

Signals and Systems

Lecturer: Dr. Lin Xu, Dr. Xiran Cai

Email: xulin1@shanghaitech.edu.cn

caixr@shanghaitech.edu.cn

Office: 3-428(Xu), 3-438(Cai) SIST

Tel: 20684449(Xu), 20684431(Cai)

ShanghaiTech University



Chapter 7: Sampling

- ❑ **Representation of a Continuous-Time Signal by Its Samples: The Sampling Theorem**
- ❑ **Reconstruction of a Signal from Its Samples Using Interpolation**
- ❑ **The Effect of Undersampling: Aliasing**
- ❑ **Discrete-Time Processing of Continuous-Time Signals**
- ❑ **Sampling of Discrete-Time Signals**



The Sampling Theorem

- What is sampling?

Converting continuous-time signals to discrete-time signals

- Why sampling?

To use the well-developed digital technology

- But, a signal could not always be uniquely specified by equally-spaced samples

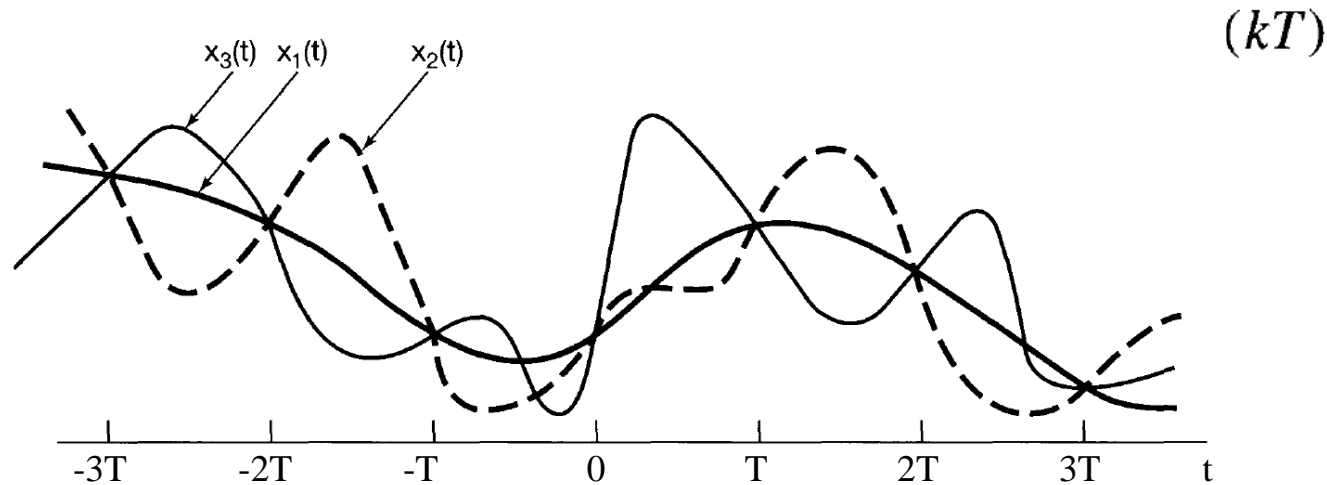


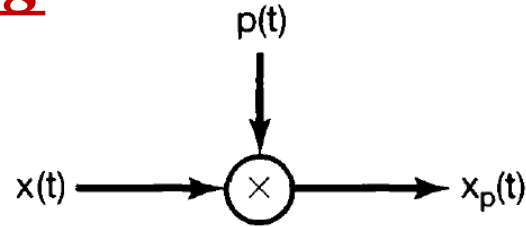
Figure 7.1 Three continuous-time signals with identical values at integer multiples of T .

- The sampling theorem should be satisfied



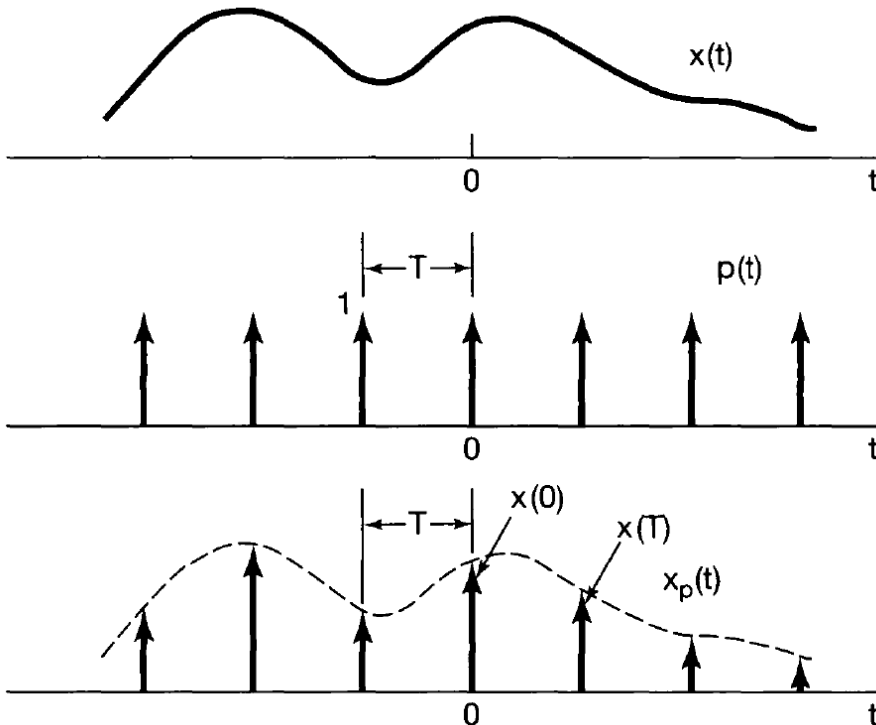
The Sampling Theorem

Impulse-Train Sampling



$$x_p(t) = x(t) \cdot p(t)$$

□ Time domain



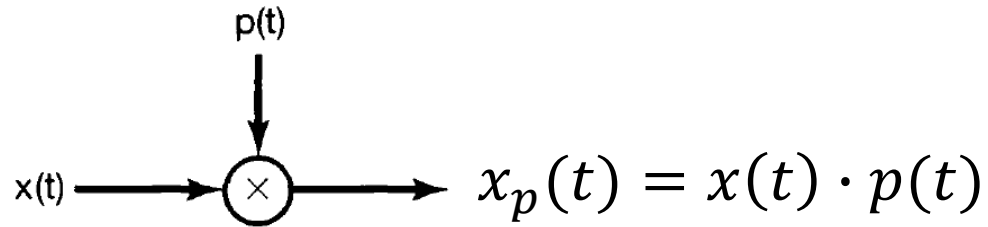
$$p(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT)$$

$$x_p(t) = \sum_{n=-\infty}^{\infty} x(nT) \cdot \delta(t - nT)$$



The Sampling Theorem

Impulse-Train Sampling

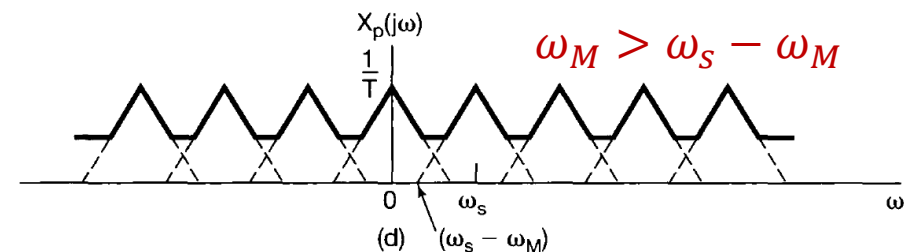
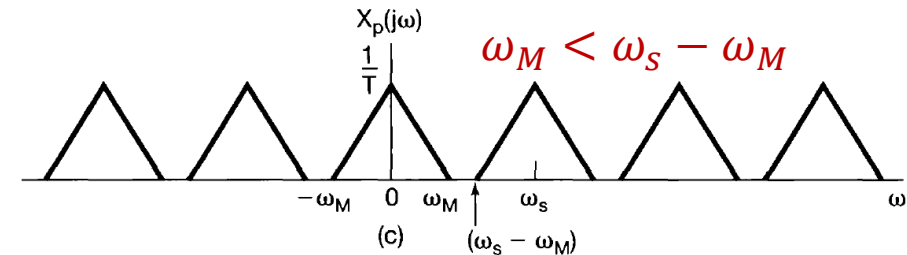
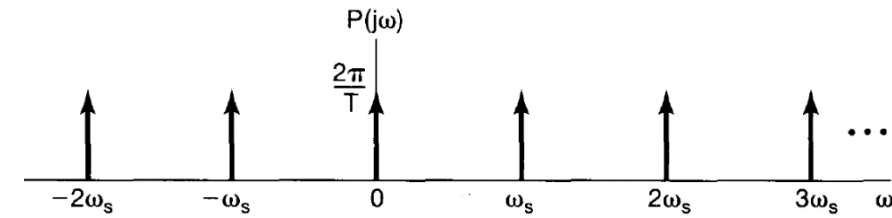
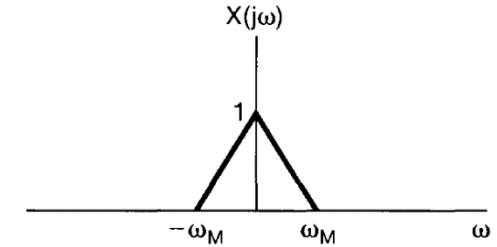


□ Frequency domain

$$X_p(j\omega) = \frac{1}{2\pi} X(j\omega) * P(j\omega)$$

$$P(j\omega) = \frac{2\pi}{T} \sum_{K=-\infty}^{\infty} \delta(\omega - k\omega_s) = \frac{2\pi}{T} \sum_{K=-\infty}^{\infty} \delta\left(\omega - k \frac{2\pi}{T}\right)$$

$$X_p(j\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(j\theta) P(j(\omega - \theta)) d\theta = \frac{1}{T} \sum_{k=-\infty}^{\infty} X(j(\omega - k \cdot \omega_s))$$



The Sampling Theorem

Sampling Theorem

Sampling Theorem:

