Chapter 9 IIR Digital Filter Design

Zhiliang Liu

Zhiliang_Liu@uestc.edu.cn

9.1 Preliminary Considerations

9.1.1 Digital Filter Specifications

Just like analog LPF, it is considered

$$1-\delta_P \le \left|G(e^{j\omega})\right| \le 1+\delta_P, \ for \ |\omega| \le \omega_P \ \cdots (9.1)$$

$$\left|G(e^{j\omega})\right| \le \delta_S, \ for \ \omega_S \le |\omega| \le \pi \ \cdots (9.2)$$

Passband edge frequency : ω_P

Stopband edge frequency : ω_S

9.1.1 Digital Filter Specifications

ripples **:** $\boldsymbol{\delta_P}$ and $\boldsymbol{\delta_S}$

peak passband ripple **:**

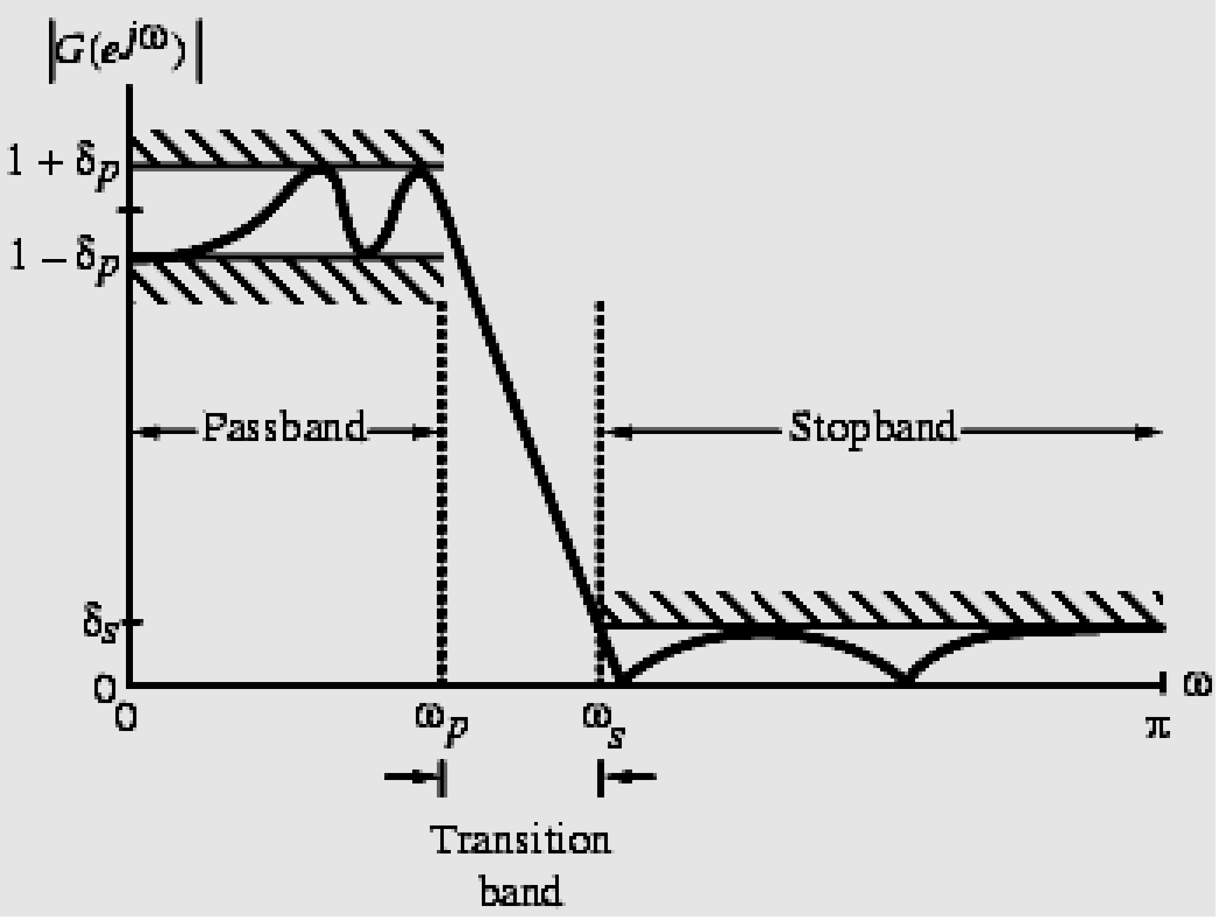
$$\alpha_p = -20\log_{10}(1-\delta_p)[dB]\cdots(9.3)$$

minimum stopband attenuation **:**

$$\alpha_S = -20\log_{10}(\delta_s)[dB]\cdots(9.4)$$

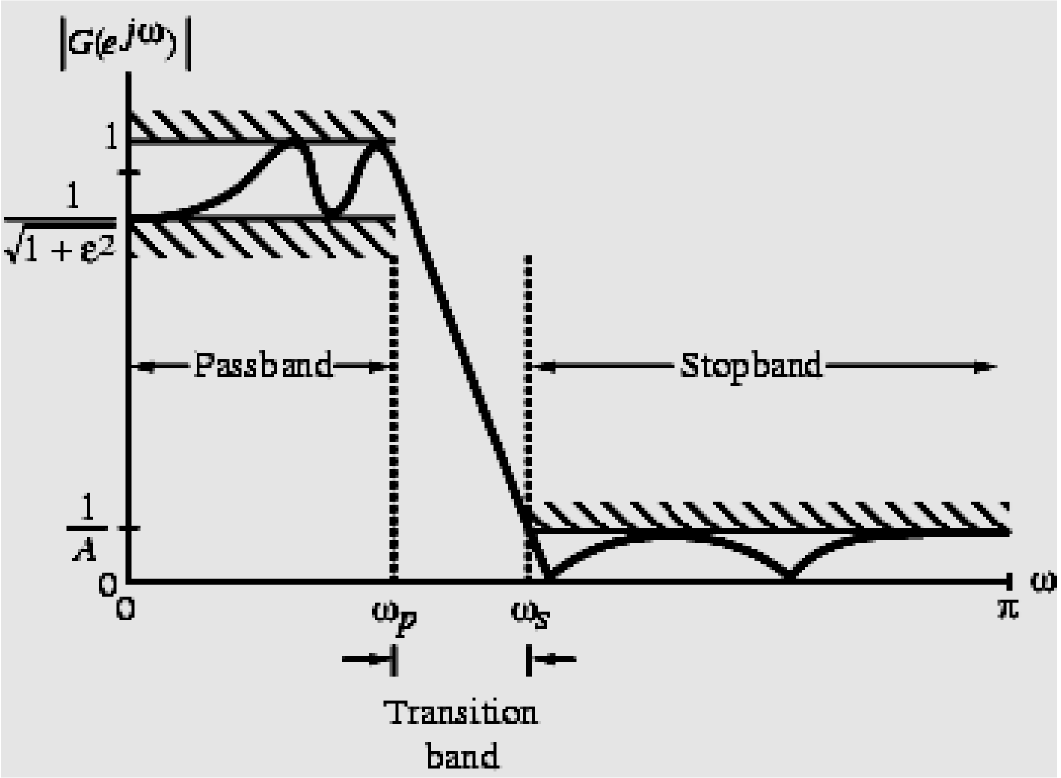
- **Specifications are often given in terms of loss function $A(\omega) = -20\log_{10}|G(e^{j\omega})|$ in dB**

9.1.1 Digital Filter Specifications



9.1.1 Digital Filter Specifications

- **Magnitude specifications may alternately be given in a normalized form as indicated below**



9.1.1 Digital Filter Specifications

- **Here, the maximum value of the magnitude in the passband is assumed to be unity.**

- **$1/\sqrt{(1+\varepsilon^2)}$ - Maximum passband deviation, given by the minimum value of the magnitude in the passband**

- **$1/A$ - Maximum stopband magnitude**

9.1.1 Digital Filter Specifications

- **For the normalized specification, maximum value of the gain function or the minimum value of the loss function is 0 dB**
- **Maximum passband attenuation**

$$\alpha_{\max} = 20 \log_{10}\left(\sqrt{1+\varepsilon^2}\right) \ dB$$

- **For δ_p<<1, it can be shown that**

$$\alpha_{\max} \cong -20 \log_{10}(1-2\,\delta_p) \ \ dB$$

9.1.1 Digital Filter Specifications

- **In practice, passband edge frequency F_p and stopband edge frequency F_s are specified in Hz**
- **For digital filter design, normalized band edge frequencies need to be computed from specifications in Hz using**

$$\omega_p = \frac{\Omega_p}{F_T} = \frac{2\pi F_p}{F_T} = 2\pi F_p T$$

$$\omega_s = \frac{\Omega_s}{F_T} = \frac{2\pi F_s}{F_T} = 2\pi F_s T$$

9.1.1 Digital Filter Specifications

- ***Example*:** Let F_p = 7 kHz, F_s= 3 kHz, and F_T = 25 kHz
- **Then**

$$\omega_p = \frac{2\pi(7\times10^3)}{25\times10^3} = 0.56\pi$$

$$\omega_s = \frac{2\pi(3\times10^3)}{25\times10^3} = 0.24\pi$$

9.1.2 Selection of the Filter Type

The *objective* of digital filter design is to develop a casual transfer function *H*(*z*) meeting the frequency response specifications.

Whether IIR or FIR should be selected?

IIR:

low order to reduce computational complexity, but must be stable.

$$H(z)=\frac{\sum_{k=0}^{M} p_k z^{-k}}{\sum_{k=0}^{N} d_k z^{-k}}$$

FIR:

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

with linear phase ,but high order.

9.1.3Basic Approaches to IIR Digital Filter Design

IIR design

The most common practice is to transform $H_a(s)$ into the desired digital transfer function $G(z)$.

convert the digital filter specifications to into analog lowpass prototype filter specifications

↓

determine the analog lowpass filter transfer function meeting these specifications

↓

transform the analog filter into the desired digital transfer function

9.1.3Basic Approaches to IIR Digital Filter Design

This approach has been widely used for many reasons:

(a) Analog approximation techniques are highly advanced.

(b) They usually yield closed-form solutions.

(c) Extensive tables are available for analog filter design.

(d) Many applications require the digital simulation of analog filters.

9.1.3Basic Approaches to IIR Digital Filter Design

The basic idea is to apply a mapping from *s*-domain to the *z*-domain so that the essential properties of the analog frequency response are preserved.

This implies that the mapping function should be such that

(1) The imaginary axis in the *s*-plane be mapped onto the unit circle of the *z*-plane.
(2) A stable analog transfer function be transferred into a stable digital transfer function.

9.1.4 Estimation of the Filter Order

For the design of IIR LPF, the order of $H_a(s)$ is estimated from its specifications using the appropriate formula given in Eq. (4.35), (4.43), or (4.54), depending on which approximation desired. the order of $G(z)$ is determined automatically from the transformation being used to convert $H_a(s)$ into $G(z)$.

