[n] + 2x[n-1] + x[n-2] - \frac{1}{4}y[n-1] + \frac{3}{8}y[n-2]$$

Z-transform:

$$Y(z) + \frac{1}{4}Y(z)z^{-1} - \frac{3}{8}Y(z)z^{-2} = X(z) + 2X(z)z^{-1} + X(z)z^{-2}$$

Transfer function:

$$H(z) = \frac{\mathrm{Y}(z)}{X(z)} = \frac{1 + 2z^{-1} + z^{-2}}{1 + \frac{1}{4}z^{-1} - \frac{3}{8}z^{-2}} = \frac{z^2 + 2z + 1}{z^2 + \frac{1}{4}z - \frac{3}{8}} = \frac{(z+1)^2}{\left(z - \frac{1}{2}\right)\left(z + \frac{3}{4}\right)}$$

Note: poles and zeros of H(z) can be represented in the zero-pole plot (complex z-plane) for further analysis (e.g., stability).

FIR Filters

Nonrecursive Digital Filters

- $a_k = 0$ for all k
- $y[n] = \sum_{k=0}^{M} b_k \, x[n-k]$
- Finite Impulse Response (FIR)
- The FIR filter above is a causal system: the output depends only on past values of inputs
- The filter coefficients b_k define the FIR filter
- The filter order is M, the number of coefficients b_k is M+1 (filter "length")

Examples of FIR Filters

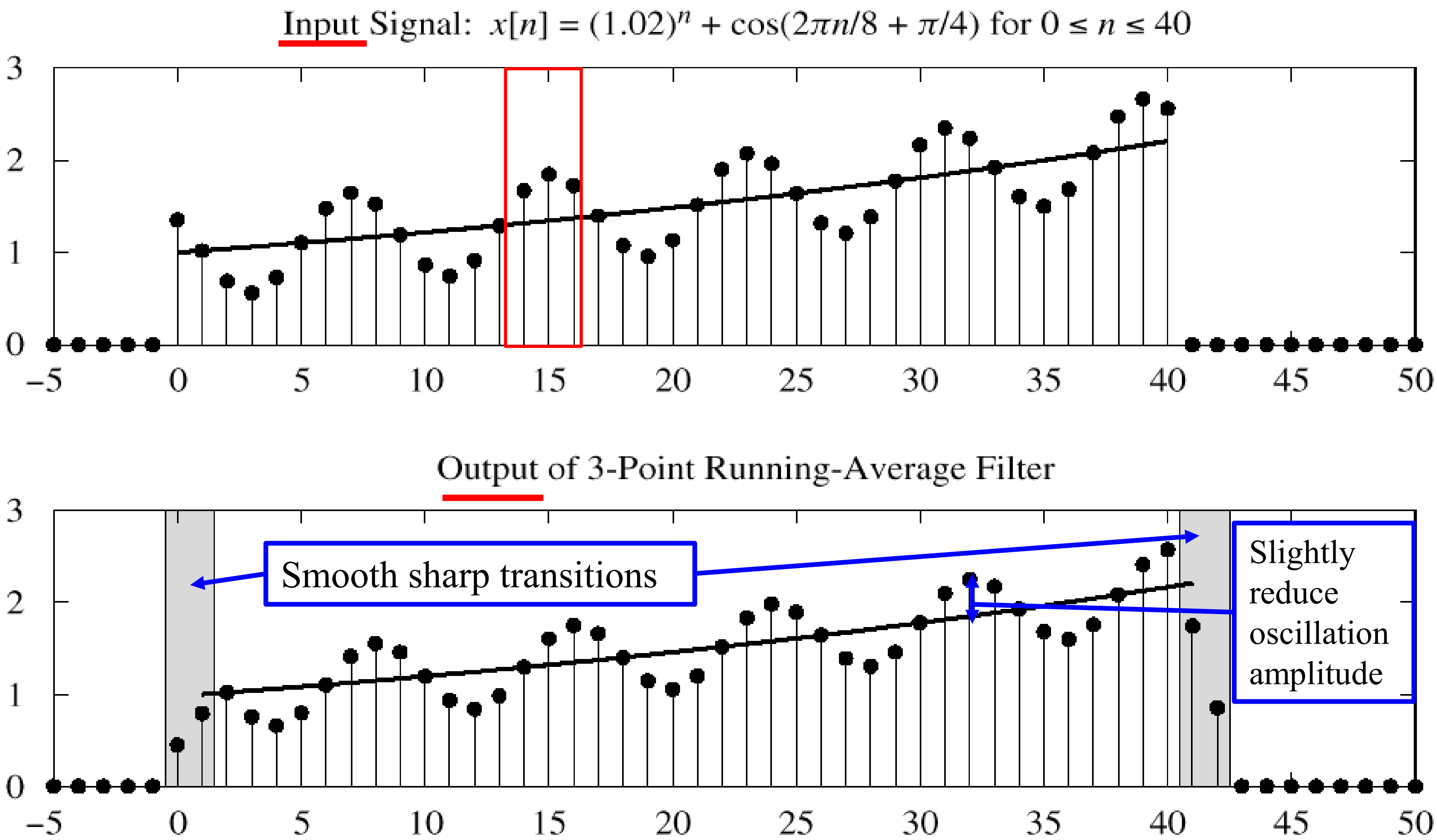
- 3-point moving average:

$$y_3[n] = \sum_{k=0}^{2} \frac{1}{3} x[n-k] = \frac{1}{3} x[n] + \frac{1}{3} x[n-1] + \frac{1}{3} x[n-2]$$

- 7-point moving average:

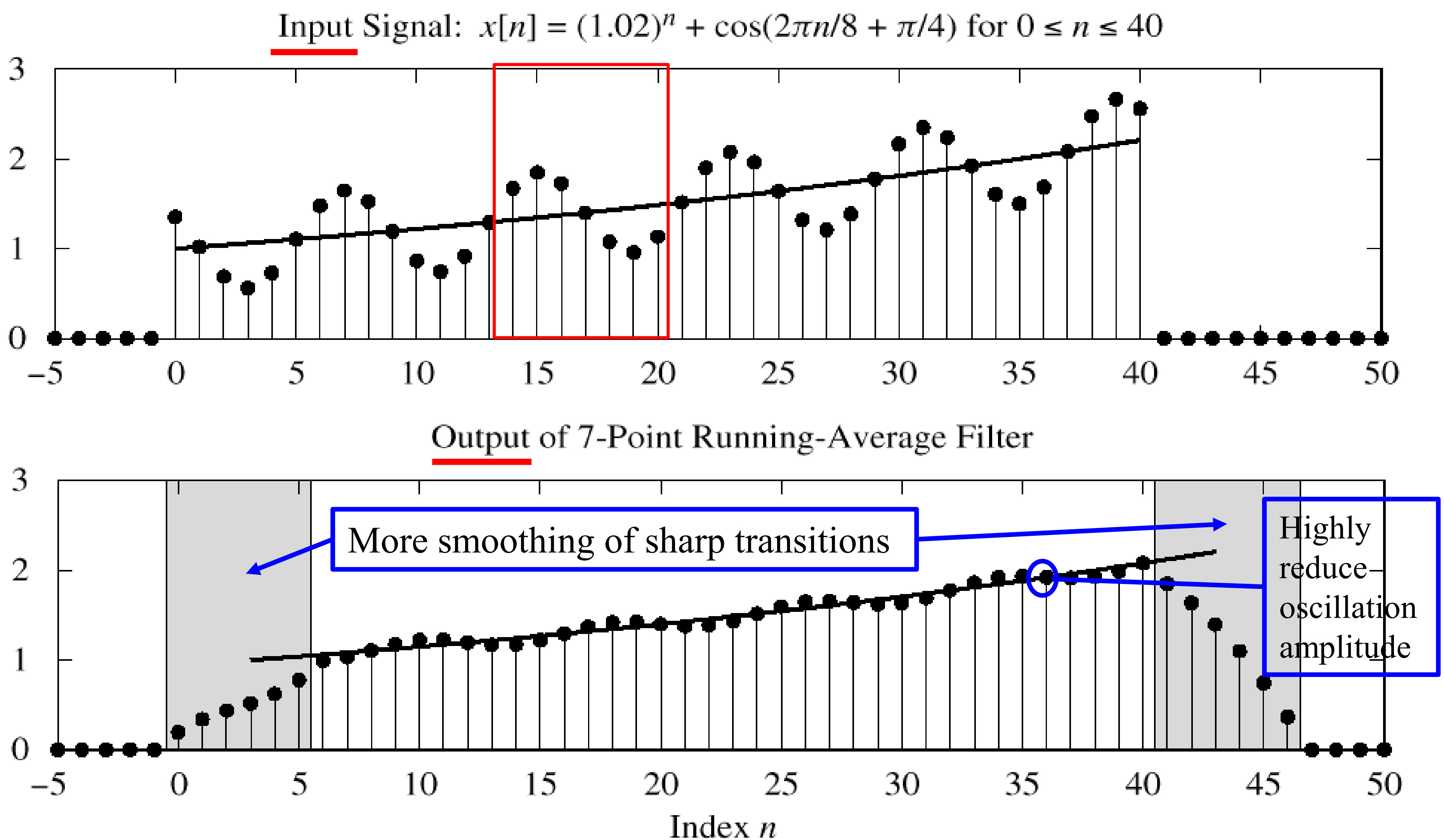
$$y_7[n] = \sum_{k=0}^{6} \frac{1}{7} x[n-k] = \frac{1}{7} x[n] + \frac{1}{7} x[n-1] + \cdots + \frac{1}{7} x[n-6]$$