Let $x(t)$ be a band-limited signal with $X(j\omega) = 0$ for $|\omega| > \omega_M$. Then $x(t)$ is uniquely determined by its samples $x(nT)$, $n = 0, \pm 1, \pm 2, \dots$, if

$$\omega_s > 2\omega_M,$$

where

$$\omega_s = \frac{2\pi}{T}.$$

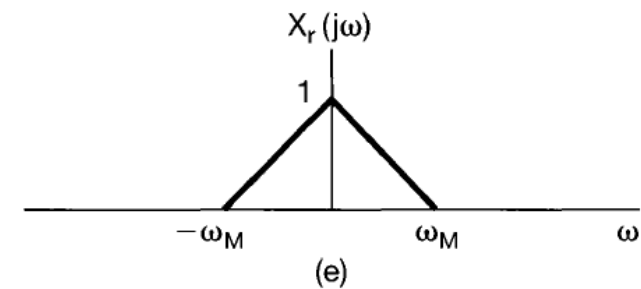
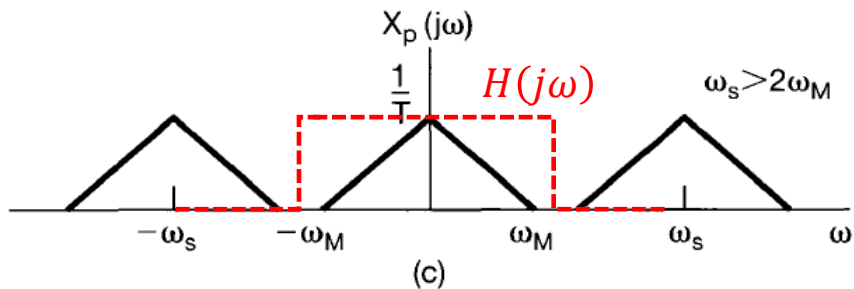
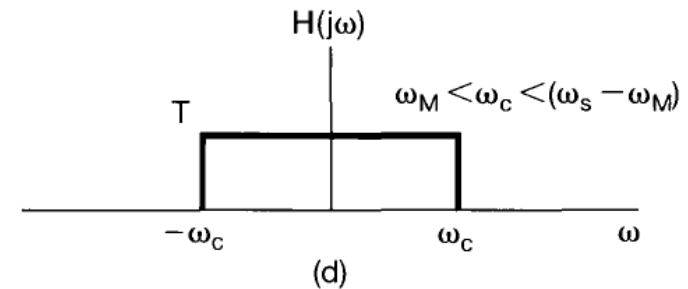
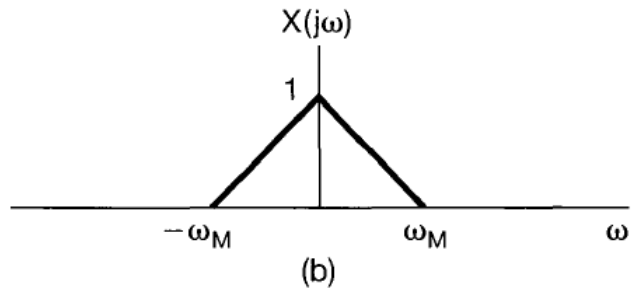
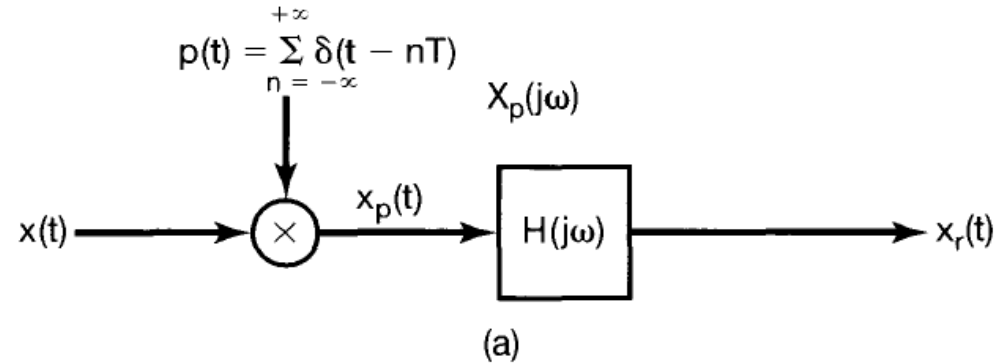
Given these samples, we can reconstruct $x(t)$ by generating a periodic impulse train in which successive impulses have amplitudes that are successive sample values. This impulse train is then processed through an ideal lowpass filter with gain T and cutoff frequency greater than ω_M and less than $\omega_s - \omega_M$. The resulting output signal will exactly equal $x(t)$.



The Sampling Theorem

Recovery of the CT signal

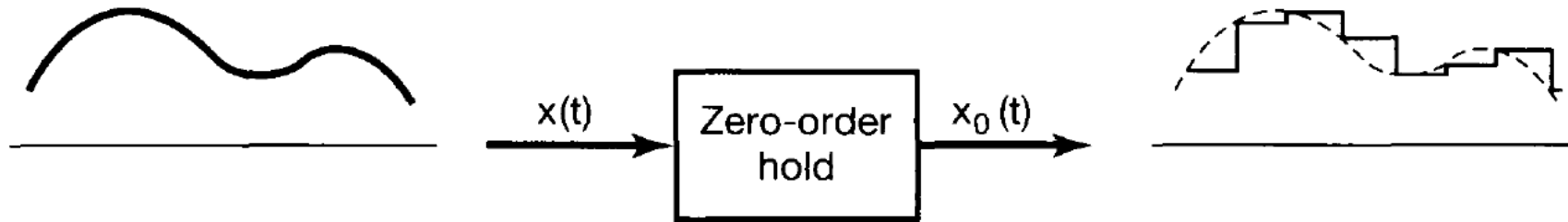
□ Ideal low-pass filtering



The Sampling Theorem

Sampling with a Zero-order Hold

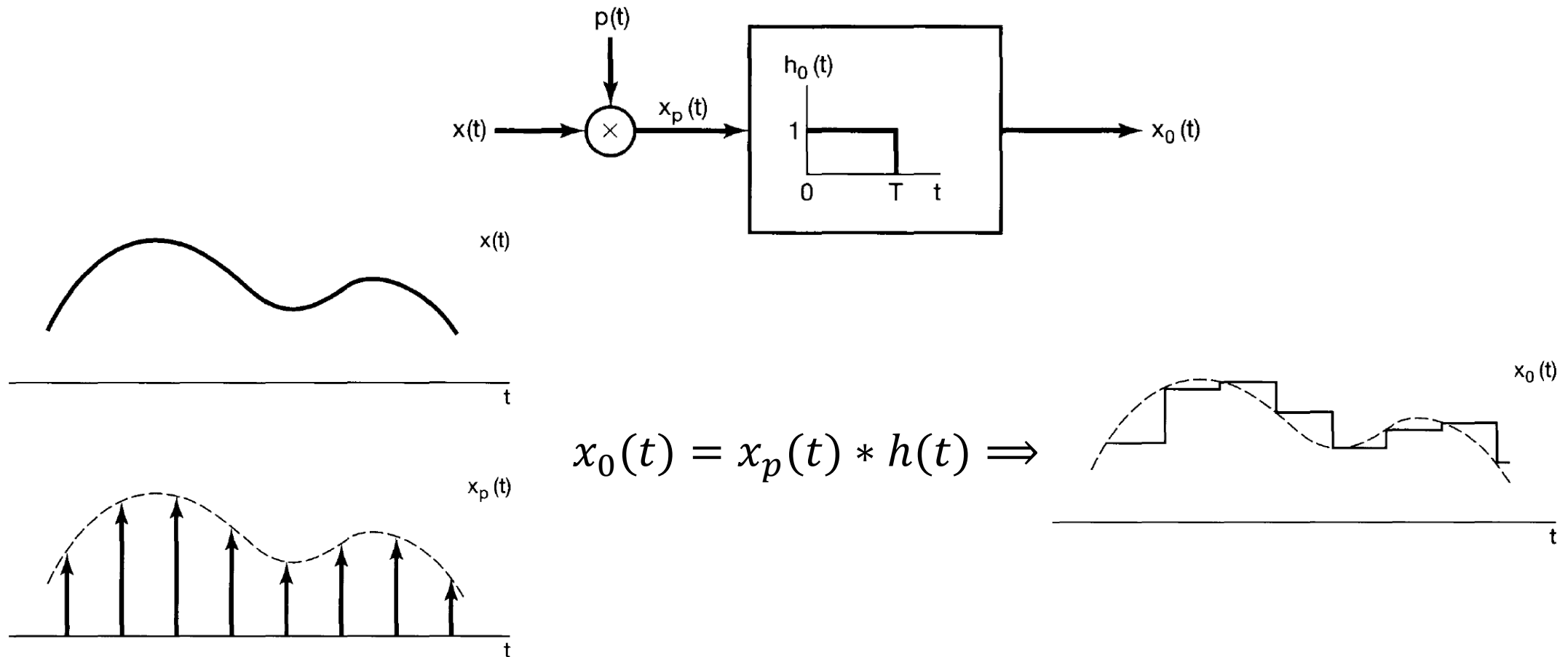
- **Why:** Impulse-train is difficult to generate
- **Principle:** Samples $x(t)$ at a given instant and holds that value until the next instant



The Sampling Theorem

Sampling with a Zero-order Hold

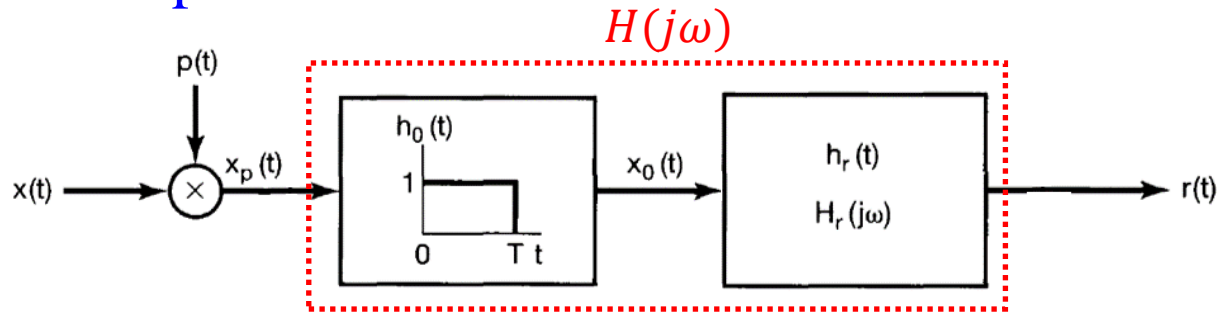
□ **Equivalent:** Impulse-train sampling + an LTI system with a rectangular impulse response



The Sampling Theorem

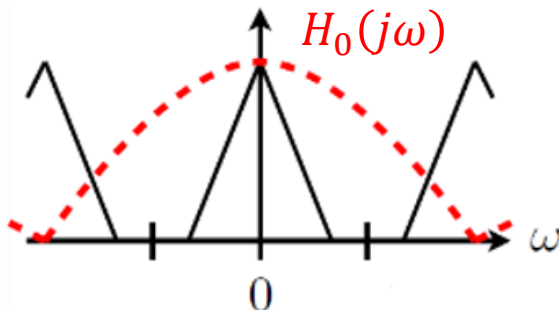
Sampling with a Zero-order Hold

□ Compensation filter

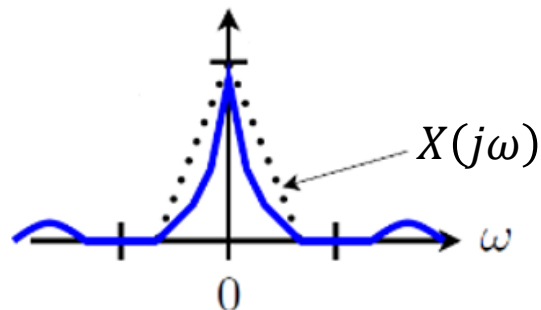


$$H_0(j\omega) = e^{-j\omega T/2} \left[\frac{2 \sin(\omega T/2)}{\omega} \right]$$

$X_p(j\omega)$

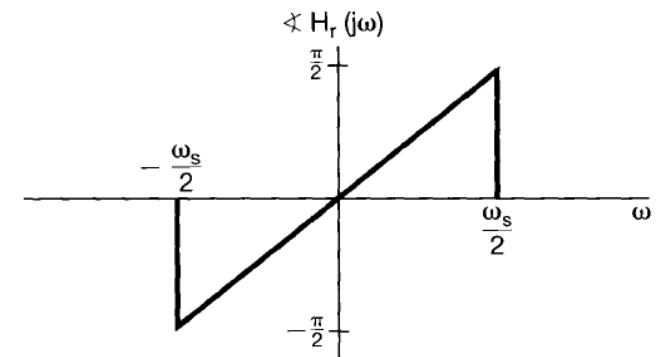
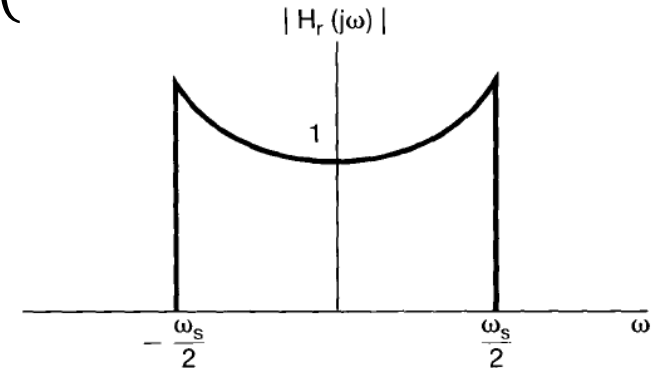