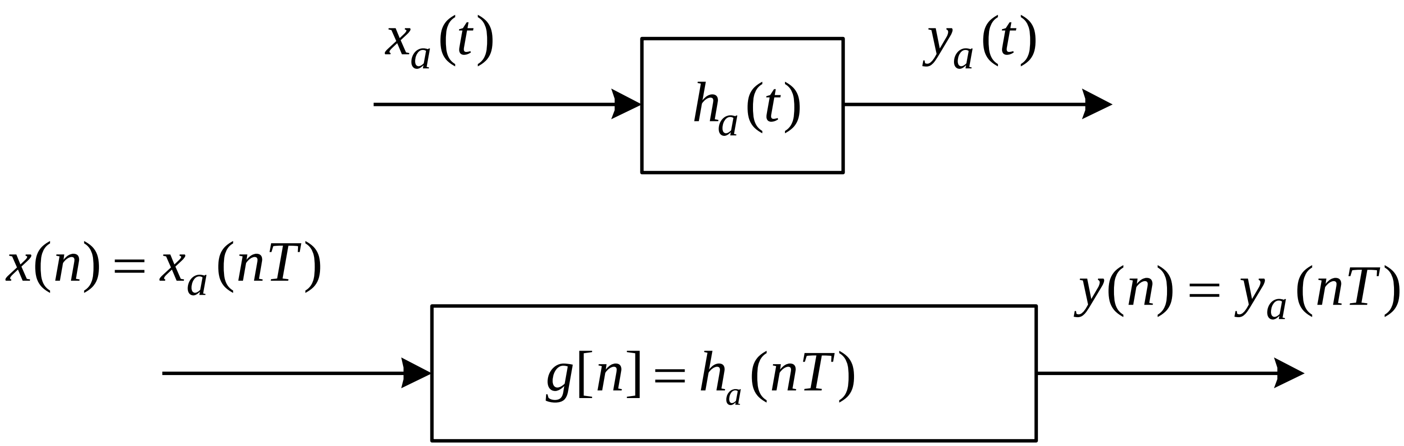
The scaled transfer function $G_t(z) = KG(z)$, $K = 1/G_{\max}(z_0)$ is BR function.

9.2 The Design Methods of IIR Filter

9.2.1 The Impulse Invariance Method

1. The Main Idea (Problem 9.6)

It is desired $g[n] = h_a(nT)$.If $h_a(t) \leftrightarrow H_a(s)$, $g[n] \leftrightarrow G(z)$. From the relation between LT and ZT, we know that



9.2.1 The Impulse Invariance Method

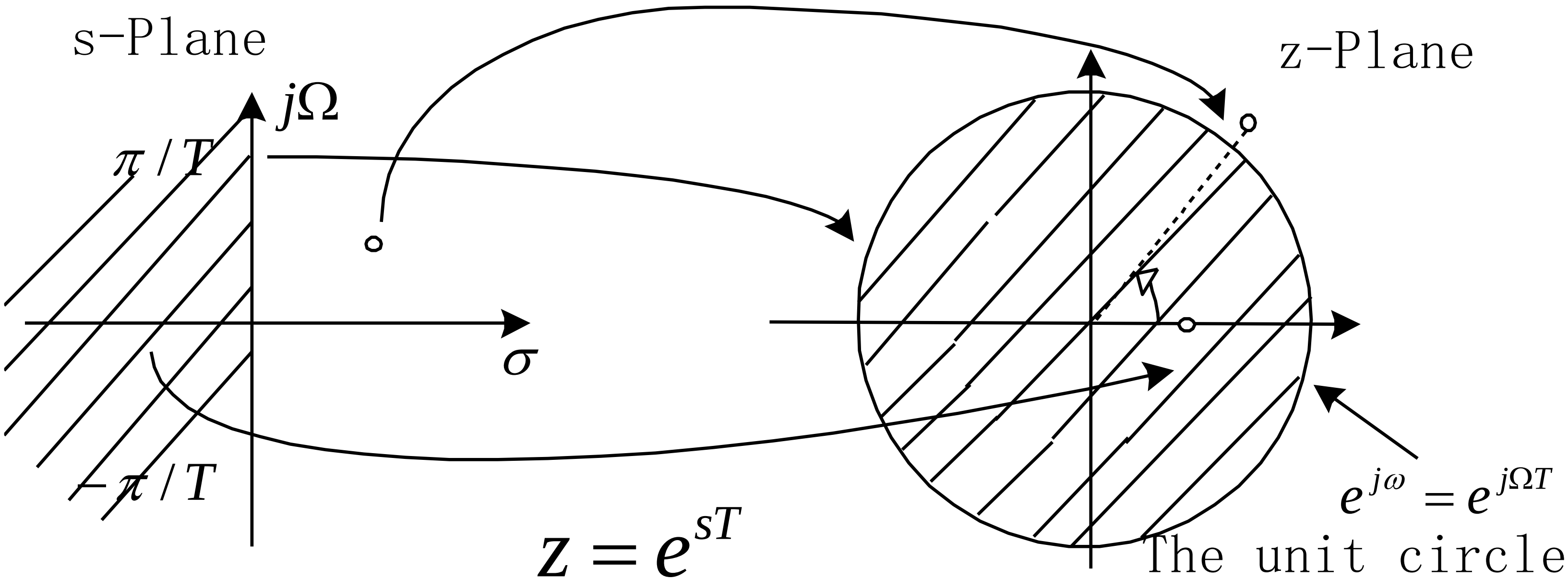
The frequency response

$$G(z)\big|_{z=e^{sT}} = \frac{1}{T}\sum_{m=-\infty}^{\infty} H_a\left(s - jm\frac{2\pi}{T}\right)$$

The mapping function is $z = e^{sT}$

or $s = \dfrac{1}{T}\ln z$.

9.2.1 The Impulse Invariance Method



Discussion: (1) The mapping satisfies the essential properties (***j*****Ω-axis mapped onto the unit circle, stable region in** ***s*****-plane mapped into the stable region in** ***z*****-plane).**

9.2.1 The Impulse Invariance Method

(2) The exact relation between $j\Omega$-axis in the s-plane and the unit circle in z-plane is $\omega=\Omega T$.

(3) When Nyquist theoriem is satisfied,

$$G(e^{j\omega}) = \frac{1}{T} H_a(j\Omega), (|\omega| < \pi)$$

If $H_a(j\Omega)$ is not bandlimited, $G(e^{j\omega})$ obtained by this method is overlapped in frequency-domain.

9.2.1 The Impulse Invariance Method

2. The Design Steps

(1) $H_a(s)$ is partial-fractional expressed as: ($h_a(t)$ is derived)

$$H_a(s) = \sum_{i=1}^{N} \frac{A_i}{s - s_i}, \ (\mathrm{Re}(s_i)_{\max} < 0)$$

(2) From $g[n] = h_a(nT)$, we can get:

$$G_1(z) = \sum_{i=1}^{N} \frac{A_i}{1 - e^{s_i T} z^{-1}}$$

9.2.1 The Impulse Invariance Method

(3) Usually, we scale *T* to *G*(*z*) and get:

$$G(z) = \sum_{i=1}^{N} \frac{TA_i}{1 - e^{s_i T} z^{-1}}$$

***Proof*: Because**

$$h_a(t) = L^{-1}\{H_a(s)\} = \sum_{i=1}^{N} A_i e^{s_i t} \mu(t)$$

$$g[n] = h_a(nT) = \sum_{i=1}^{N} A_i e^{s_i nT} \mu[n]$$

$$G_1(z) = \sum_{i=1}^{N} \frac{A_i}{1 - e^{s_i T} z^{-1}}$$

9.2.1 The Impulse Invariance Method

Example **If** $H_a(s) = \dfrac{2}{s^2 + 3s + 2}$ **,the sampling period is *T*. Then**

$$H_a(s) = \frac{2}{s+1} - \frac{2}{s+2}$$

The poles: $s_1 = -1$, $s_2 = -2$.

Using the impulse invariance method, we get:

$$G(z) = T\frac{2}{1-e^{-T}z^{-1}} - T\frac{2}{1-e^{-2T}z^{-1}}$$

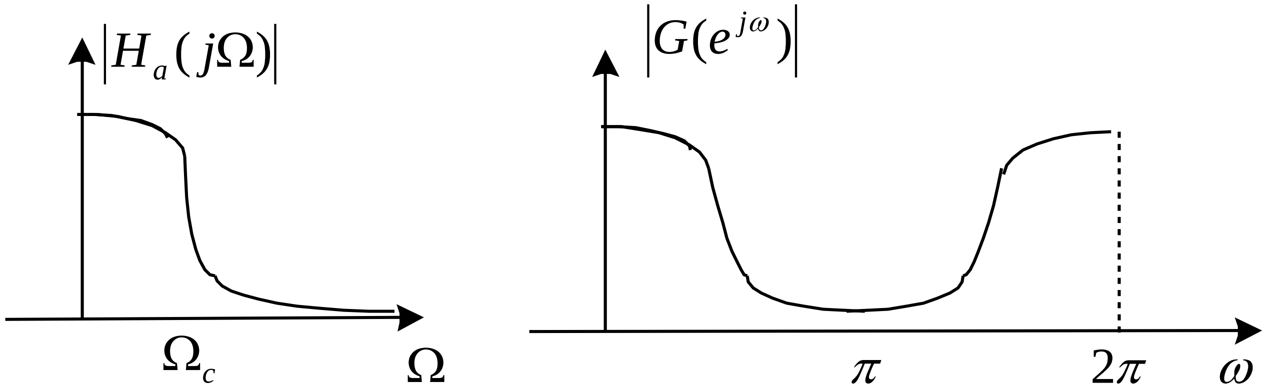
$$= \frac{2T(e^{-T} - e^{-2T})z^{-1}}{1-(e^{-2T} + e^{-T})z^{-1} + e^{-3T}z^{-2}}$$

9.2.1 The Impulse Invariance Method

When $T = 1$,

$$G(z) = \frac{0.4651\, z^{-1}}{1 - 0.5032\, z^{-1} + 0.04979\, z^{-2}}$$

Note: $G(e^{j\omega})$ **is overlapped in the frequency-domain.**



9.2.2 The Bilinear Translation Method

The Bilinear Transformation

The mapping from *s*-plane to *z*-plane:

$$s = \frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right) \cdots (9.14)$$

The relation between the digital transfer function and the parent analog transfer function:

$$G(z) = H_a(s)\Big|_{s=\frac{2}{T}\left(\frac{1-z^{-1}}{1+z^{-1}}\right)} \cdots (9.15)$$

9.2.2 The Bilinear Translation Method

- **Digital filter design consists of 3 steps:**

  (1) Develop the specifications of $H_a(s)$ by applying the inverse bilinear transformation to specifications of $G(z)$

  (2) Design $H_a(s)$

  (3) Determine $G(z)$ by applying bilinear transformation to $H_a(s)$

- **As a result, the parameter T has no effect on $G(z)$ and $T = 2$ is chosen for convenience**