3-pt Moving Average Example



[Picture from McClellan, Schafer, and Yoder, "DSP First: A Multimedia Approach"]

7-pt Moving Average Example



[Picture from McClellan, Schafer, and Yoder, “DSP First: A Multimedia Approach”]

Impulse Response of a Discrete-Time LTI System

$$\delta[n] \longrightarrow \boxed{\text{Discrete-Time LTI System}} \longrightarrow h[n]$$

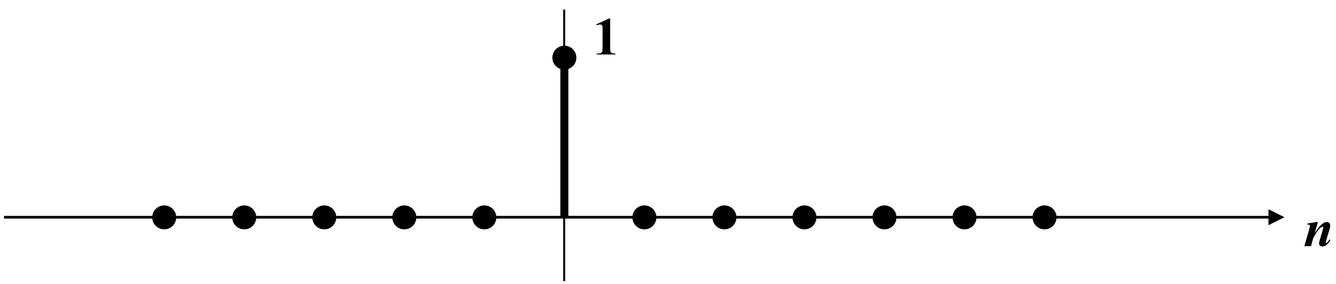
From W4, s. 16-17

$$f[n] \longrightarrow \boxed{h[n]} \longrightarrow (f * h)[n] = \sum_{m=-\infty}^{\infty} f[m]h[n-m]$$