$X_0(j\omega)$



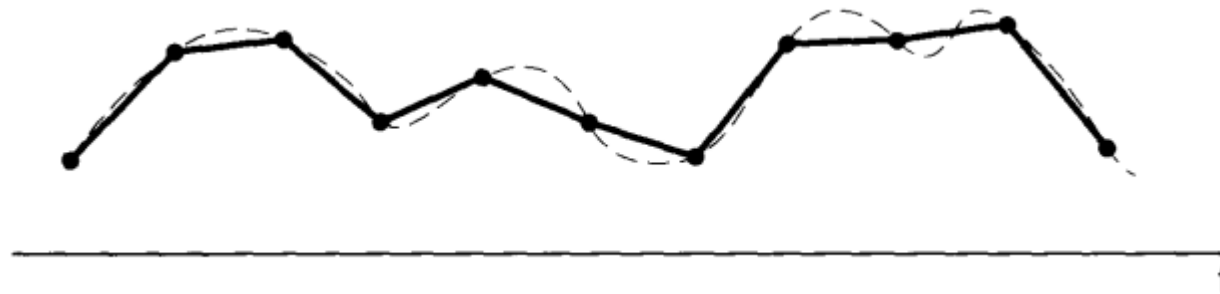
Let $H_0(j\omega)H_r(j\omega) = H(j\omega)$

$$H_r(j\omega) = \begin{cases} e^{j\omega T/2} / \left[\frac{2 \sin(\omega T/2)}{\omega} \right], & |\omega| \leq \frac{\omega_s}{2} \\ 0, & |\omega| > \frac{\omega_s}{2} \end{cases}$$



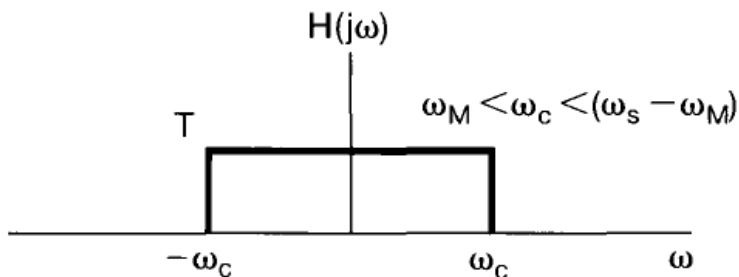
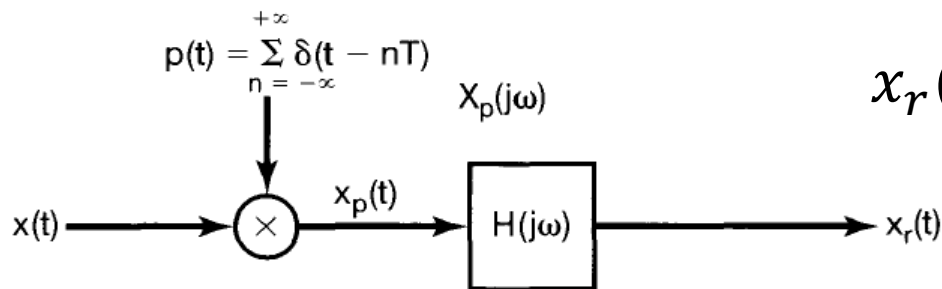
Chapter 7: Sampling

- ❑ Representation of a Continuous-Time Signal by Its Samples: The Sampling Theorem
- ❑ Reconstruction of a Signal from Its Samples Using Interpolation
- ❑ The Effect of Undersampling: Aliasing
- ❑ Discrete-Time Processing of Continuous-Time Signals
- ❑ Sampling of Discrete-Time Signals



Reconstruction of a Signal Using Interpolation

Band-limited interpolation: (ideal low-pass filter)



$$h(t) = \frac{T\omega_c}{\pi} \frac{\sin \omega_c t}{\omega_c t}$$

$$x_r(t) = x_p(t) * h(t) = \left[\sum_{n=-\infty}^{\infty} x(nT) \cdot \delta(t - nT) \right] * h(t)$$

$x_p(t)$

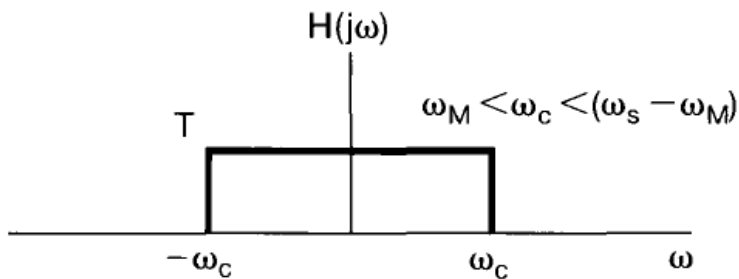
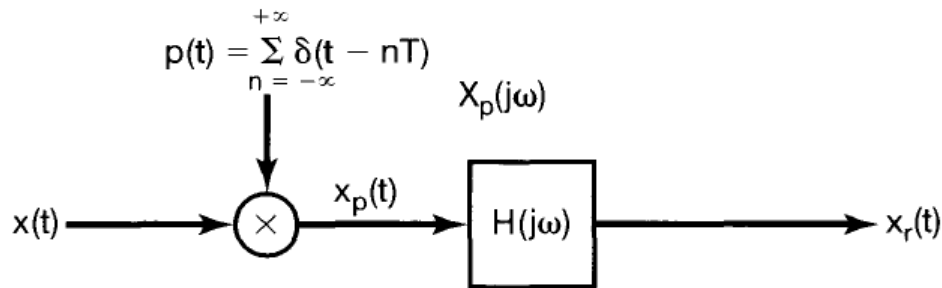
$$= \sum_{n=-\infty}^{\infty} x(nT) [\delta(t - nT) * h(t)]$$

$$= \sum_{n=-\infty}^{\infty} x(nT) h(t - nT)$$

$$= \sum_{n=-\infty}^{\infty} x(nT) \frac{T\omega_c}{\pi} \frac{\sin \omega_c(t - nT)}{\omega_c(t - nT)}$$

Reconstruction of a Signal Using Interpolation

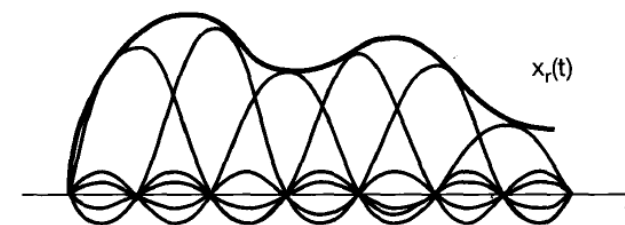
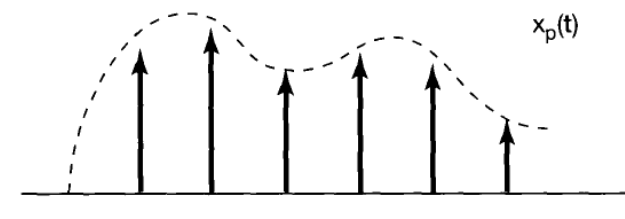
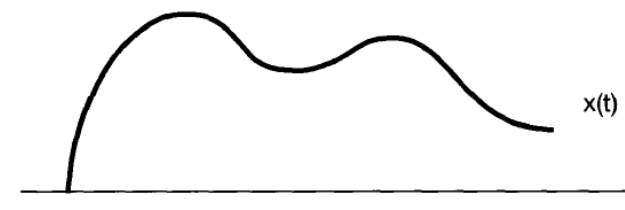
Band-limited interpolation: (ideal low-pass filter)



$$h(t) = \frac{T\omega_c}{\pi} \frac{\sin \omega_c t}{\omega_c t}$$

$$\omega_c = \frac{\omega_s}{2}$$

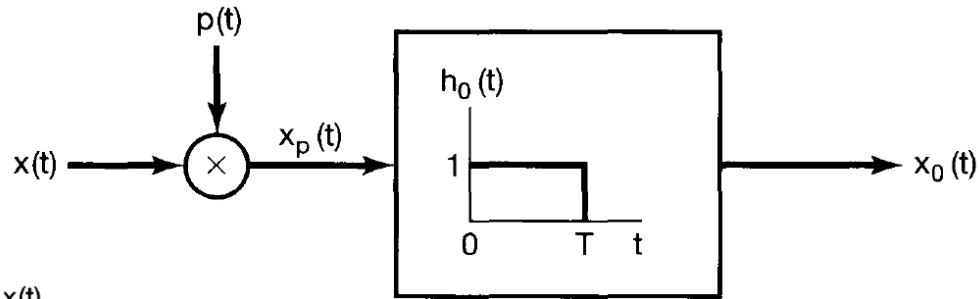
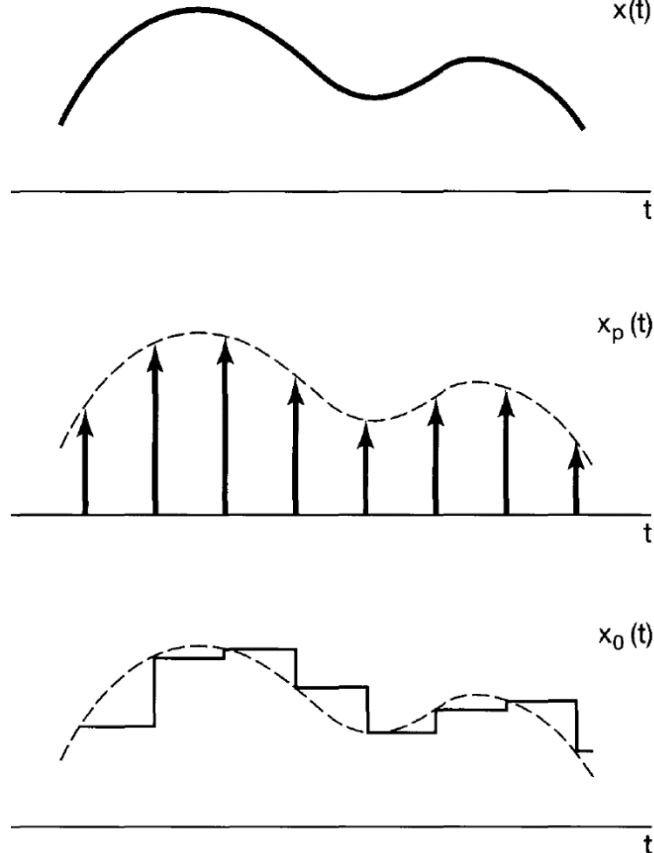
$$x_r(t) = \sum_{n=-\infty}^{\infty} x(nT) \frac{T\omega_c}{\pi} \frac{\sin \omega_c(t - nT)}{\omega_c(t - nT)}$$