9.2.2 The Bilinear Translation Method

- **Inverse bilinear transformation for *T*=2 is**

$$z = \frac{1+s}{1-s}$$

For $s = \sigma_0 + j\Omega_0$

$$z = \frac{(1+\sigma_0) + j\Omega_0}{(1-\sigma_0) - j\Omega_0} \Rightarrow |z|^2 = \frac{(1+\sigma_0)^2 + \Omega_0^2}{(1-\sigma_0)^2 + \Omega_0^2}$$

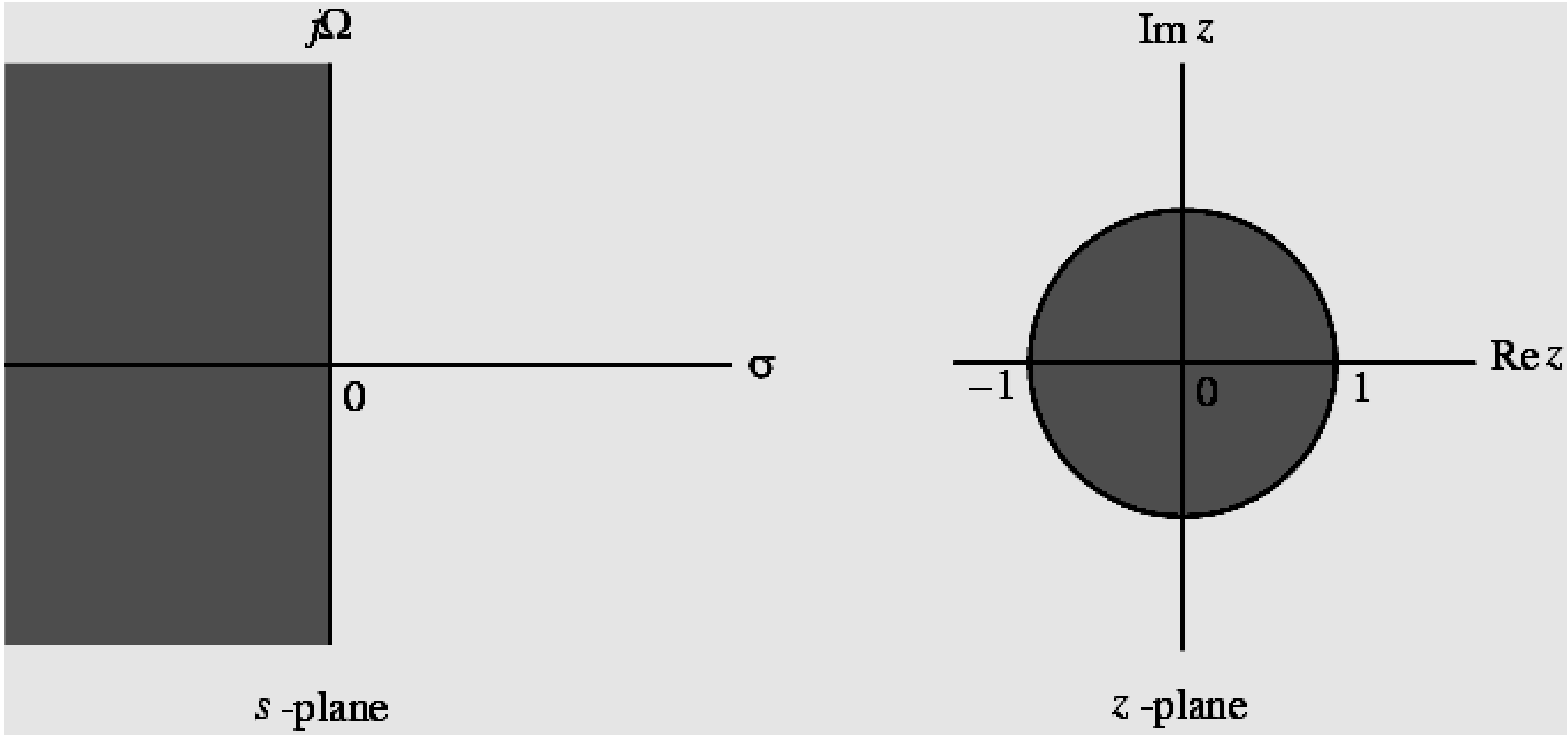
Thus, $\sigma_0 = 0 \rightarrow |z| = 1$

$\sigma_0 < 0 \rightarrow |z| < 1$

$\sigma_0 > 0 \rightarrow |z| > 1$

9.2.2 The Bilinear Translation Method

- **Mapping of *s*-plane into the *z*-plane**



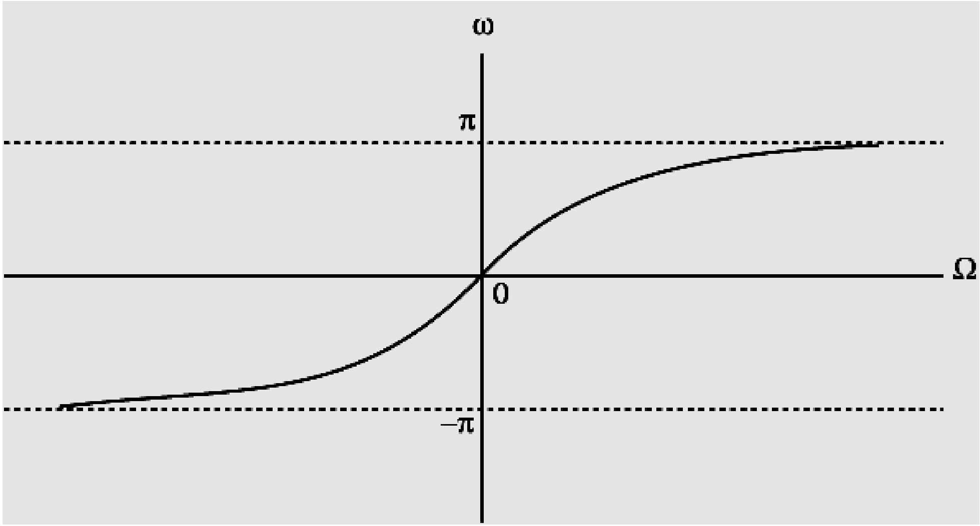
9.2.2 The Bilinear Translation Method

- **For $z = e^{j\omega}$ with $T = 2$ we have**

$$j\Omega = \frac{1-e^{-j\omega}}{1+e^{-j\omega}} = \frac{e^{-j\omega/2}(e^{j\omega/2}-e^{-j\omega/2})}{e^{-j\omega/2}(e^{j\omega/2}+e^{-j\omega/2})}$$

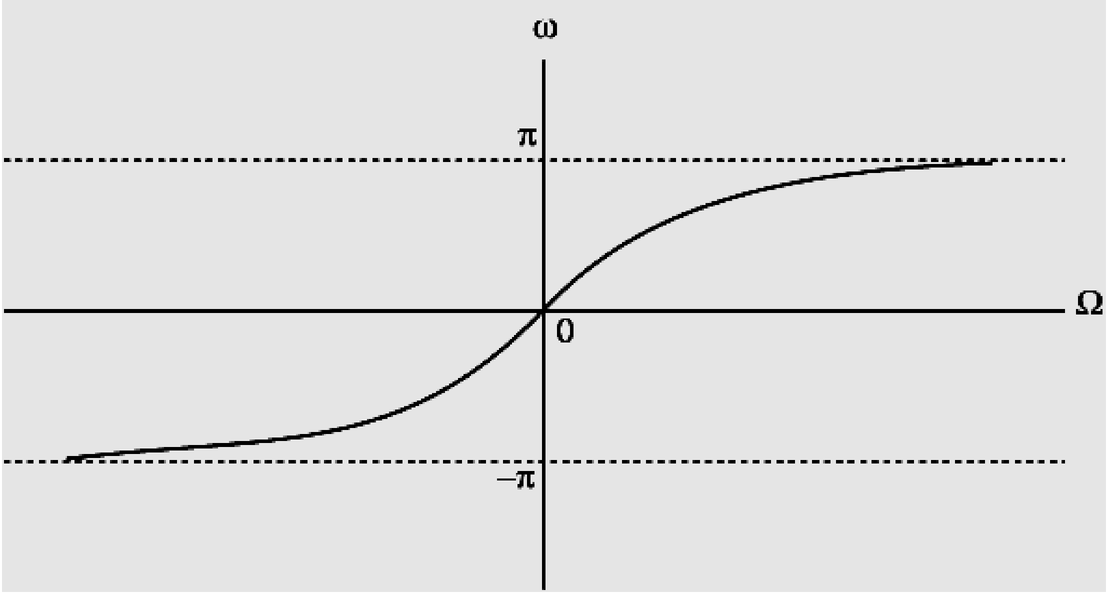
$$= \frac{j2\sin(\omega/2)}{2\cos(\omega/2)} = j\tan(\omega/2)$$

or $\Omega = \tan(\omega/2)$



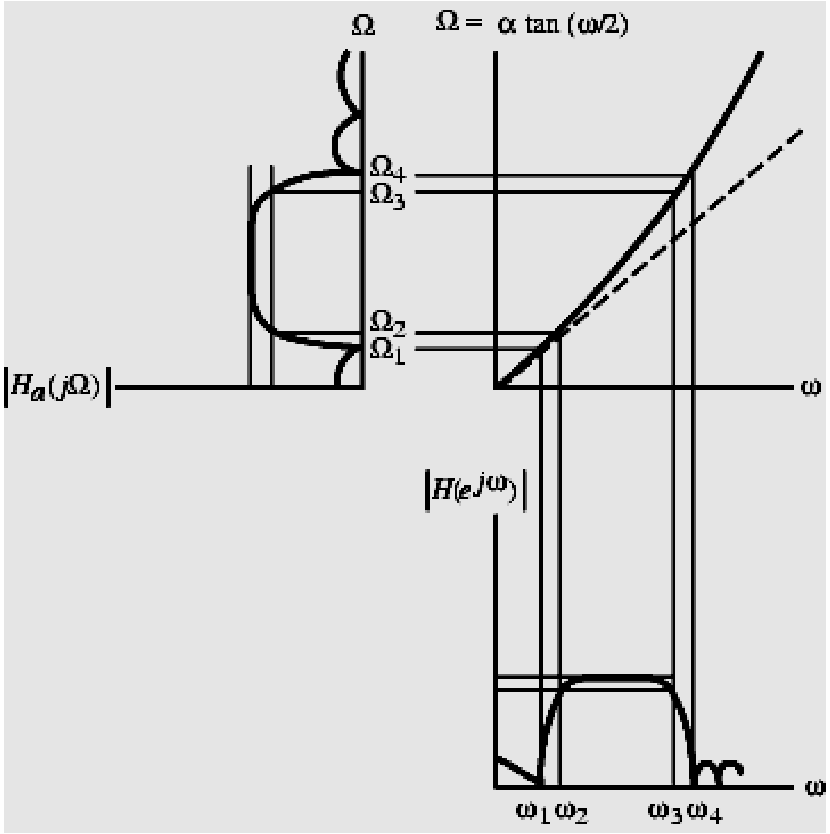
The exact relation between $j\Omega$-axis in the s-plane and the unit circle in z-plane (from Eq. (9.18) with T = 2):

$$\Omega = \tan(\frac{\omega}{2}) \cdots (9.18)$$



This introduces a distortion the frequency axis called *frequency warping*.