The Unit Impulse

$$\delta[n] = \begin{cases} 1 & n = 0 \\ 0 & n \neq 0 \end{cases}$$

Notation:
Kronecker delta function



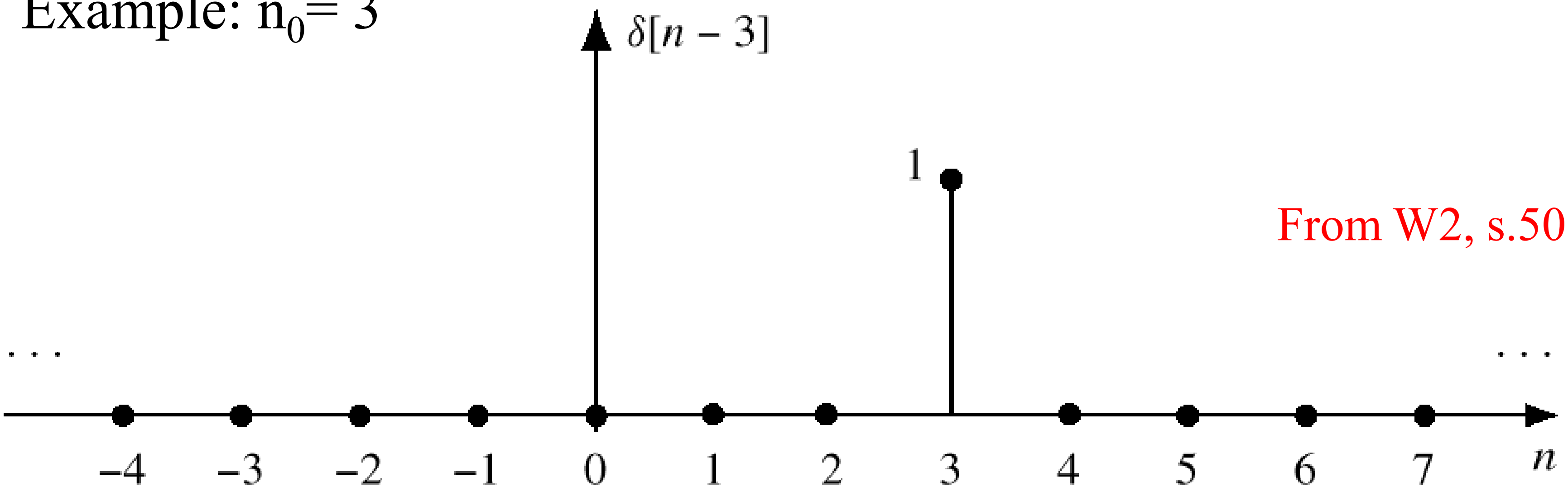
From W2, s.49

[Picture from McClellan, Schafer, and Yoder, "DSP First: A Multimedia Approach"]

Time-Shifted Unit Impulse

$$\delta[n-n_0] = \begin{cases} 1 & n = n_0 \\ 0 & n \neq n_0 \end{cases}$$

Example: $n_0 = 3$



From W2, s.50

[Picture from McClellan, Schafer, and Yoder, “DSP First: A Multimedia Approach”]

Impulse Response of a FIR Filter

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

A FIR filter usually described in terms of coefficients b_k

Alternatively, as any other LTI system, we can describe a FIR filter using its impulse response.

If $x[n] = \delta[n]$ then $\mathrm{y}[n] = h[n]$, i.e. the impulse response

$$\Longrightarrow \quad \boxed{h[n] = \sum_{k=0}^{M} b_k \delta[n-k]}$$

Note: you can check $(x * h)[n] = \mathrm{y}[\mathrm{n}]$

Impulse Response of a FIR Filter

$$h[n] = \sum_{k=0}^{M} b_k \delta[n-k]$$

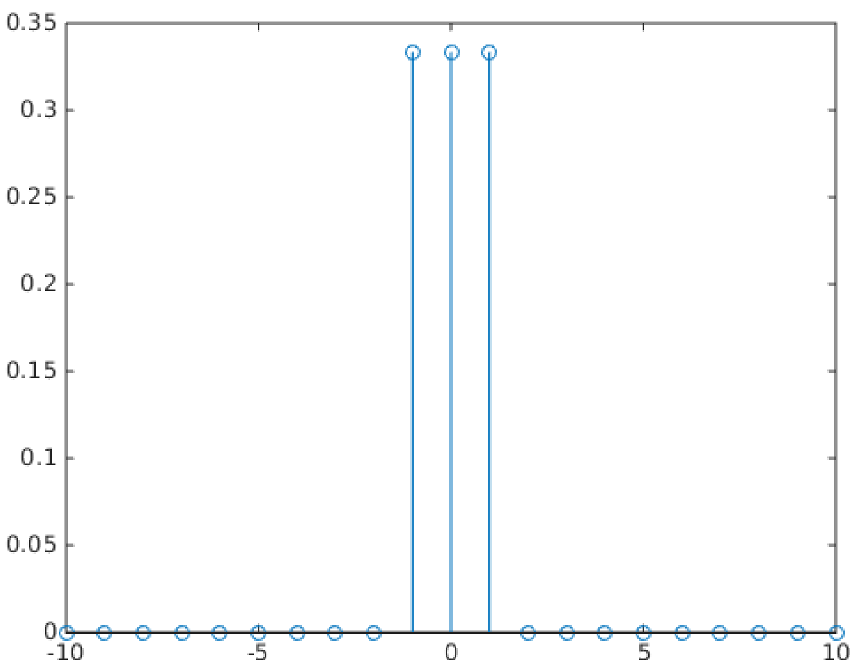
n	$n < 0$	0	1	2	3	$\dots$	M	$M+1$	$n > M+1$
$x[n] = \delta[n]$	0	$\boxed{1}$	0	0	0	0	0	0	0
$y[n] = h[n]$	0	b_0	b_1	b_2	b_3	$\dots$	b_M	0	0

Impulse response is finite!