Reconstruction of a Signal Using Interpolation

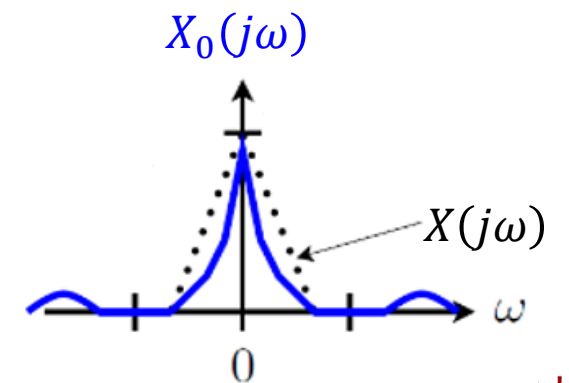
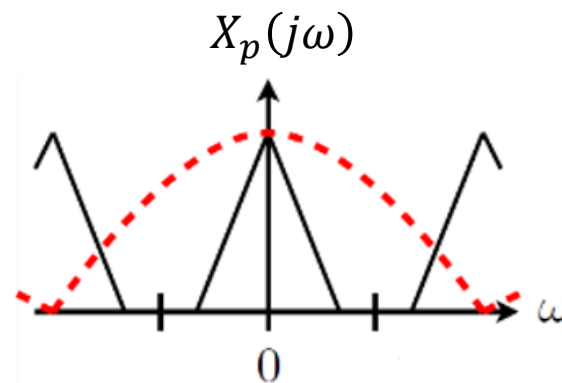
Zero-order hold

□ Time domain



□ Frequency domain

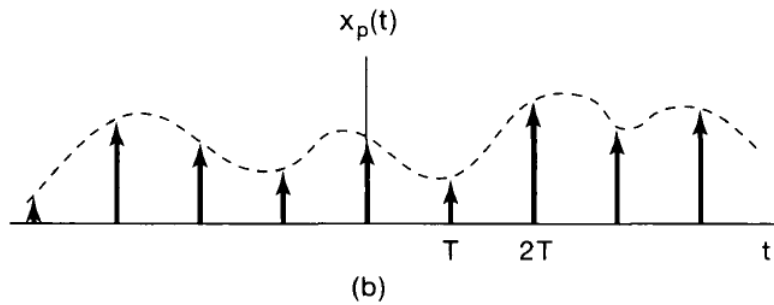
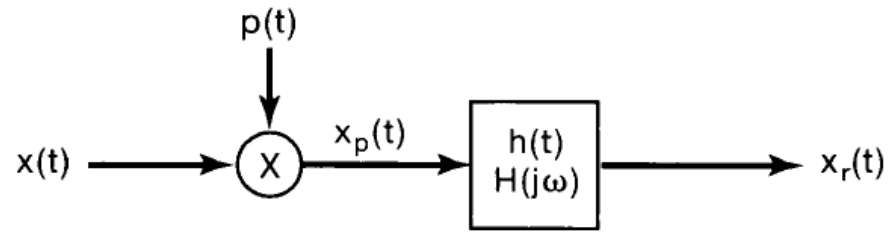
$$H_0(j\omega) = e^{-j\omega T/2} \left[\frac{2 \sin \omega T}{\omega} \right]$$



Reconstruction of a Signal Using Interpolation

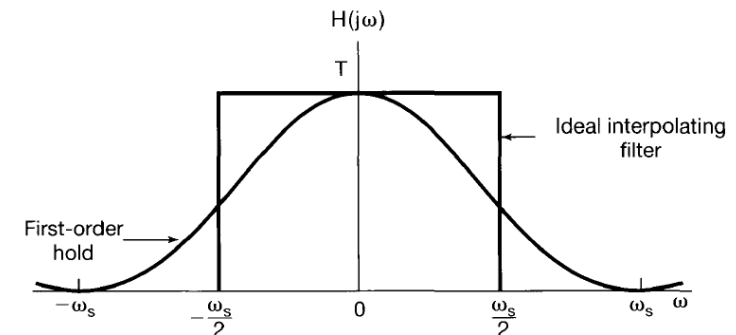
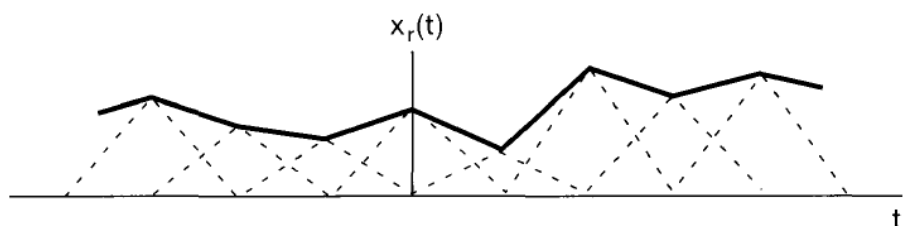
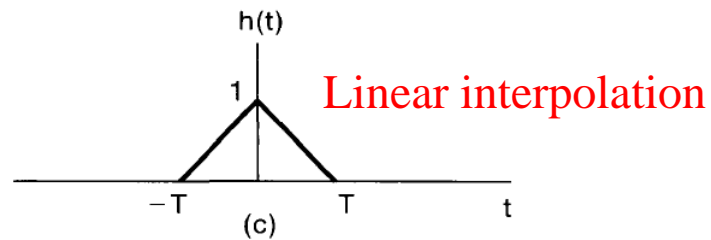
First-order hold: Impulse-train sampling + an LTI system with a triangular impulse response

□ Time domain



□ Frequency domain

$$H(j\omega) = \frac{1}{T} \left[\frac{\sin(\omega T/2)}{\omega/2} \right]^2$$



Chapter 7: Sampling

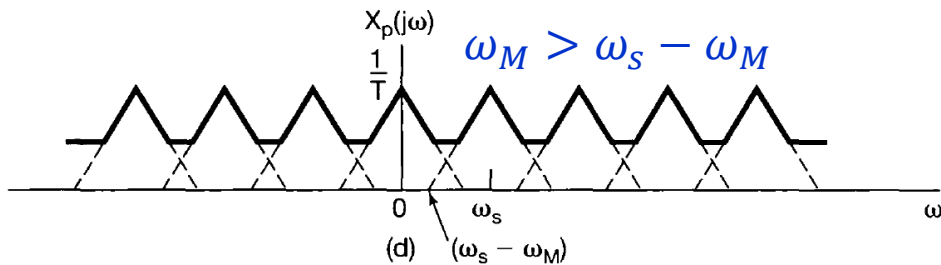
- ❑ Representation of a Continuous-Time Signal by Its Samples: The Sampling Theorem
- ❑ Reconstruction of a Signal from Its Samples Using Interpolation
- ❑ The Effect of Undersampling: Aliasing
- ❑ Discrete-Time Processing of Continuous-Time Signals
- ❑ Sampling of Discrete-Time Signals



Aliasing

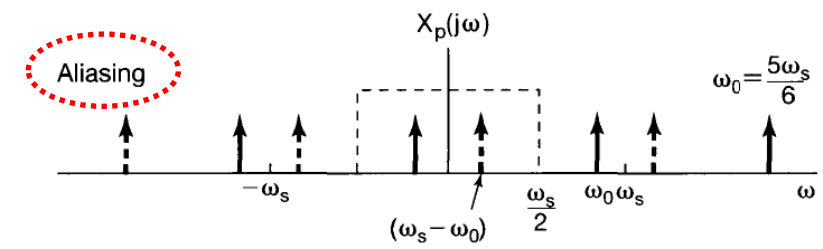
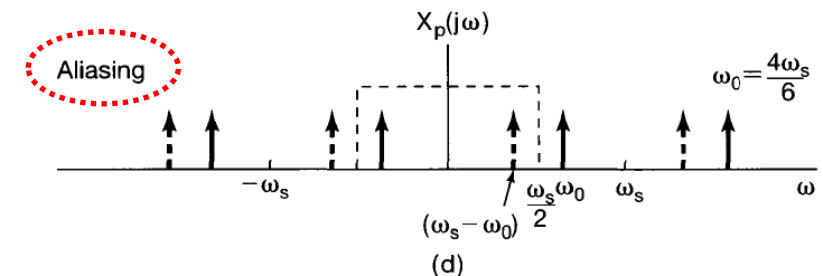
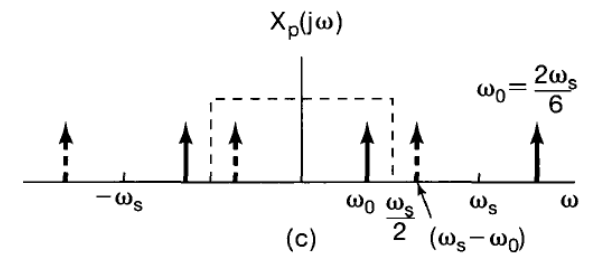
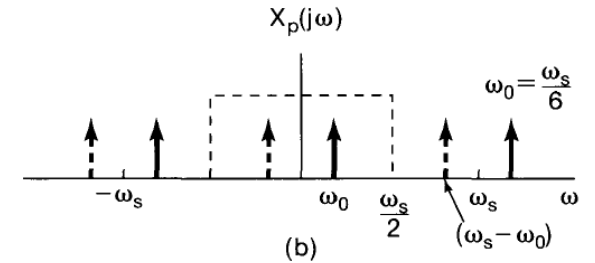
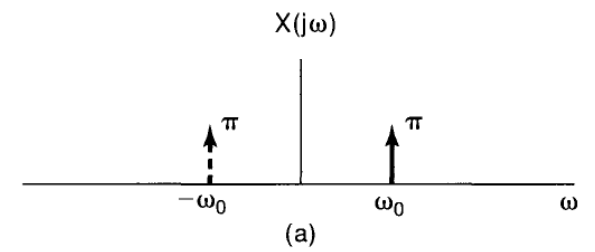
Aliasing

□ When $\omega_s < 2\omega_M$, the individual spectrums overlap



□ Consider original signal is $x(t) = \cos \omega_0 t$, with different ω_0 but sampled at same ω_s