Thus, to develop a digital filter meeting a specified magnitude, we must *first prewarp* the critical bandedge frequencies(ω_P and ω_S) to find their analog Equivalents (Ω_S and Ω_P) using Eq.(9.18).



9.2.2 The Bilinear Translation Method

- **Steps in the design of a digital filter**

  (1) Prewarp (ω_p, ω_s) to find their analog equivalents (Ω_p, Ω_s)

  (2) Design the analog filter $H_a(s)$

  (3) Design the digital filter $G(z)$ by applying bilinear transformation to $H_a(s)$

- **Transformation can be used only to design digital filters with prescribed magnitude response with piecewise constant values**

- **Transformation does not preserve phase response of analog filter**

IIR Digital Filter Design Using Bilinear Transformation

- ***Example*** **- Consider**

$$H_a(s) = \frac{\Omega_c}{s + \Omega_c}$$

Applying bilinear transformation to the above we get the transfer function of a first-order digital lowpass Butterworth filter

$$G(z) = H_a(s)\Big|_{s=\frac{1-z^{-1}}{1+z^{-1}}} = \frac{\Omega_c(1+z^{-1})}{(1-z^{-1}) + \Omega_c(1+z^{-1})}$$

IIR Digital Filter Design Using Bilinear Transformation

- **Rearranging terms we get**

$$G(z) = \frac{1-\alpha}{2} \cdot \frac{1+z^{-1}}{1-\alpha z^{-1}}$$

where

$$\alpha = \frac{1-\Omega_c}{1+\Omega_c} = \frac{1-\tan(\omega_c/2)}{1+\tan(\omega_c/2)}$$

IIR Digital Filter Design Using Bilinear Transformation

- ***Example*** **- Consider the second-order analog notch transfer function**

$$H_a(s) = \frac{s^2 + \Omega_o^2}{s^2 + Bs + \Omega_o^2}$$

for which $|H_a(j\Omega_0)| = 0$

$$|H_a(j0)| = |H_a(j\infty)| = 1$$

- **Ω_0 is called the notch frequency**
- **If $|H_a(j\Omega_2)| = |H_a(j\Omega_1)| = 1/\sqrt{2}$ then $B = \Omega_2 - \Omega_1$ is the 3-dB notch bandwidth**

IIR Digital Filter Design Using Bilinear Transformation

- **Then**

$$G(z) = H_a(s)\Big|_{s=\frac{1-z^{-1}}{1+z^{-1}}}$$

$$= \frac{(1+\Omega_o^2) - 2(1-\Omega_o^2)\,z^{-1} + (1+\Omega_o^2)\,z^{-2}}{(1+\Omega_o^2 + B) - 2(1-\Omega_o^2)\,z^{-1} + (1+\Omega_o^2 - B)\,z^{-2}}$$

$$= \frac{1+\alpha}{2} \cdot \frac{1 - 2\beta\, z^{-1} + z^{-2}}{1 - \beta\,(1+\alpha) z^{-1} + \alpha\, z^{-2}}$$

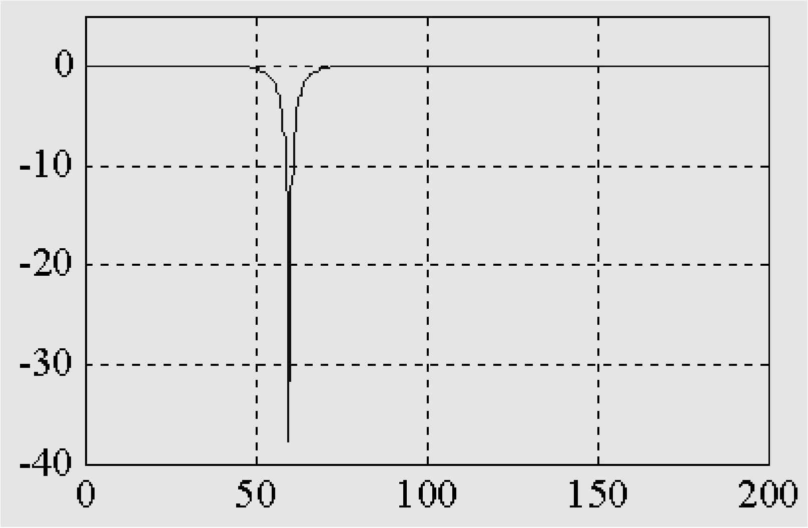
where

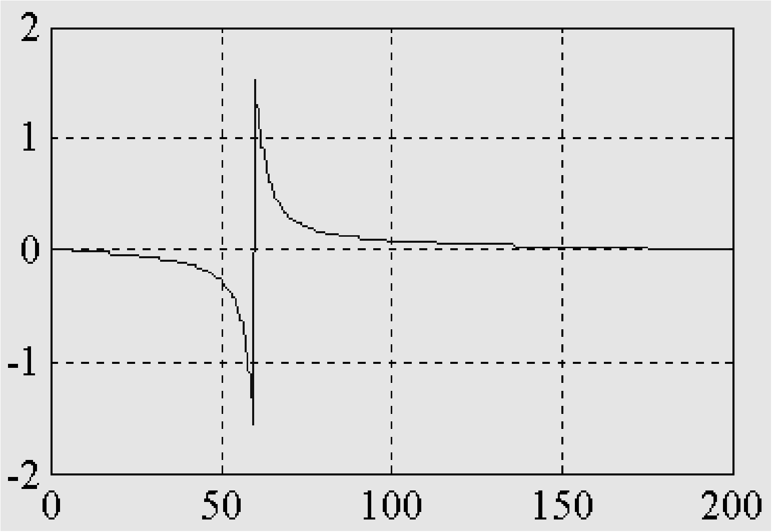
$$\alpha = \frac{1+\Omega_o^2 - B}{1+\Omega_o^2 + B} = \frac{1-\tan(B_w/2)}{1+\tan(B_w/2)},\ \beta = \frac{1-\Omega_o^2}{1+\Omega_o^2} = \cos\omega_o$$

Example

$$G(z)=\frac{0.954965-1.1226287z^{-1}+0.954965z^{-2}}{1-1.1226287z^{-1}+0.90993z^{-2}}$$

The gain and phase responses are shown below





9.3 Design of Lowpass IIR Digital Filters

Consider $G(z)$ with a <u>maximally flat</u> magnitude, and a pass ripple not exceeding 0.5dB,

$$\omega_P = 0.25\pi, \omega_S = 0.55\pi$$

and the minimum stopband attenuation 15dB.

$$20\log_{10}\left|G(e^{j0.25\pi})\right| \geq -0.5[dB] \qquad \textbf{…(9.34a)}$$

$$20\log_{10}\left|G(e^{j0.55\pi})\right| \leq -15[dB] \qquad \textbf{…(9.34b)}$$

1. The Bilinear Translation Method

(1) Prewarpping

$$\Omega_P = \tan(\frac{\omega_P}{2}) = \tan(\frac{0.25\pi}{2}) = 0.4142136$$

$$\Omega_S = \tan(\frac{\omega_S}{2}) = \tan(\frac{0.55\pi}{2}) = 1.1708496$$

(2) Design the parent analog filter $H_a(s)$

From the specifications we obtain

pass ripple 0.5dB $\Rightarrow$ $\varepsilon^2 = 0.1220185$

stopband attenuation 15dB $\Rightarrow$ $A^2 = 31.622777$

Using Eqs. (4.31), (4.32),

$$k = \frac{\Omega_s}{\Omega_p} = \frac{1.1708496}{0.4142136} = 2.8266809$$

$$k_1 = \frac{\varepsilon}{\sqrt{A^2 - 1}} = 15.841979$$

Using Eq. (4.35).

$$N = \frac{\log_{10}(1/k_1)}{\log_{10}(1/k)} = \frac{\log_{10}(15.841979)}{\log_{10}(2.8266814)} = 2.6586997$$

The least order of Butterworth LPF is *N* = 3.

Using Eq. (4.34a) we get the 3-dB frequency

$$\Omega_c = (\varepsilon)^{-1/N}\Omega_P = 1.419915(\Omega_P) = 0.588148$$

third-order normalized lowpass Butterworth transfer function

$$H_{an}(s) = \frac{1}{(s+1)(s^2+s+1)} = \frac{1}{1+2s+2s^2+s^3}$$

which has 3-dB frequency at Ω_c = 1.

Therefore, the parent transfer function

$$H_a(s) = H_{an}\left(\frac{s}{\Omega_c}\right) = \frac{0.203451}{(s+0.588148)(s^2+0.588148s+0.345918)}$$

(3) Design the digital filter $G(z)$ by applying bilinear transformation to $H_A(s)$

$$G(z) = H_A(s)\,\Big|_{s=\frac{1-z^{-1}}{1+z^{-1}}}$$

$$G(z) = \frac{0.0662272\,(1+z^{-1})^3}{(1-0.2593284\,z^{-1})(1-0.6762858\,z^{-1}+0.3917468\,z^{-2})}$$

…………….(9.35)

Note: If this digital filter is used to process an analog signal, it has the equivalent passband-edge frequency at

$$f_p = \frac{\omega_P}{2\pi} \times f_T = 10 \text{ kHz}$$

when the sampling frequency is f_T = 80 kHz.