[Table from McClellan, Schafer, and Yoder, "DSP First: A Multimedia Approach"]

Example of FIR Filter: Noncausal 3-Point Moving Average

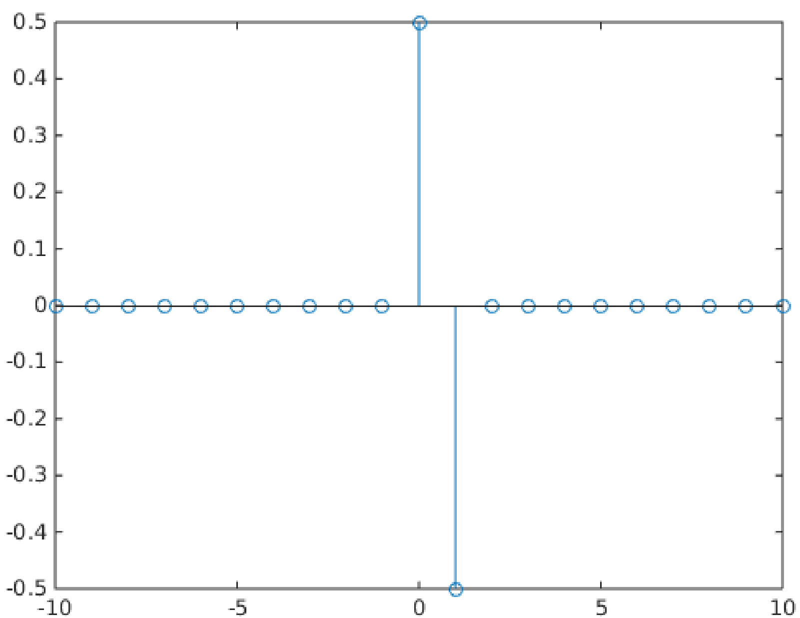
- $b_k = \{\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\}$ for k= -1,0,1 **Filter coefficients**
- $y[n] = \frac{1}{3}(x[n-1] + x[n] + x[n+1])$ **Difference equation**
- $h[n] = \frac{1}{3}(\delta[n-1] + \delta[n] + \delta[n+1])$ **Impulse response**



[Picture from Prof. A. S. Willsky, Signals and Systems course]

Example of FIR Filter: Causal High-Pass Filter

- $b_k = \{\frac{1}{2}, -\frac{1}{2}\}$ for k= 0,1 **Filter coefficients**
- $y[n] = \frac{1}{2}(x[n] - x[n-1])$ **Difference equation**
- $h[n] = \frac{1}{2}(\delta[n] - \delta\,[n-1])$ **Impulse response**



[Picture from Prof. A. S. Willsky, Signals and Systems course]

IIR Filters

Motivation

Can we remove the blur in postprocessing?
Yes! With a deconvolution filter!

[Picture from McClellan, Schafer, and Yoder, “DSP First: A Multimedia Approach”]

Motivation

Original Image

Blurred (Motion)

Restored w/ Inverse Filter

$$s[n] \longrightarrow \boxed{\begin{matrix} h_1[n] \\ \textbf{Conv} \end{matrix}} \xrightarrow{x[n]} \boxed{\begin{matrix} h_2[n] \\ \textbf{Deconv} \end{matrix}} \xrightarrow{y[n]}$$

Given $h_1[n]$, can we find $h_2[n]$ so that $y[n] = s[n]$?