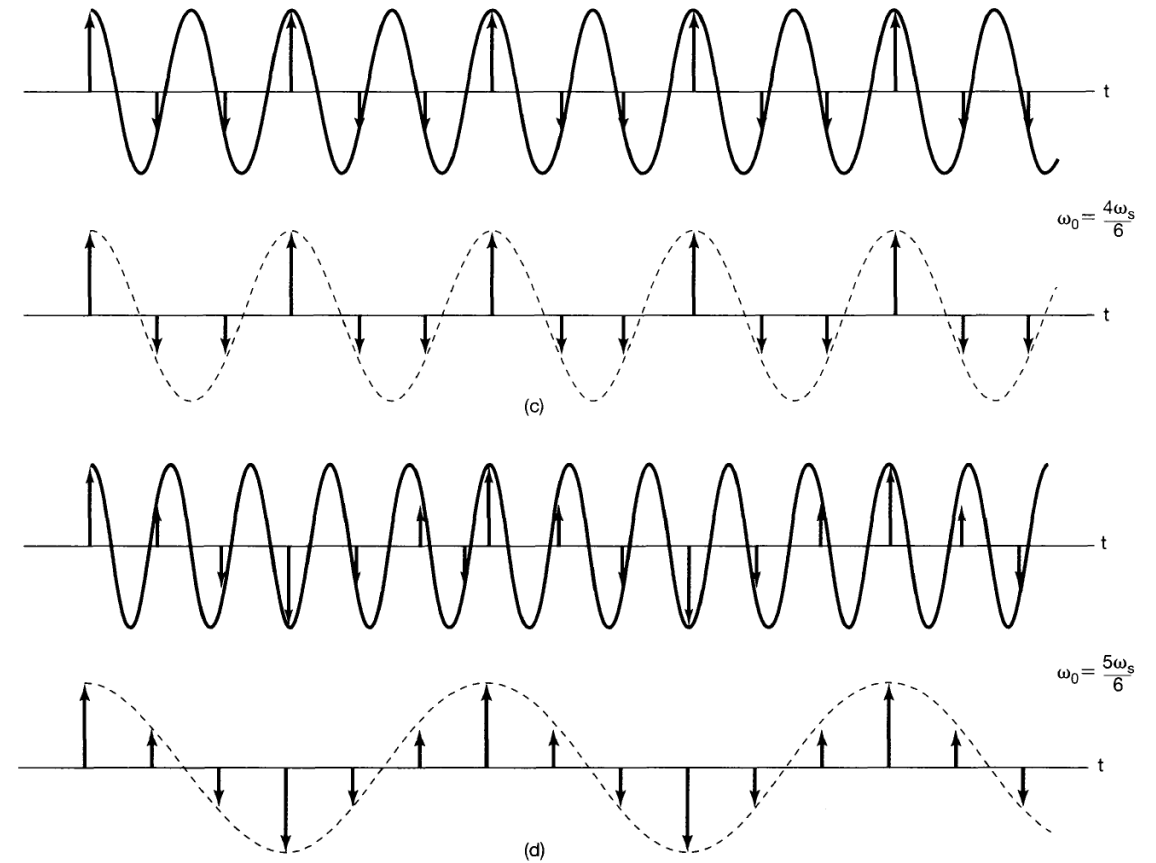
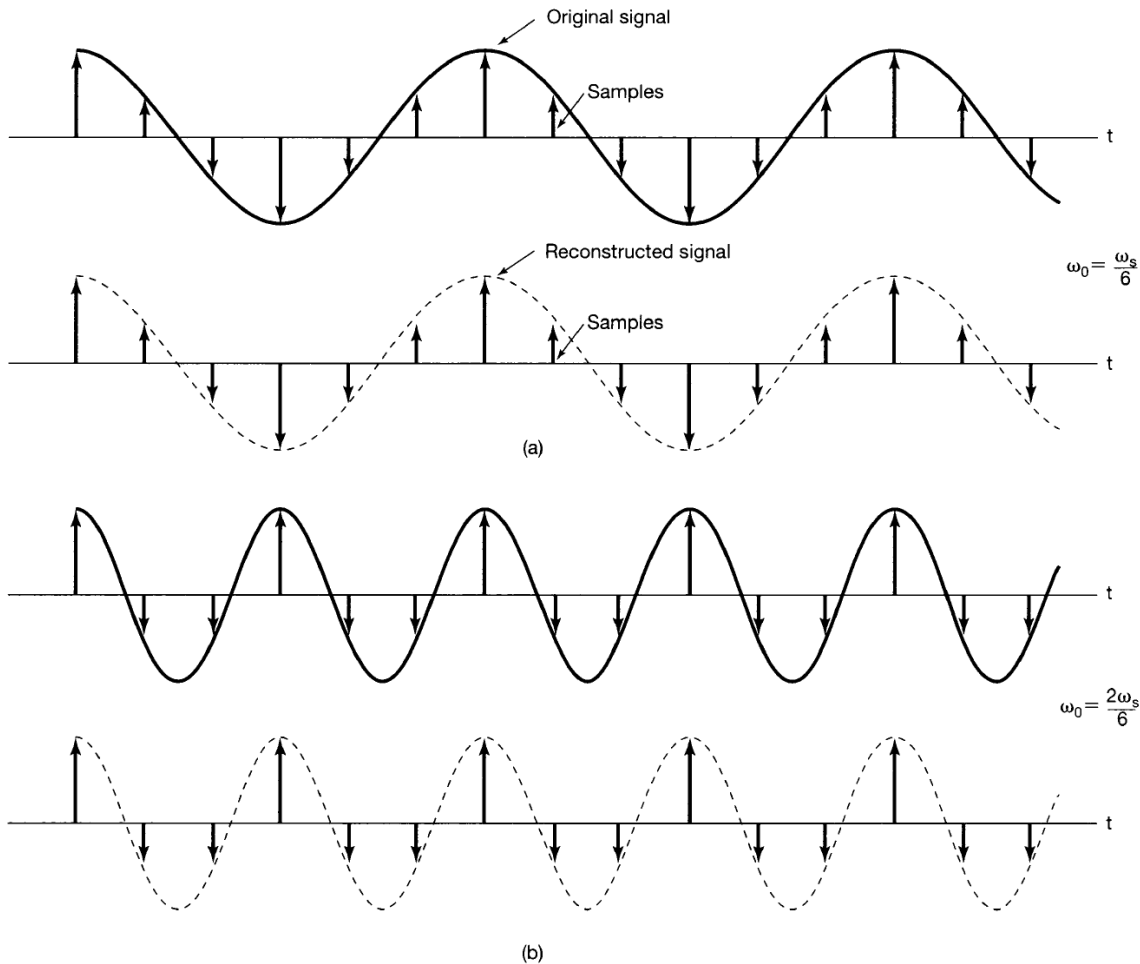
- When aliasing occurs, the original frequency ω_0 takes on the identity of lower frequency $(\omega_s - \omega_0)$.



Aliasing

Aliasing

$$x(t) = \cos \omega_0 t \quad \text{Time domain}$$



Aliasing

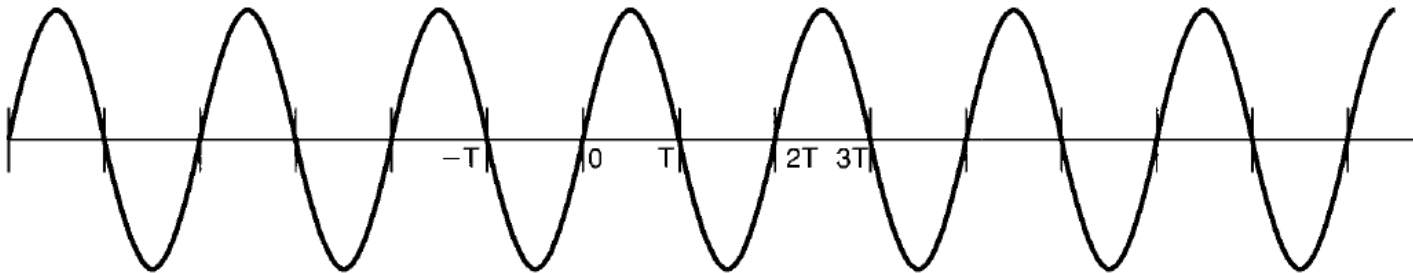
Aliasing

□ $\omega_s = 2\omega_M$ is not sufficient to avoid aliasing

- Consider a signal $x(t) = \cos(\omega_0 t + \phi)$ is sampled using impulse sampling with $\omega_s = 2\omega_0$
- The reconstructed signal using ideal low-pass filter is

$$x_r(t) = \cos(\phi) \cos(\omega_0 t) = x(t) \text{ only if } \phi = 2k\pi$$

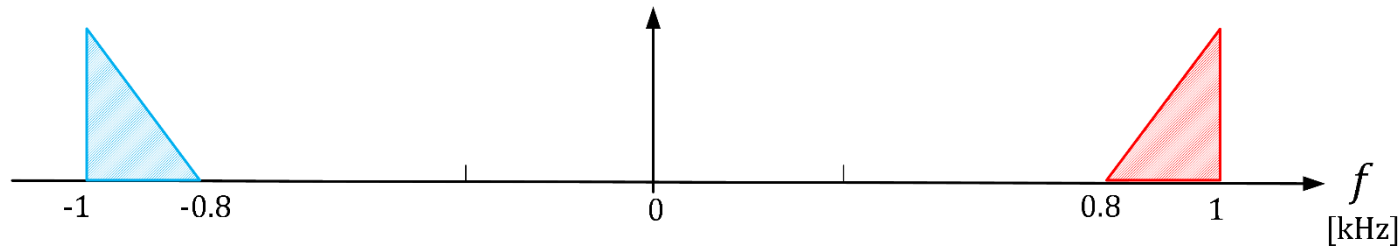
- Particularly, if $\phi = -\pi/2$, then $x(t) = \sin \omega_0 t$ and $x_r(t) = 0$



Aliasing

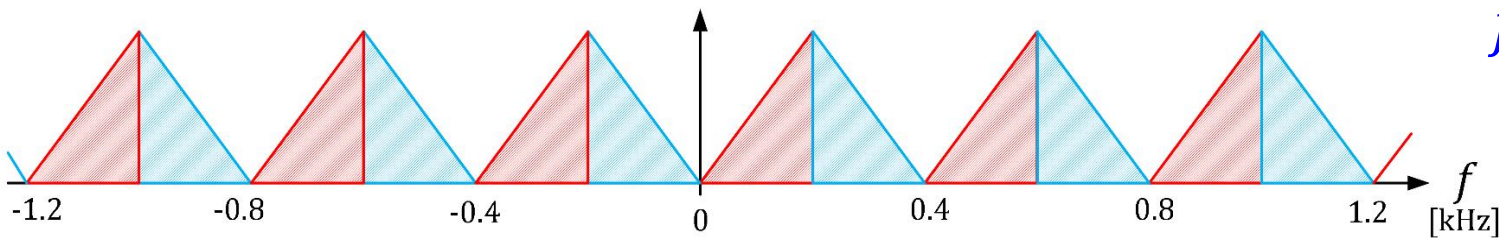
Aliasing

□ For signal with $f_c > B/2$, where $f_c = (f_h + f_l)/2$ and $B = f_h - f_l$



$$f_l = 800 \text{ Hz}, f_h = 1000 \text{ Hz}$$

Determine the lowest f_s with no aliasing



$$f_s = 400 \text{ Hz}$$

Q: What about $f_l = 850 \text{ Hz}$?

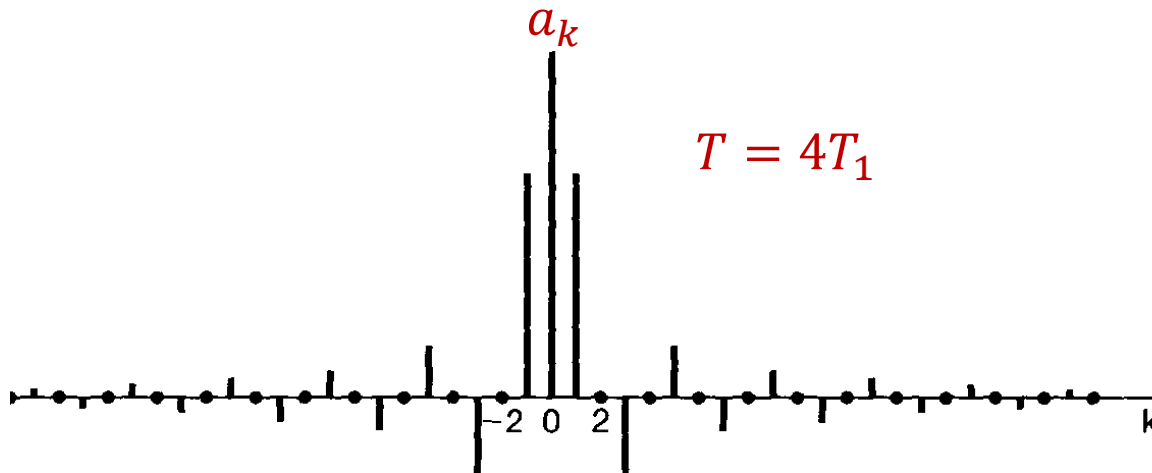
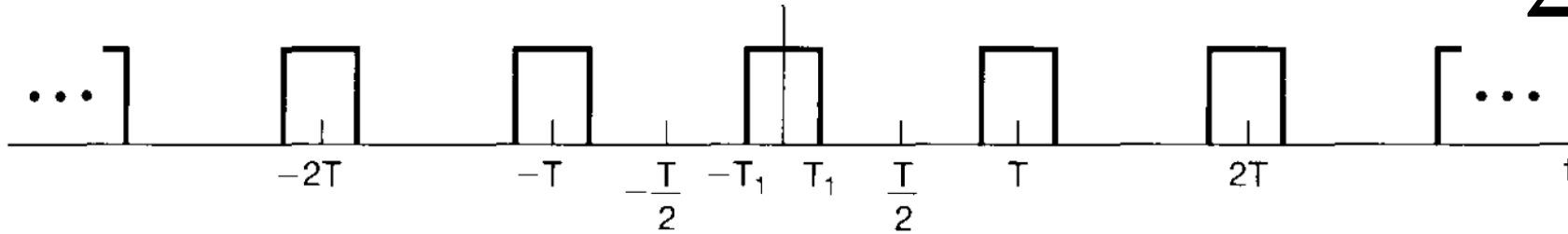