2. The Impulse Invariance Method

The third-order lowpass Butterworth transfer function which has 3-dB frequency at Ω_c.

$$H_a(s) = H_{an}\left(\frac{s}{\Omega_c}\right) = \frac{1}{1 + 2\frac{s}{\Omega_c} + 2\left(\frac{s}{\Omega_c}\right)^2 + \left(\frac{s}{\Omega_c}\right)^3}$$

We can get the poles and represent it as

$$H_a(s) = \frac{\Omega_c^3}{(s+\Omega_c)(s-\Omega_c e^{j\frac{2\pi}{3}})(s-\Omega_c e^{-j\frac{2\pi}{3}})}$$

It is partial-fractional expressed as:

$$H_a(s)=\frac{\Omega_c}{s+\Omega_c}+\frac{-(\Omega_c/\sqrt{3})e^{j\pi/6}}{s+\Omega_c(1-j\sqrt{3})/2}+\frac{-(\Omega_c/\sqrt{3})e^{-j\pi/6}}{s+\Omega_c(1+j\sqrt{3})/2}$$

Then we get digital transfer function as:

$$G(z)=\frac{\omega_c}{1-e^{-\omega_c}z^{-1}}+\frac{-(\omega_c/\sqrt{3})e^{j\pi/6}}{1-e^{-\omega_c(1-j\sqrt{3})/2}z^{-1}}+\frac{-(\omega_c/\sqrt{3})e^{-j\pi/6}}{1-e^{-\omega_c(1+j\sqrt{3})/2}z^{-1}}$$

$$\omega_c=\pi/4$$

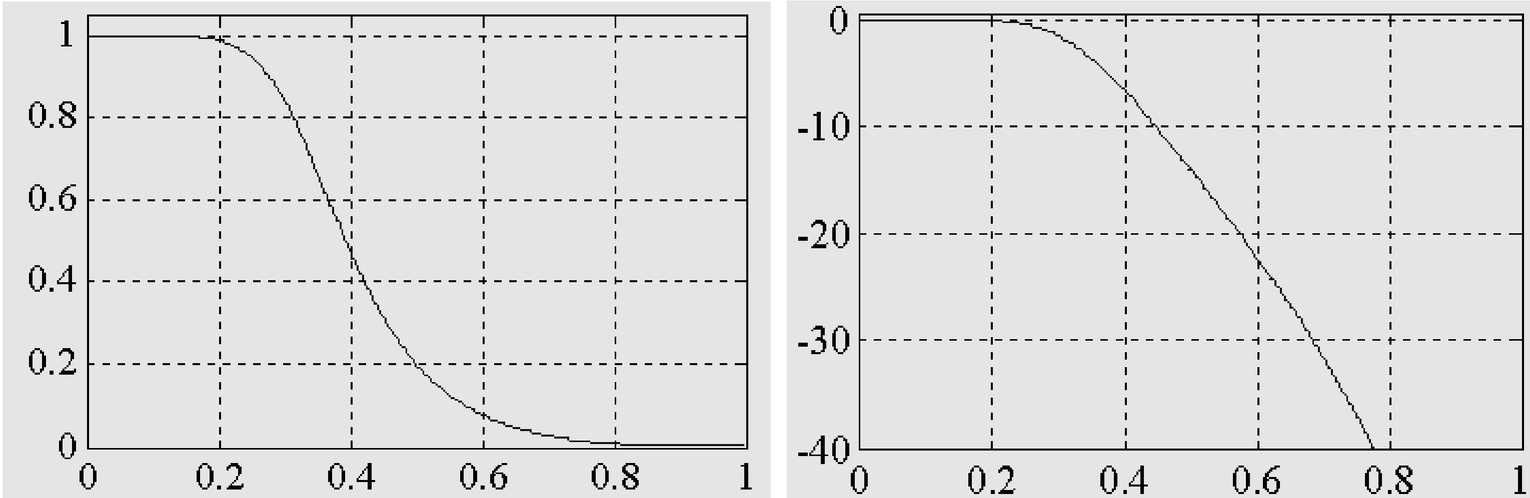
$$G(z)=\frac{0.785398}{1-0.455938z^{-1}}+\frac{-0.785398+0.604884z^{-1}}{1-1.0500756z^{-1}+0.455938z^{-2}}$$

Finally,

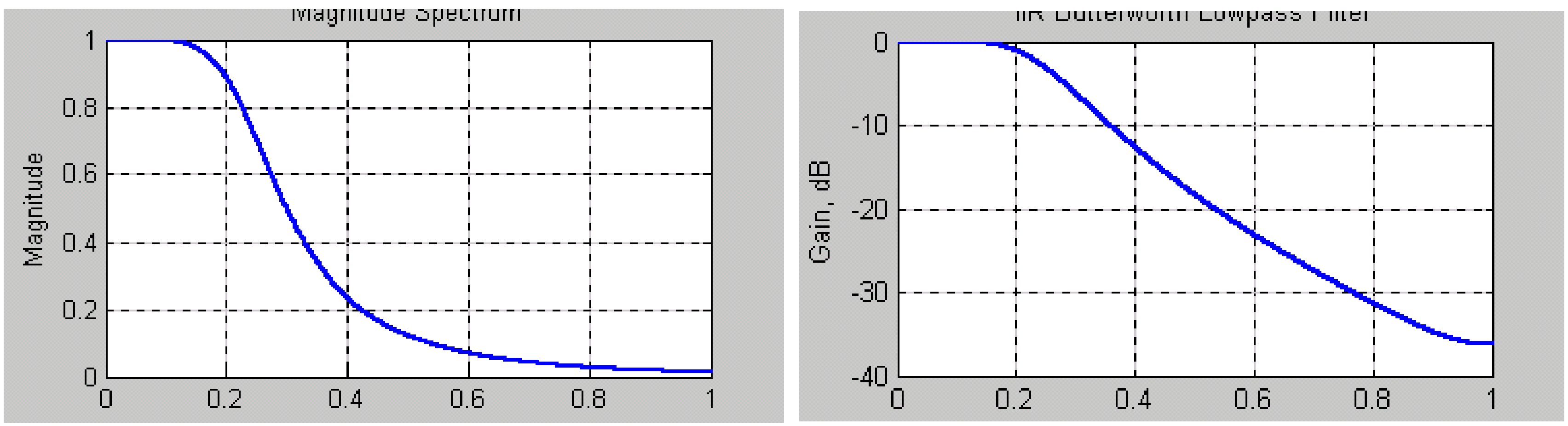
$$G(z)=\frac{0.13825z^{-1}+0.08230z^{-2}}{1-1.50601z^{-1}+0.93471z^{-2}-0.20788z^{-3}}$$

Discussion:

Magnitude and gain responses of *G*(*z*) derived by The Impulse Invariance Method and The Bilinear Translation Method are shown below:



The Bilinear Translation Method



The Impulse Invariance Method

9.4 Design of Highpass, Bandpass, and Bandstop IIR Digital Filters

Read and exercise by yourself !

9.5 Spectral Transformations of IIR Filters

Read Table 9.1 and exercise by yourself !

$$z = F(\hat{z}) \qquad \text{....(9.36)}$$

$$G_D(\hat{z}) = G_L(F(\hat{z})) \qquad \text{....(9.37)}$$

where $z^{-1} = F^{-1}(\hat{z}) = \dfrac{1}{F(\hat{z})}$ **Must be an allpass function.**

9.6 IIR Digital Filter Design Using MATLAB

Read and exercise by yourself!

Note: LPF design directly in digital form

1. Order Estimation

```
[N,Wn] = buttord(Wp, Ws, Rp, Rs);
```

2. Filter Design

```
[b,a] = butt(N,Wn);
% Getting:G(z)=B(z)/A(z)
```

3. Other type filter

```
[b,a] = cheby1(N,Rp,Wn,'high');
%Wn=[W1, W2]
```

<u>Note</u>: Digital LPF design in analog form by Bilinear Transform

1. Order Estimation

```
[N,Wn] = ellipord(Wp,Ws,Rp,Rs,'s');
```

2. Filter Design

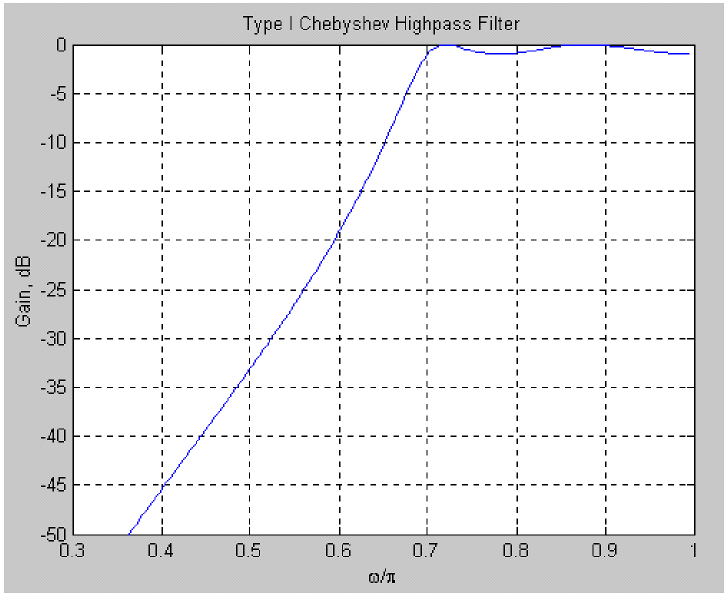
```
[bt,at] = ellip(N,Wn);
%Getting: Ha(z) = Bt(z)/At(z)
```

3. Bilinear Transform

```
[num,den]= bilinear(b,a,0.5);
% 0.5 means T=2,Fs=0.5
% Getting:G(z)=B(z)/A(z)
```

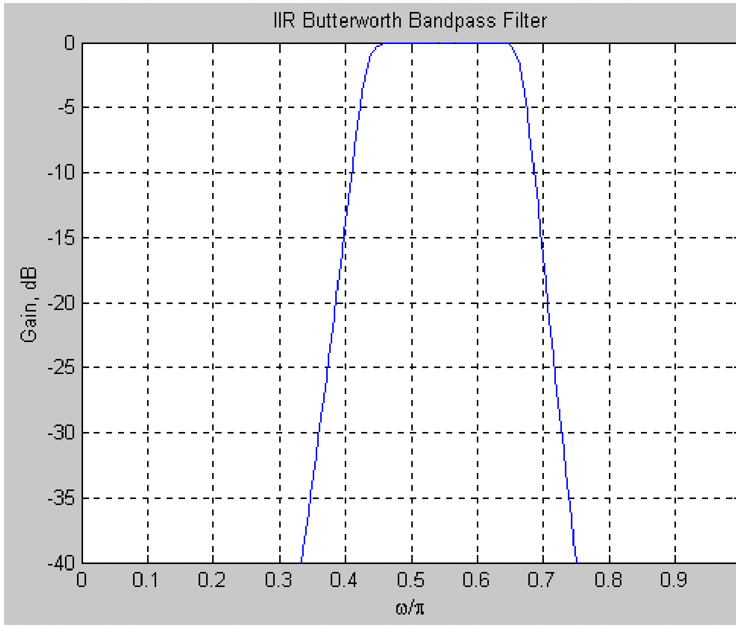
Example (HPF)

(Chebyshev 1)



Example (BPF)

(N = 6, 12th-order)



§ 9 Analysis of Finite Word-length Effects

There are three types of finite word-length effect in DSP.

(1) A/D Conversion noise

(2) Coefficient Quantization of Filter

(3) The Quantization of Arithmetic Operations

Read 8.4~8.6 by yourself to understand the representation of Fixed-Point number and arithmetic operations of binary data.