$x[n] = s[n] * h_1[n]$

$y[n] = x[n] * h_2[n] = s[n] * h_1[n] * h_2[n]$

$\Rightarrow h_1[n] * h_2[n] = \delta[n]$ (convolution with unit impulse $\leftrightarrow$ identity)

[Adapted from McClellan, Schafer, and Yoder, “DSP First: A Multimedia Approach”]

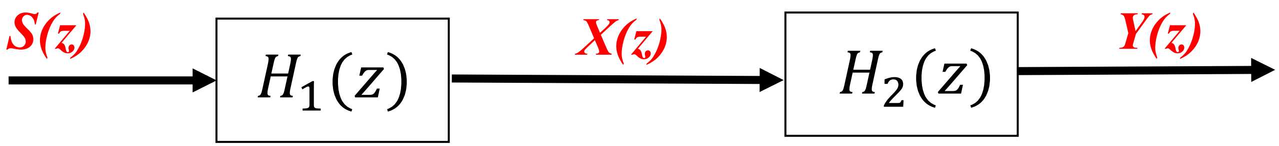
Motivation

Blurring filter with parameter a:

$$x[n] = s[n] - as[n-1] \Rightarrow h_1[n] = \delta[n] - a\delta[n-1]$$

Difficult to solve in time domain within convolution sum.

Z-domain:



$$Y(z) = H_2(z)H_1(z)S(z) = H(z)S(z)$$
$$H(z) = 1 = H_2(z)H_1(z)$$
$$\Rightarrow H_2(z) = 1/H_1(z)$$

Blurring filter in z-domain (from DE above): $H_1(z) = 1 - az^{-1}$

$$\Rightarrow H_2(z) = \frac{1}{1 - az^{-1}}$$

[Adapted from McClellan, Schafer, and Yoder, "DSP First: A Multimedia Approach"]

Motivation

$$H_2(z) = \frac{1}{1 - az^{-1}}$$

What type of filter?

$$H_2(z) = \frac{Y(z)}{X(z)} = \frac{1}{1 - az^{-1}}$$

$$Y(z) = aY(z)z^{-1} + X(z)$$

$$y[n] = ay[n-1] + x[n]$$

If a $\neq 0$ not a FIR (see s. 42)!

In fact H_2 is a first order IIR filter!

Recursive Digital Filters

- $y[n] = -\sum_{k=1}^{N} a_k\, y[n-k] + \sum_{k=0}^{M} b_k\, x[n-k]$
- $a_k \neq 0$ and $b_k \neq 0$ at least for one k
- a_k: feedback coefficients
- b_k: feed-forward coefficients
- Infinite Impulse Response (IIR)
- The IIR filter above is a causal system: the output depends only on past values of output and input
- The filter order is typically N and the total number of coefficients is N+M+1

Impulse Response of a IIR Filter

Example: generalized first order IIR filter:

$$y[n] = a_1 y[n-1] + b_0 x[n] \Rightarrow h[n] = a_1 h[n-1] + b_0 \delta[n]$$

n	$n<0$	0	1	2	3	4	$\cdots$
$\delta[n]$	0	$\boxed{1}$	0	0	0	0	$\cdots$
$h[n-1]$	0	0	b_0	$b_0(a_1)$	$b_0(a_1)^2$	$b_0(a_1)^3$	$\cdots$
$h[n]$	0	b_0	$b_0(a_1)$	$b_0(a_1)^2$	$b_0(a_1)^3$	$b_0(a_1)^4$	$\cdots$

Impulse response is infinite!

Types and Order of Digital Filters

Digital Filter Type and Order

- Several digital filters types and design methods exist and are defined by the resulting coefficient distributions of the filter
- Order:

$y[n] = b_0 x[n]$	Filter order: 0
$y[n] = b_0 x[n] + b_1 x[n-1]$	Filter order: 1
$y[n] = b_0 x[n] + b_1 x[n-1] + b_2 x[n-2]$	Filter order: 2

$\ldots$

- Filter with a high order are closer to the ideal filter (rectangular function) but are computationally more expensive

Conclusion

Take Home Messages

- Bode plots reproduce an estimated frequency response of (cascaded) LTI; they have been developed for continuous-time systems and expanded to time-discrete systems
- The Z-transform is the dual of the Laplace transform for time-discrete systems
- Both analog and digital filters are:
  – characterized by different order and coefficient distributions
  – they can be expressed in time and (complex) frequency domains
- Several equivalent forms to define digital filters are possible:
  - time domain: coefficient set, difference equation, impulse response
  - frequency domain: transfer function
- The impulse response of a digital filter which can be finite (filter non recursive) or infinite (filter recursive); FIR and IIR filters are correspondingly defined

Additional Literature – Week 5

Books

- J. H. McClellan, R. W. Schafer, M. A. Yoder "DSP First: A Multimedia Approach", Prentice Hall, 1999.
- A. Oppenheim and A. S. Willsky with S. Nawab, "Signals and Systems", Prentice Hall, 1997.