Aliasing

Aliasing

□ For harmonic related signal, e.g., a square wave

$$x(t) = \sum_{k=-\infty}^{\infty} a_k e^{jk\omega_0 t}$$



- $\omega_s > 2K\omega_0$, with K the k th harmonics you want to include
- Low-pass filtering before sampling

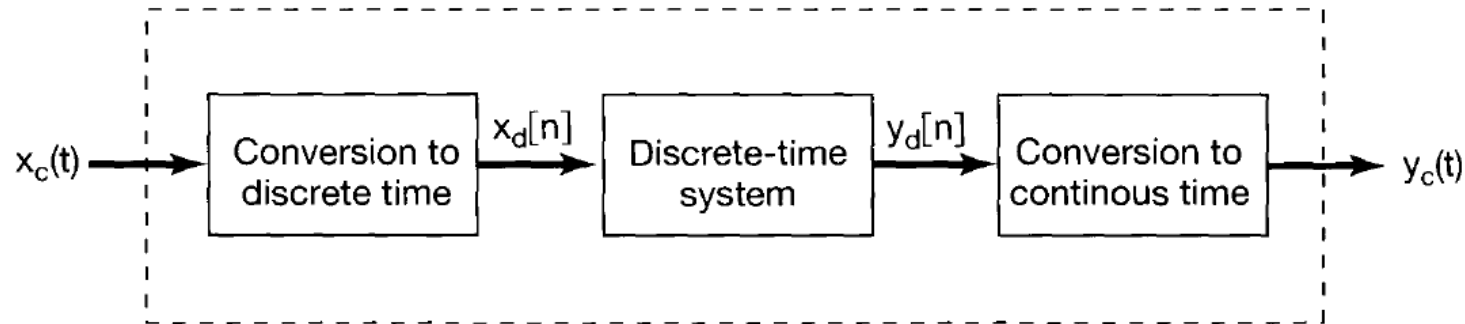
Chapter 7: Sampling

- ❑ Representation of a Continuous-Time Signal by Its Samples: The Sampling Theorem
- ❑ Reconstruction of a Signal from Its Samples Using Interpolation
- ❑ The Effect of Undersampling: Aliasing
- ❑ Discrete-Time Processing of Continuous-Time Signals
- ❑ Sampling of Discrete-Time Signals



Discrete-Time Processing of Continuous-Time Signals

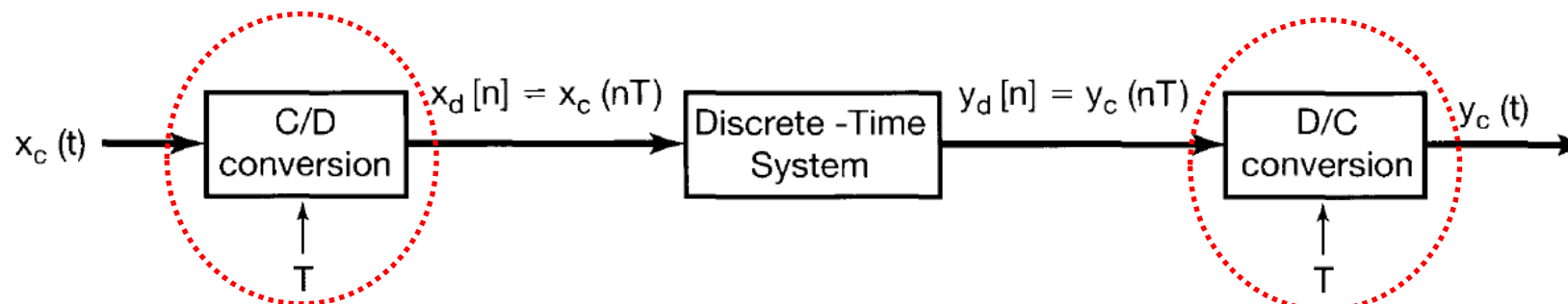
General scheme



$$x_d[n] = x_c(nT)$$

$$y_d[n] = y_c(nT)$$

- C/D: continuous-to-discrete-time conversion
- D/C: discrete-to-continuous-time conversion



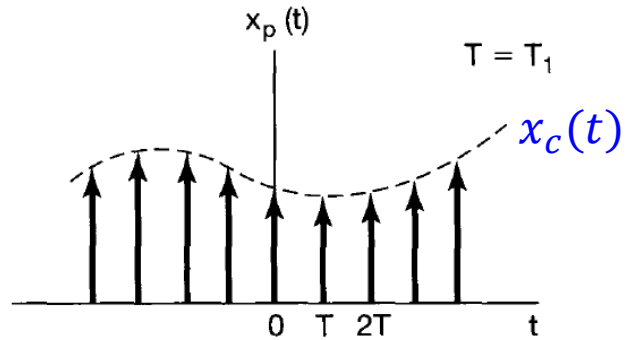
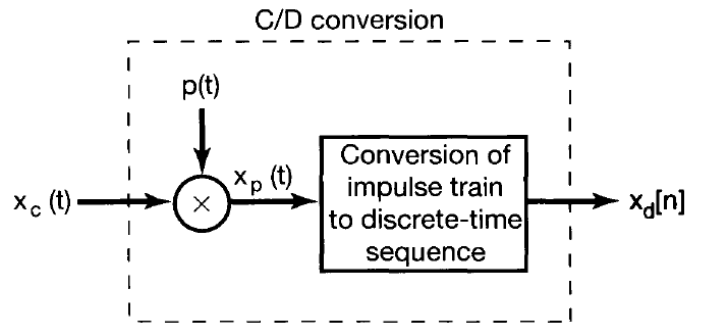
analog-to-digital (A/D) in practice

digital-to-analog (D/A) in practice

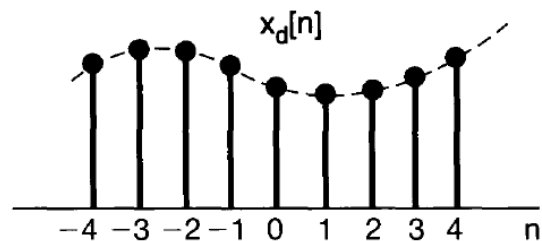


Discrete-Time Processing of Continuous-Time Signals

C/D conversion Time domain



$$x_p(t) = \sum_{n=-\infty}^{\infty} x_c(nT) \cdot \delta(t - nT)$$

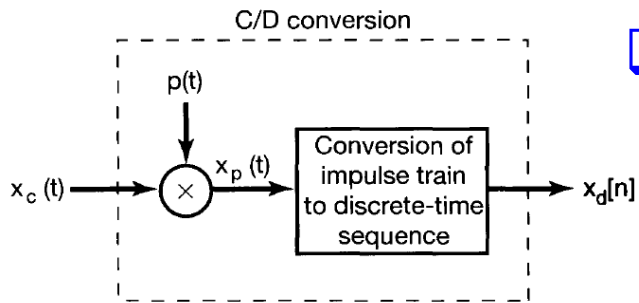


$$x_d[n] = x_c(nT)$$



Discrete-Time Processing of Continuous-Time Signals

C/D conversion Frequency domain: ω for continuous time and Ω for discrete time

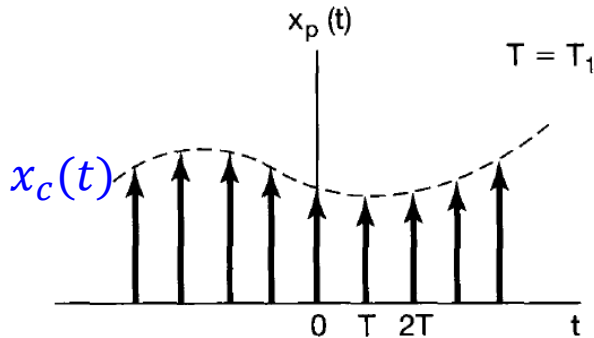


□ Spectrum of $x_d[n]$

$$X_d(e^{j\Omega}) = \sum_{n=-\infty}^{\infty} x_d[n] e^{-jn\Omega} = \sum_{n=-\infty}^{\infty} x_c(nT) e^{-jn\Omega}$$

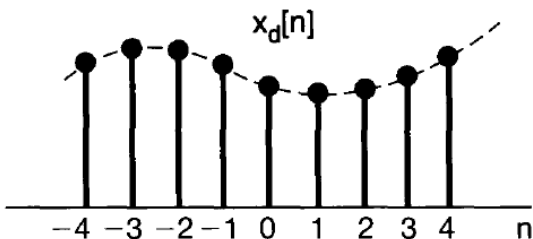
□ Spectrum of $x_p(t)$

$$x_p(t) = \sum_{n=-\infty}^{\infty} x_c(nT) \cdot \delta(t - nT) \Rightarrow X_p(j\omega) = \sum_{n=-\infty}^{\infty} x_c(nT) \cdot e^{-j\omega nT}$$



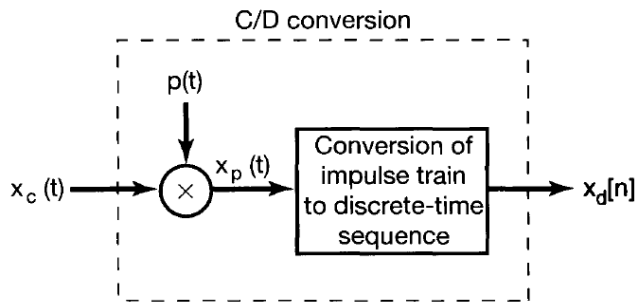
□ If $\omega = \Omega/T$, $X_d(e^{j\Omega}) = X_p(j\Omega/T)$

□ The spectrum of $x_d[n]$ can be obtained from $X_p(j\omega)$ by replacing ω with Ω/T .