§ 9.1 The Quantization Process and Errors

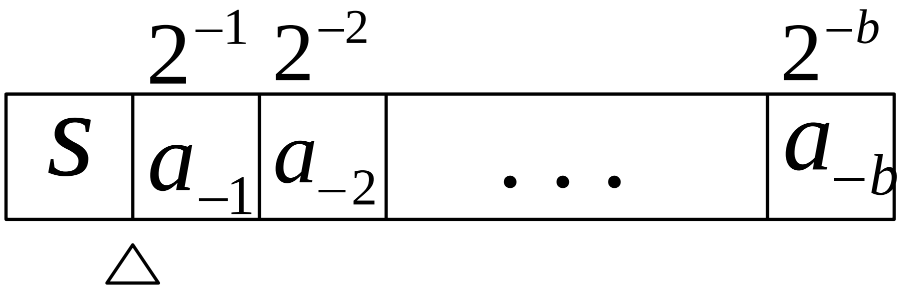


Fig. 9.2 A general (*b*+1)-bit fixed-point fraction

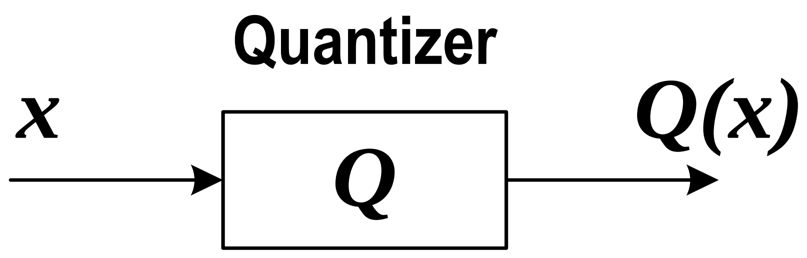


Fig. 9.3: The quantization process model

2^{-b} **is called as *quantization step*.**

x: **the original data**

***Q*(*x*): (b+1)-bit (employed *truncation* or *rounding*)**

§ 9.2 Quantization of Fixed-Point Numbers

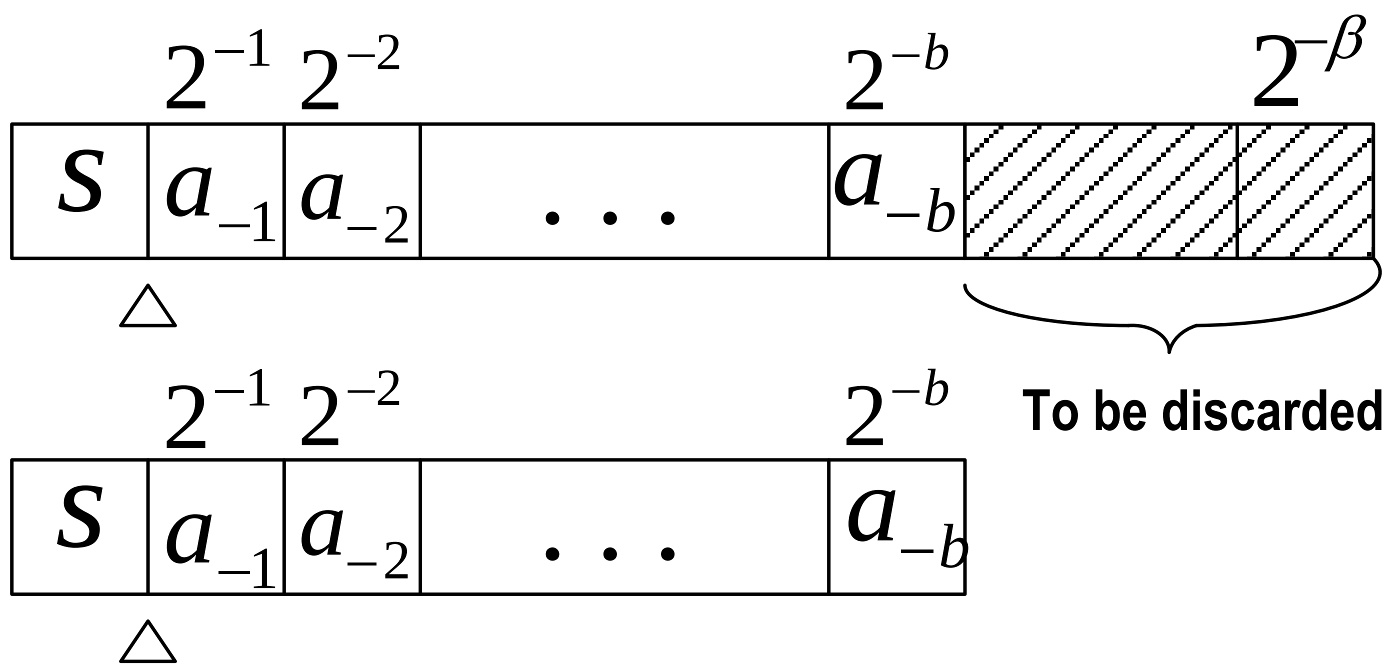
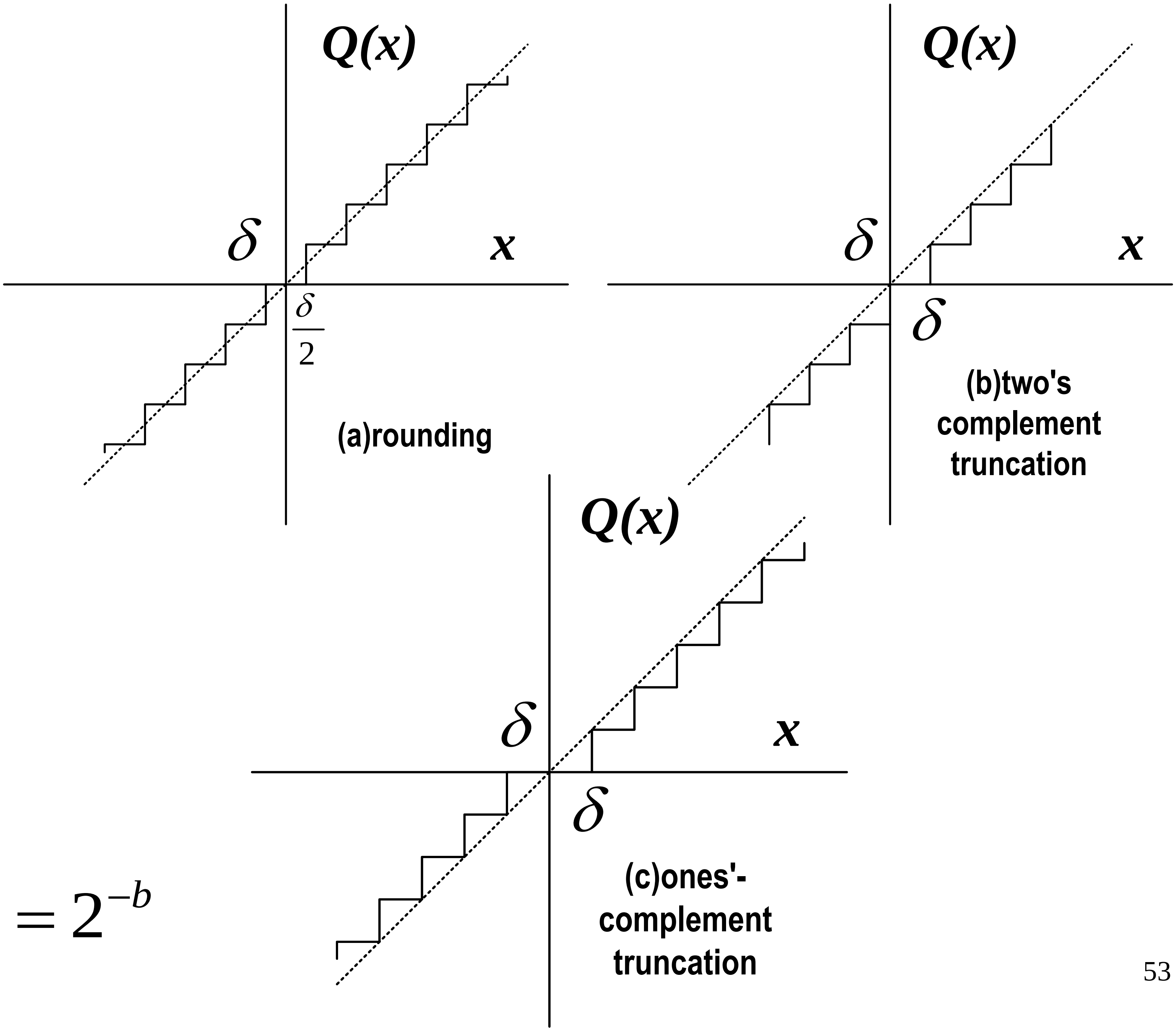


Fig 9.4 and 9.5 illustrate quantization process.


Q(x)
δ
x
δ/2
(a)rounding
Q(x)
δ
x
δ
(b)two's complement truncation
Q(x)
δ
x
δ
(c)ones'-complement truncation

$$\delta = 2^{-b}$$

Table 9.1 Range of quantization error.

Rounding: $$-\frac{1}{2}\delta \le \varepsilon_r \le \frac{1}{2}\delta$$

Truncation $$0 \le \varepsilon_t \le \delta$$ **or** $$-\delta \le \varepsilon_t \le 0$$