Discrete-Time Processing of Continuous-Time Signals

C/D conversion Frequency domain: ω for continuous time and Ω for discrete time



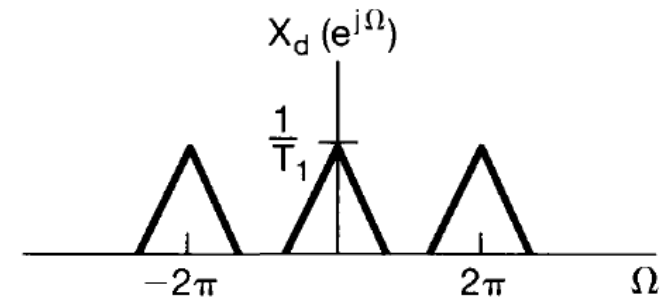
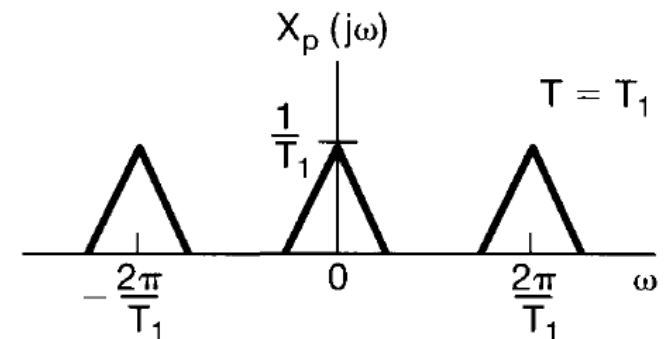
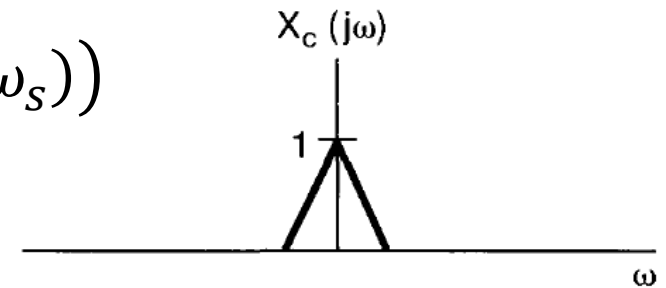
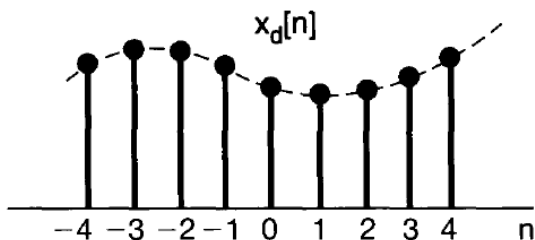
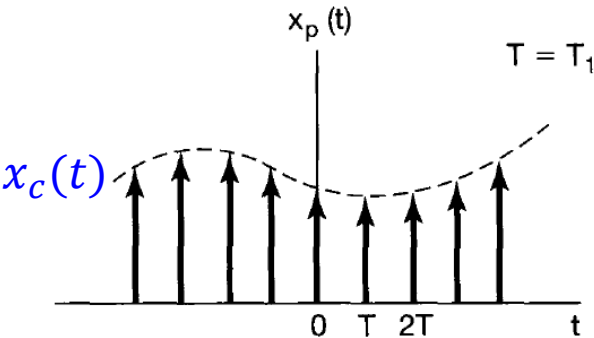
□ Recall: $X_p(j\omega) = \frac{1}{T} \sum_{K=-\infty}^{\infty} X(j(\omega - k \cdot \omega_s))$

$$\therefore X_d(e^{j\Omega}) = \frac{1}{T} \sum_{K=-\infty}^{\infty} X(j(\Omega - 2k\pi)/T)$$

□ $X_d(e^{j\Omega})$ is a frequency-scaled version of $X_p(j\omega)$, and is periodic with period of 2π

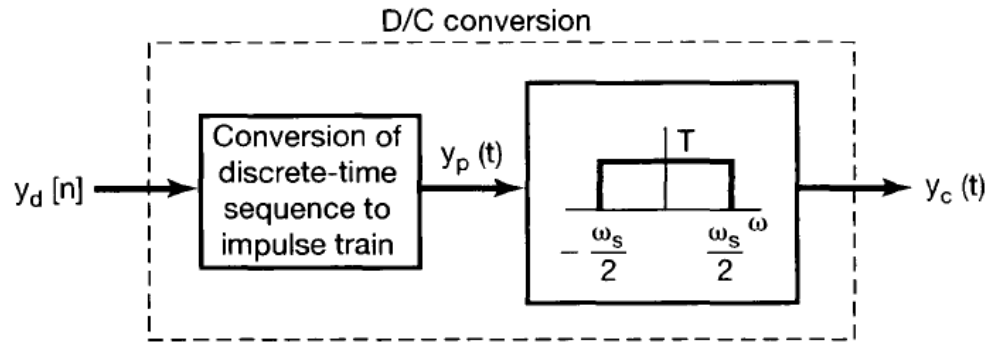
□ Informally

- t to n : time scaling by $1/T$
- ω to Ω : frequency scaling by T



Discrete-Time Processing of Continuous-Time Signals

D/C conversion



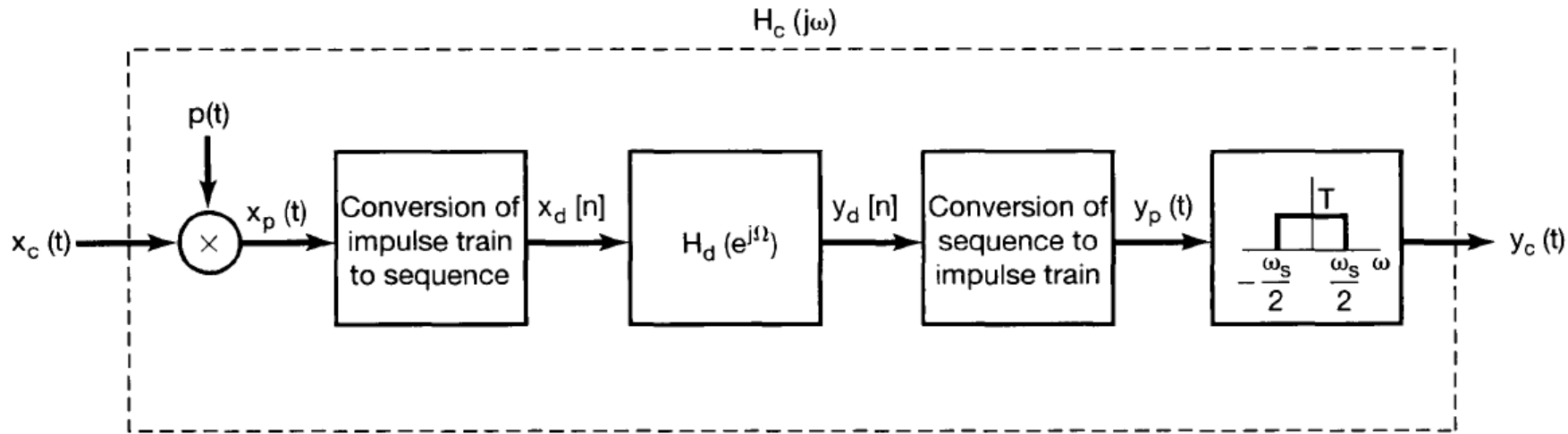
□ $Y_d(e^{j\Omega})$: Spectrum of $y_d[n]$

□ $Y_p(j\omega)$: Spectrum of $y_p(t)$

□ $Y_p(j\omega)$ can be obtained from $Y_d(e^{j\Omega})$ by replacing Ω with ωT .

Discrete-Time Processing of Continuous-Time Signals

Overall system



□ $x_c(t)$: input

□ $y_c(t)$: output

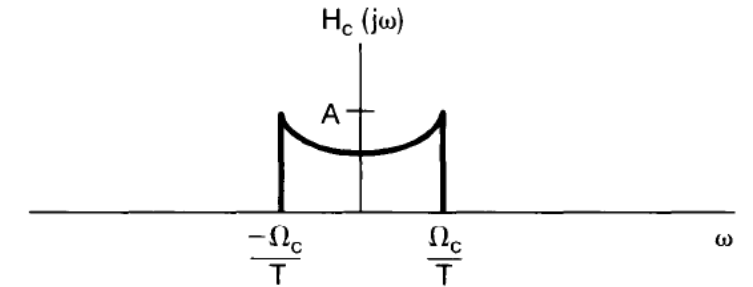
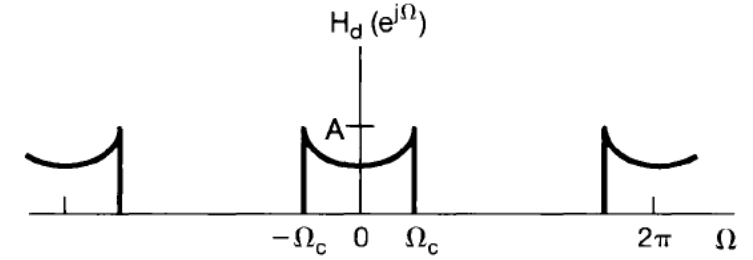
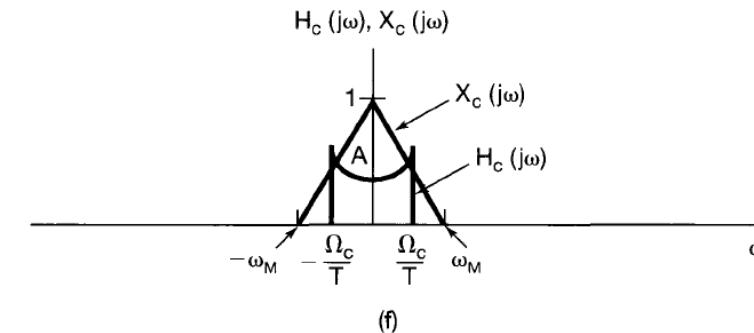
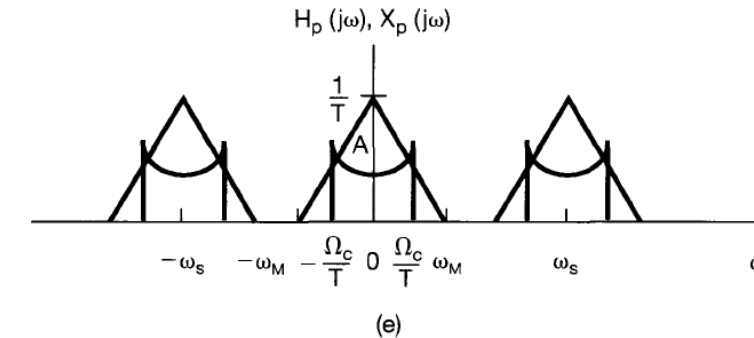
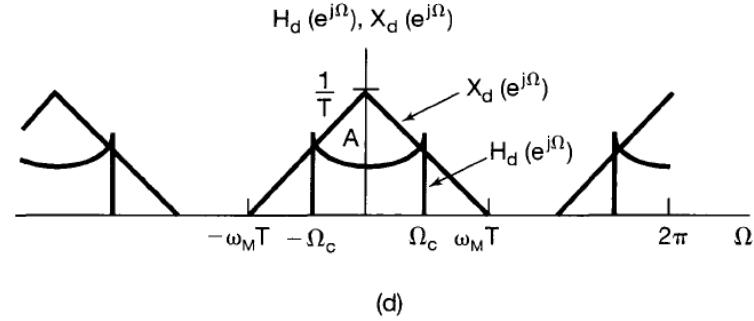
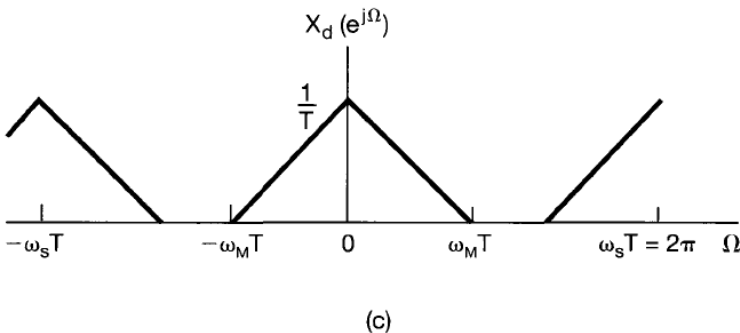
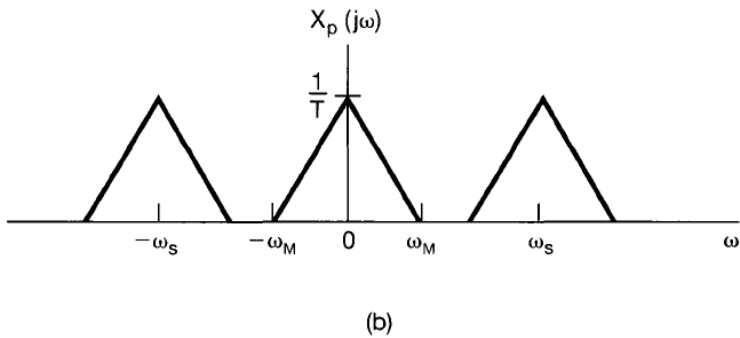
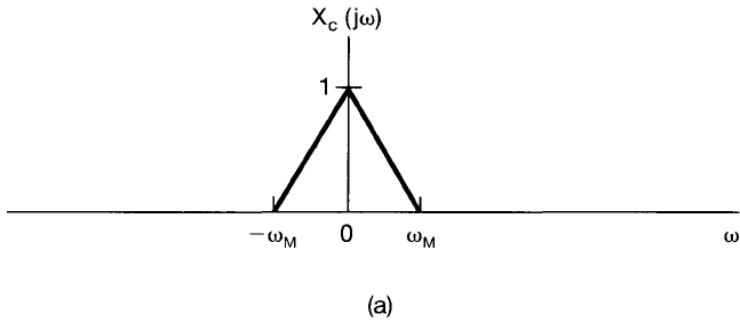
□ The overall system is equivalent to a continuous-time system with frequency response $H_c(j\omega)$