§ 9.3 (9.5) A/D Conversion Noise

9.3.1 Quantization Noise Model

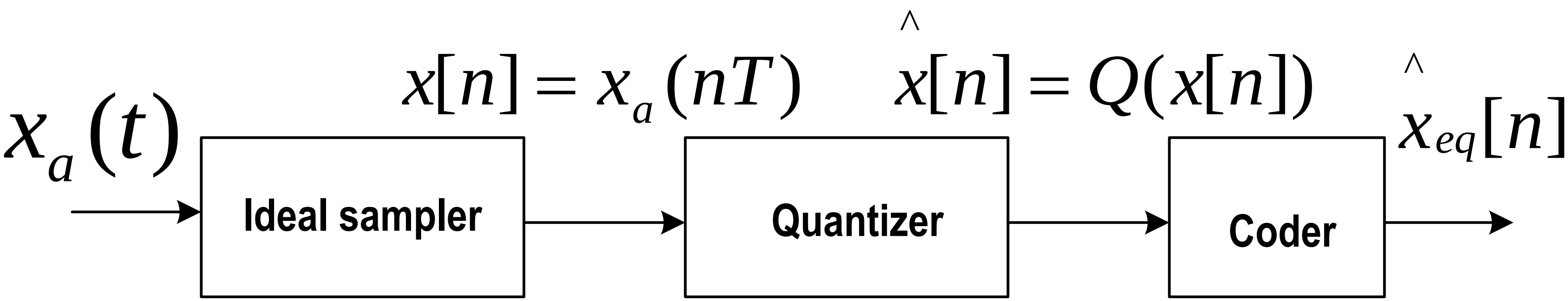
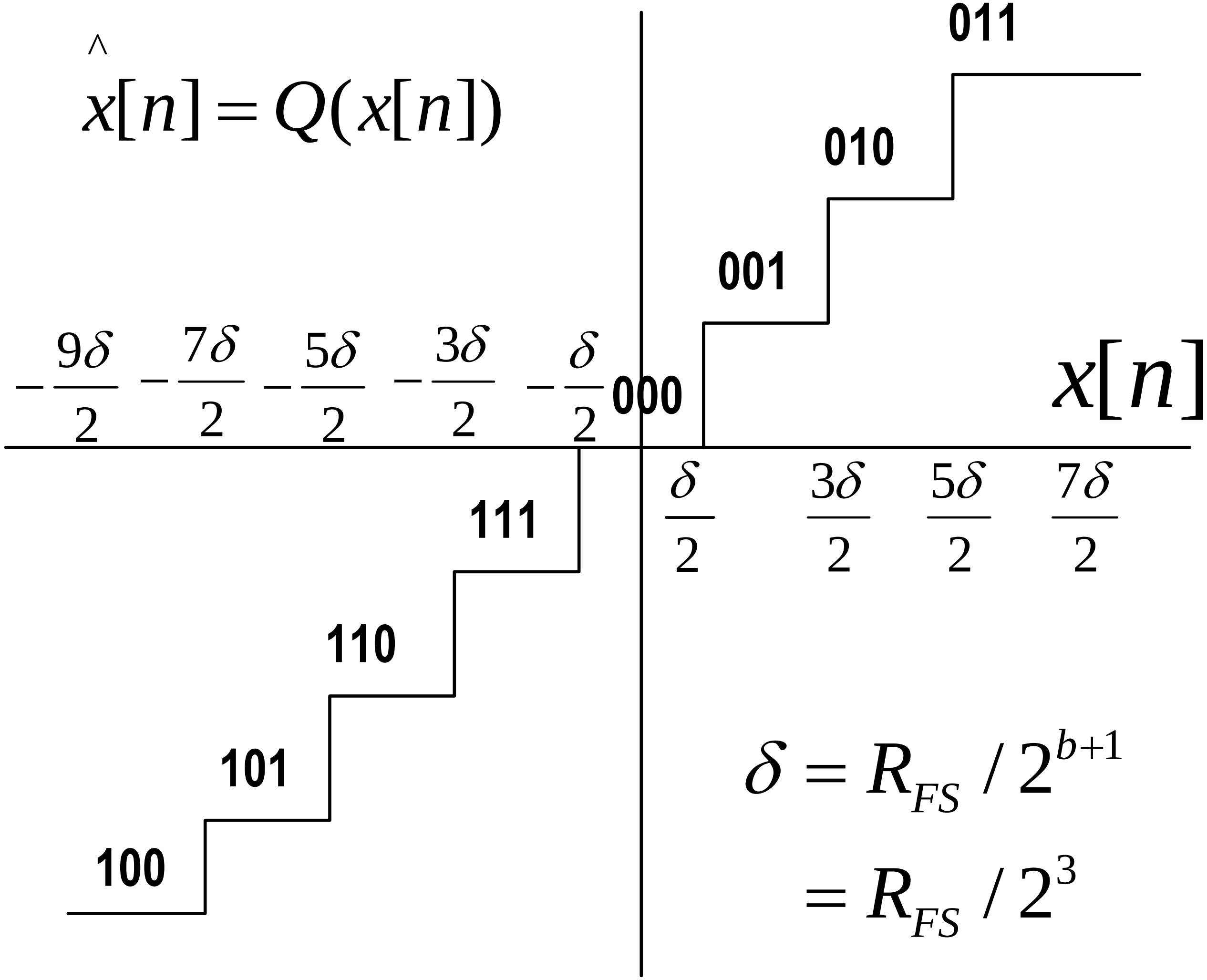


Fig 9.13:Model of a practical A/D conversion system

For example 3-bit bipolar A/D conversion with two's-complement representation (Fig 9.14).

$$\hat{x}[n] = Q(x[n])$$

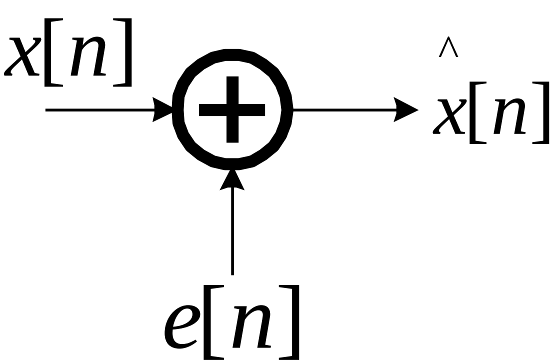


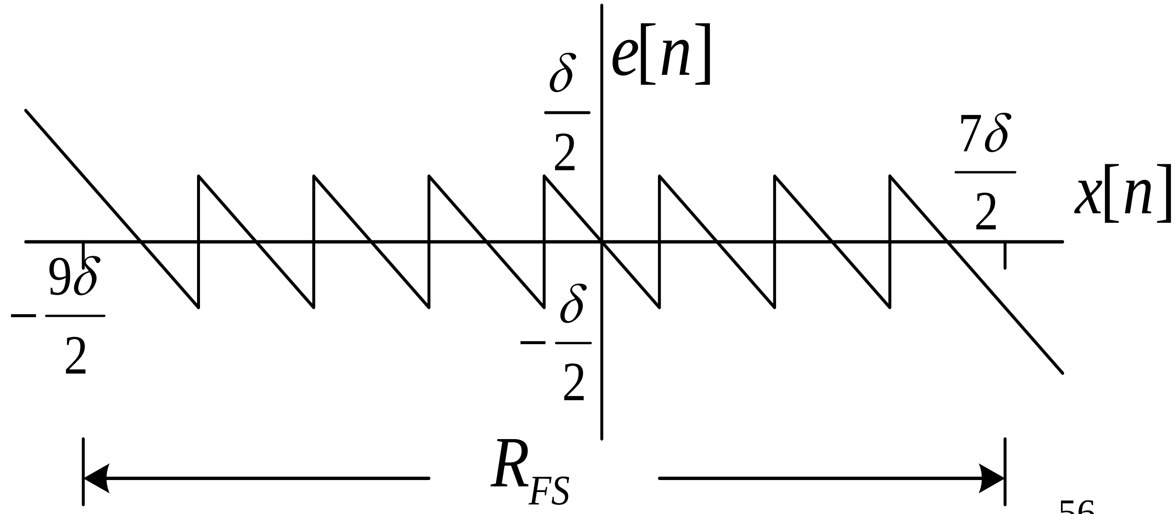
$$\delta = R_{FS} / 2^{b+1}$$
$$= R_{FS} / 2^{3}$$

$$-1 \le \hat{x}_{eq}[n] < 1 \qquad \hat{x}_{eq}[n] = \frac{2\hat{x}[n]}{R_{FS}}$$

R_{FS} : **full-scale range.**

$$e[n] = Q(x[n]) - x[n] \quad \textbf{and} \quad -\frac{\delta}{2} < e[n] \le \frac{\delta}{2}$$





The analysis of error is based on assumptions:

(a) $e[n]$ **is a wide-sense stationary random process, white noise, with uniformly distribution(as Fig 9.17)**

(b) $e[n]$ **is uncorrelated with** $x[n]$

(c) $x[n]$ **is a sample sequence of a stationary random process.**

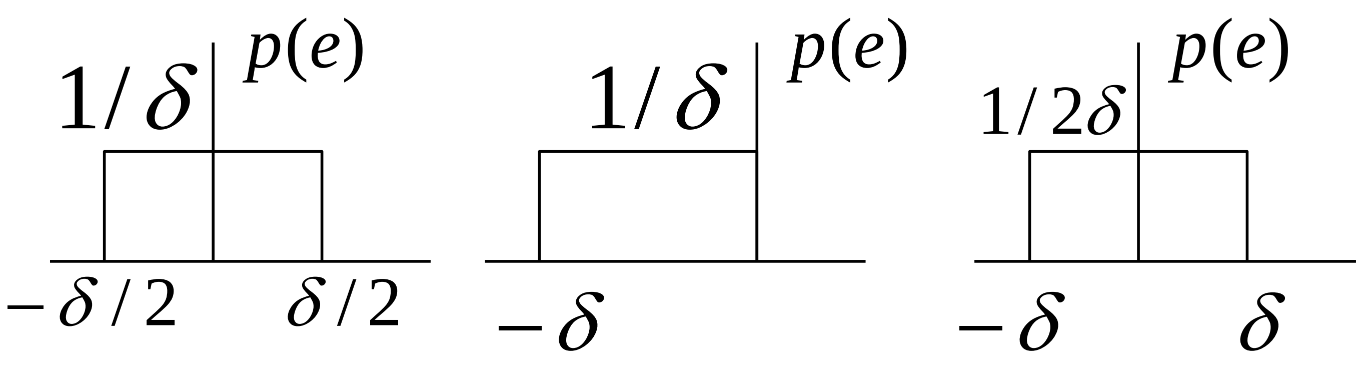


Fig 9.17

In the case of rounding error, the mean and variance are given by

$$m_e = E[e[n]] = \int_{-\infty}^{\infty} e p(e) de$$

$$m_e = \int_{-\delta/2}^{\delta/2} e \frac{1}{\delta} de = 0$$

$$\sigma_e^2 - E\left[(e[n] - m_e)^2\right] = \int_{-\infty}^{\infty} (e - m_e)^2 p(e) de$$

$$\sigma_e^2 = \delta^2 / 12 \qquad \delta = 2^{-b}$$

The corresponding parameters for the two's-complement truncation are as follows:

$$m_e = \int_{-\delta}^{0} e \frac{1}{\delta} de = -\delta / 2 \qquad \sigma_e^2 = \delta^2 / 12$$

9.3.2 Signal-to-Quantization Noise Ratio

$$SNR_{A/D} = 10 \log_{10}(\sigma_x^2 / \sigma_e^2) \; (dB) \cdots (9.72)$$

where σ_x^2 is the input signal variance representing the signal power.

In the case of a bipolar (*b*+1)-bit A/D converter, for rounding

Hence $\delta = 2^{-(b+1)} R_{FS} \quad \sigma_e^2 = 2^{-2b} (R_{FS})^2 / 48$

So,

$$SNR_{A/D} = 10 \log_{10} (48 \sigma_x^2 / (2^{-2b} R_{FS}^2))(dB)$$

$$= 6.02b + 16.81 - 20 \log_{10} (R_{FS} / \sigma_x)(dB)$$

…...(9.74)

σ_x: the RMS value of input signal

This expression can be used to determine the minimum word-length of an A/D converter needed to meet a specified $SNR_{A/D}$

- **<u>Note</u>: $SNR_{A/D}$ increases by 6 dB for each added bit to the word-length**

- **Computed values of the SNR for various values of *K* are as given below:**

Table 9.3

	$b=7$	$b=9$	$b=11$	$b=13$	$b=15$
$K=4$	46.91	58.95	70.99	83.04	95.08
$K=6$	43.39	55.43	67.47	79.51	91.56
$K=8$	40.89	52.93	64.97	77.01	89.05

9.3.3 Effect of Input Scaling on SNR

$$SNR_{A/D} = 6.02b + 16.81 - 20\log_{10}(K) + 20\log_{10}(A)(dB)$$

...(9.77)

- **For a given wordlength, the actual SNR depends on σ_x , the RMS value of the input signal amplitude and the full-scale range R_{FS} of the A/D converter**

- ***Example*** **- Determine the SNR in the digital equivalent of an analog sample $x[n]$ with a zero-mean Gaussian distribution using a $(b+1)$-bit A/D converter having $R_{FS} = K\sigma_x$**

$$SNR_{A/D} = 6.02b + 16.81 - 20\log_{10}(K)\ (dB)$$

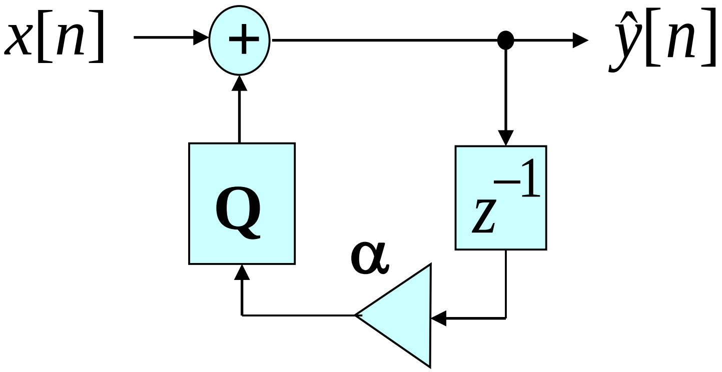
$$K = R_{FS} / \sigma_x \cdots (9.75)$$

Limit Cycles in IIR Digital Filters

- **This type of instability usually results in an oscillatory periodic output called a limit cycle**
- **The system remains in this condition until an input of sufficiently large amplitude is applied to move the system into a more conventional operation**
- **Limit cycles occur in IIR filters due to the presence of feedback**
- **Such oscillations are absent in FIR filters as they do not have any feedback path**

Limit Cycles in IIR Digital Filters

- **Consider the first-order IIR filter as shown right**



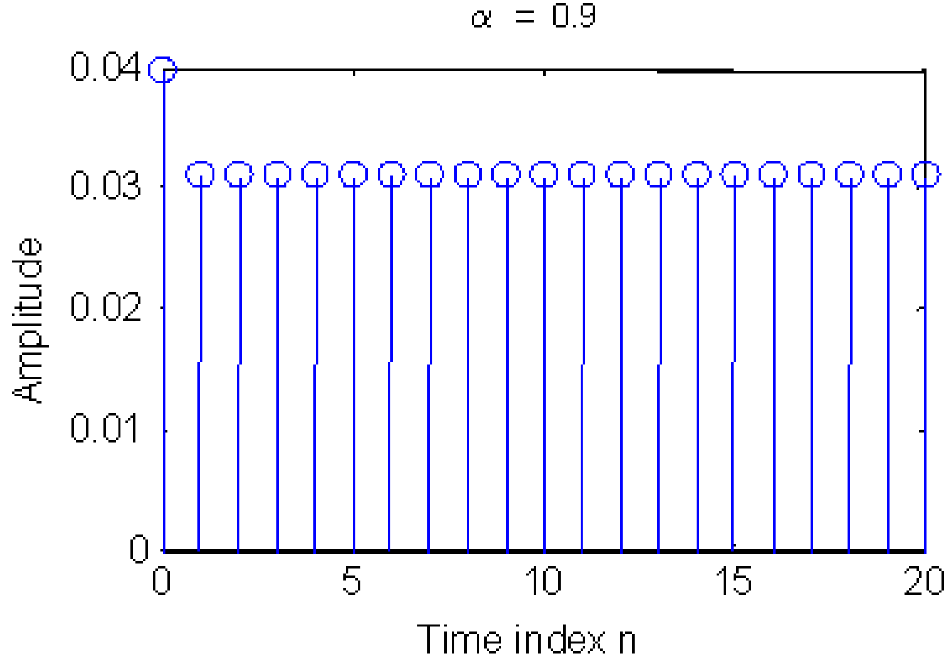
Assume the quantization operation to be rounding and the filter to be implemented with a signed 6-bit fractional arithmetic

- **The nonlinear difference equation characterizing the filter is given by**

$$\hat{y}[n] = \mathrm{Q}\big(\alpha \cdot \hat{y}[n-1]\big) + x[n]$$

Limit Cycles in IIR Digital Filters

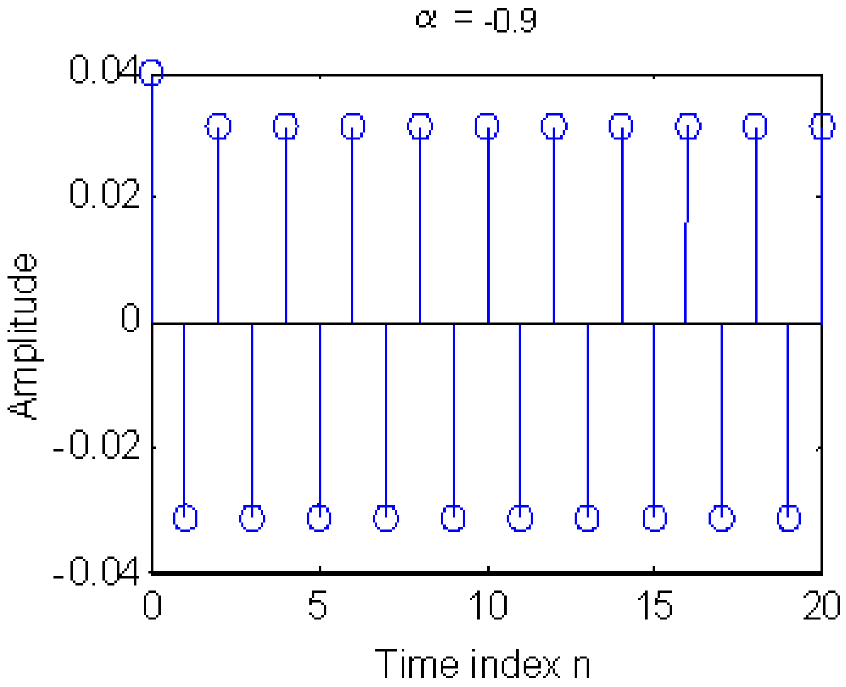
- **For $x[n]$ = 0.03δ[n],** $\hat{y}[-1] = 0$ **, and** α **= 0.9, the output of the filter is as shown below**



- **The limit cycle generated has a period of 1**

Limit Cycles in IIR Digital Filters

- **For $x[n] = 0.03\delta[n]$, $\hat{y}[-1] = 0$, and the output of the filter is as shown below**



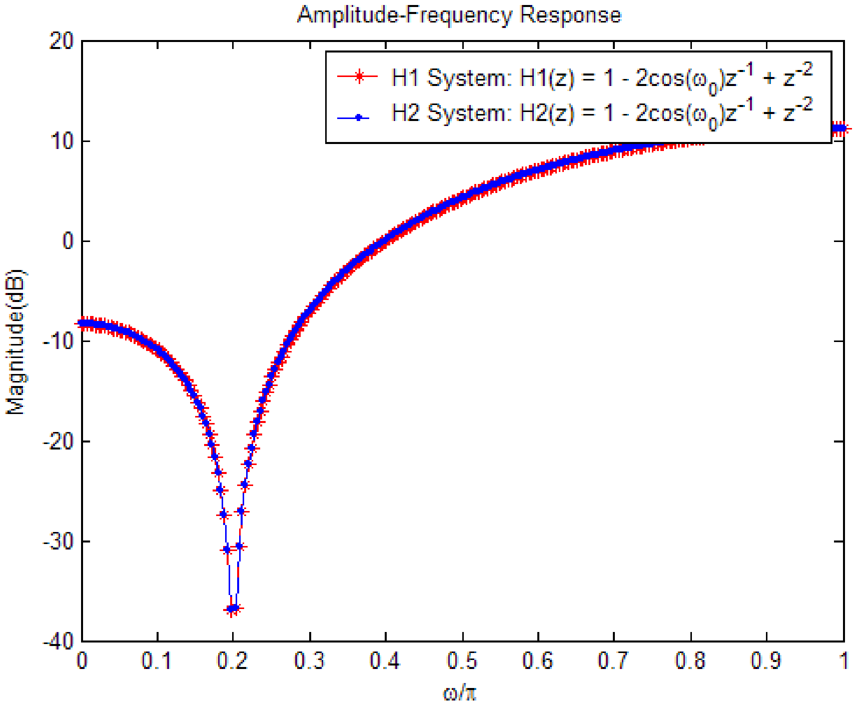
- **The limit cycle generated has a period of 2**

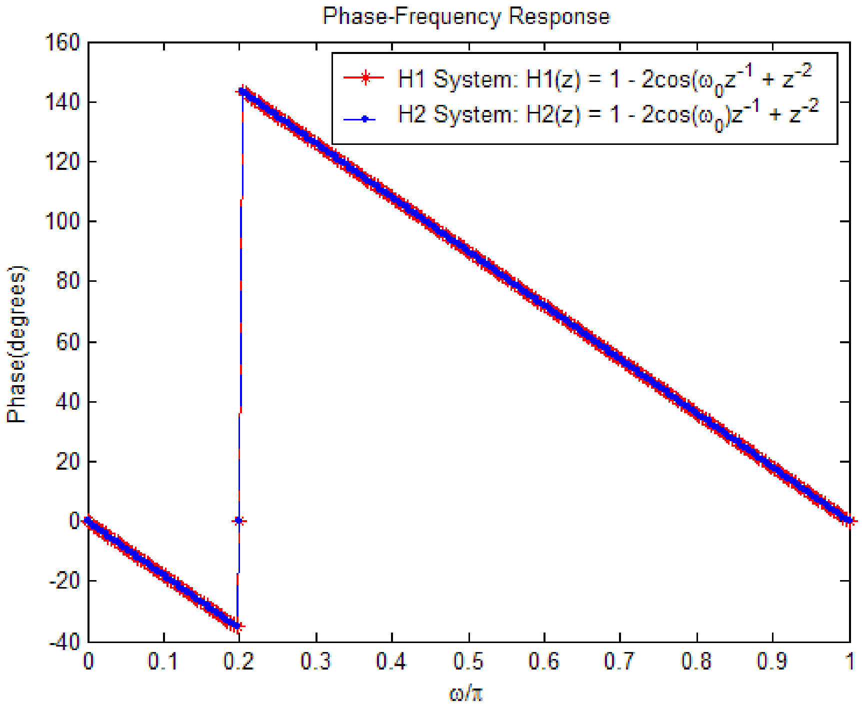
Coefficient Quantization of Filter

- **Notch filter design**
  - No coefficient quantization error
  - No pole

$$H_1(z) = 1 - 2\cos\left(0.2\pi\right)z^{-1} + z^{-2}$$

$$H_2(z) = 1 - 2\cos\left(0.2\pi\right)z^{-1} + z^{-2}$$


Amplitude-Frequency Response
H1 System: H1(z) = 1 - 2cos(ω0)z-1 + z-2
H2 System: H2(z) = 1 - 2cos(ω0)z-1 + z-2
Magnitude(dB)
ω/π
20
10
0
-10
-20
-30
-40
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


Phase-Frequency Response
H1 System: H1(z) = 1 - 2cos(ω0z-1 + z-2
H2 System: H2(z) = 1 - 2cos(ω0)z-1 + z-2
Phase(degrees)
ω/π
160
140
120
100
80
60
40
20
0
-20
-40
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

Thanks!

Any questions?