□ $H_c(j\omega) = ?$



Discrete-Time Processing of Continuous-Time Signals

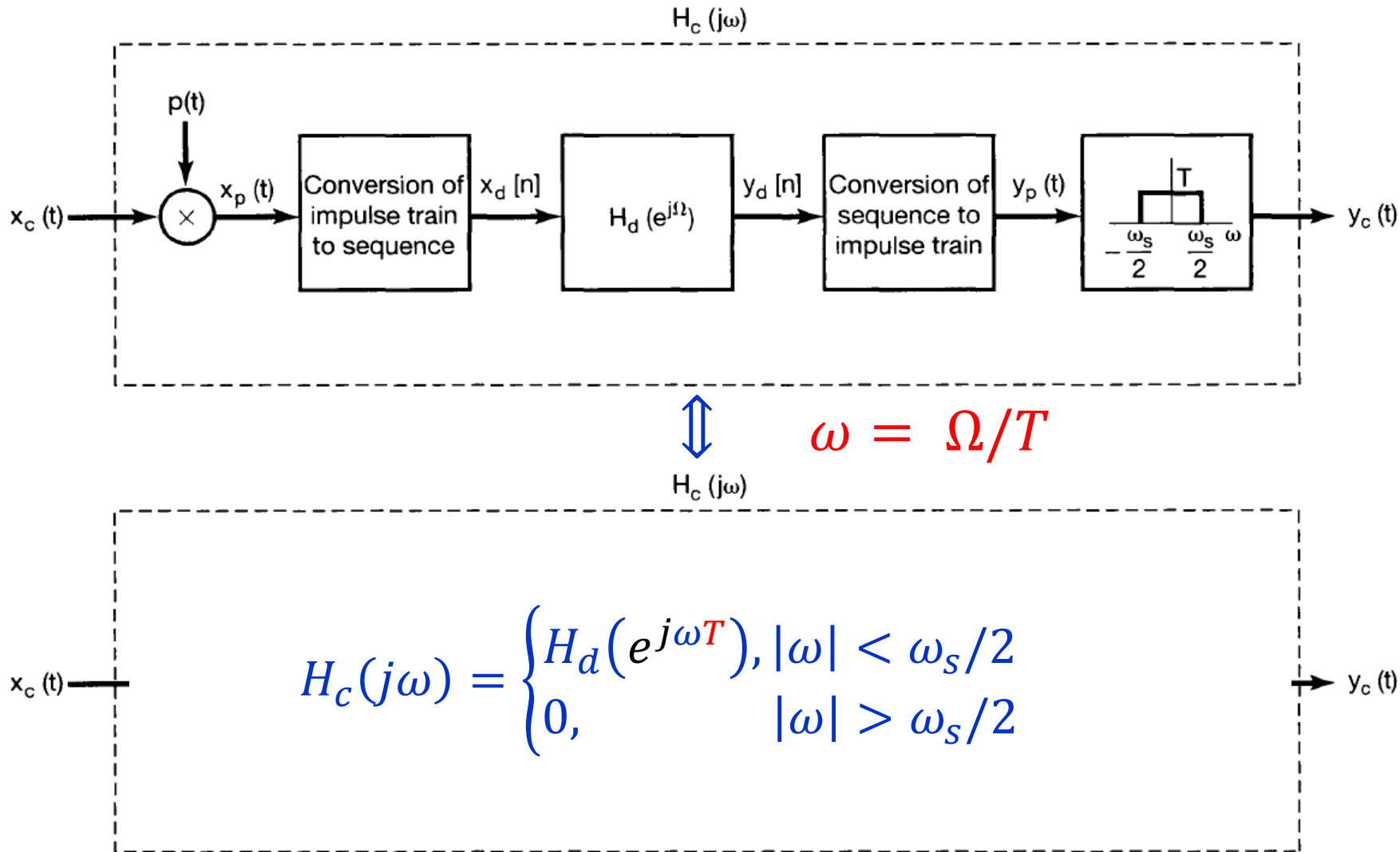
Overall system



$$H_c(j\omega) = \begin{cases} H_d(e^{j\omega T}), & |\omega| < \omega_s/2 \\ 0, & |\omega| > \omega_s/2 \end{cases}$$

Discrete-Time Processing of Continuous-Time Signals

Overall system

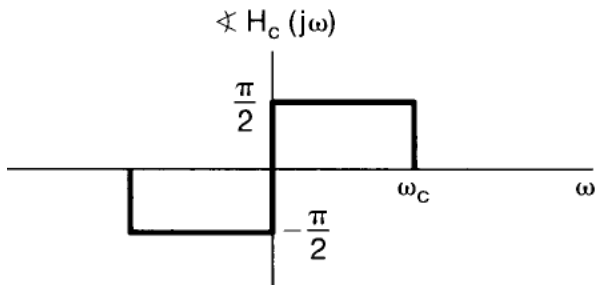
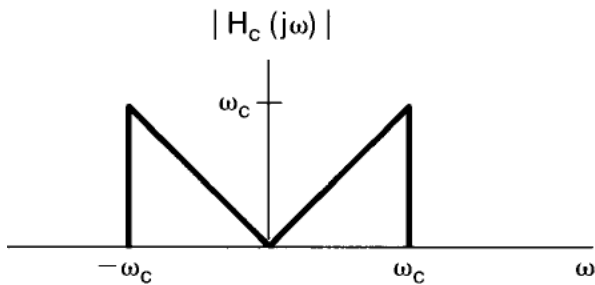


Discrete-Time Processing of Continuous-Time Signals

Digital differentiator: frequency response

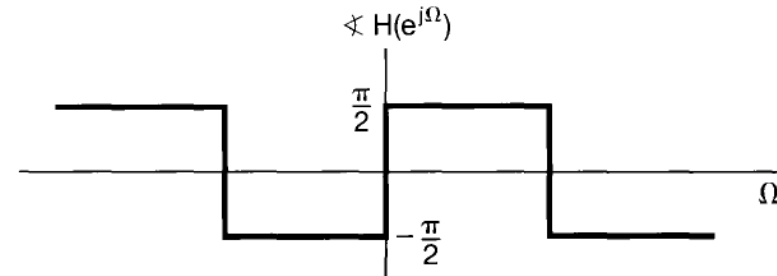
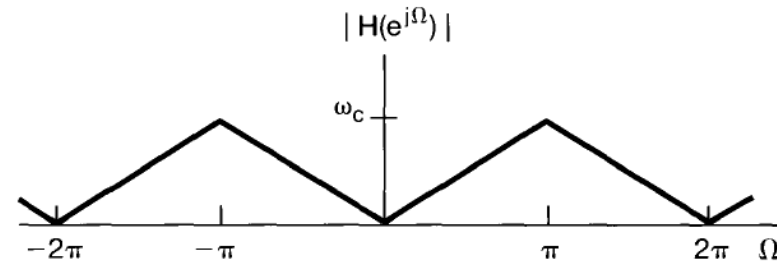
□ Band-limited CT differentiator \Leftrightarrow □ Corresponding DT differentiator

$$H_c(j\omega) = \begin{cases} j\omega, & |\omega| < \omega_c \\ 0, & |\omega| > \omega_c \end{cases}$$



$$H_d(e^{j\Omega}) = j \frac{\Omega}{T}, |\Omega| < \pi$$

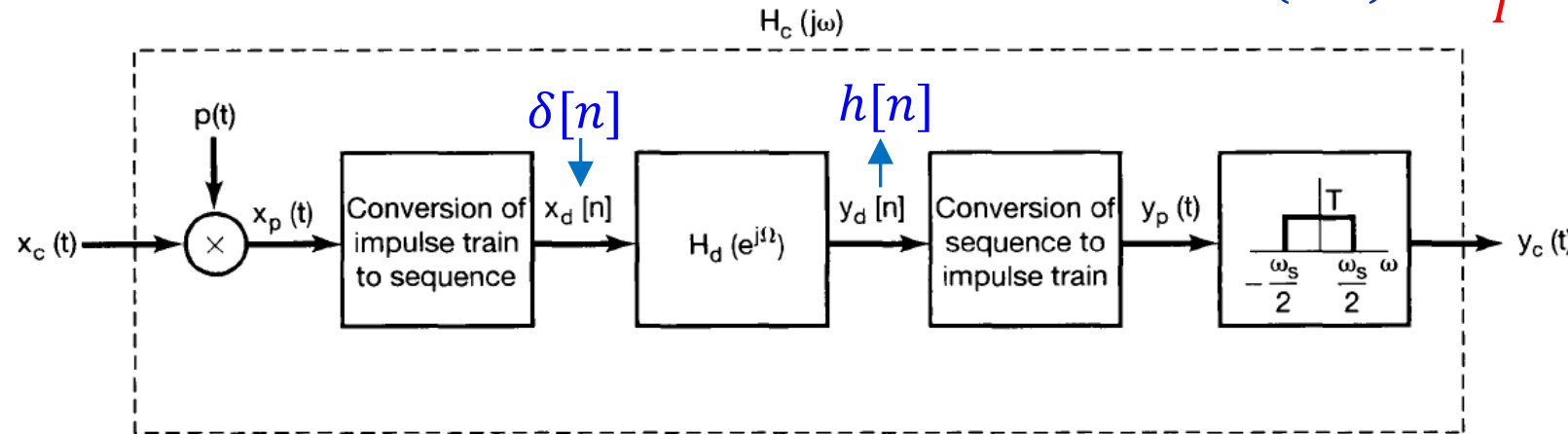
$$\omega_c = \omega_s/2$$



Discrete-Time Processing of Continuous-Time Signals

Digital differentiator: impulse response

$$H_d(e^{j\Omega}) = j\frac{\Omega}{T}, |\Omega| < \pi$$



$$\square x_c(t) = \frac{\sin(\pi t/T)}{\pi t} \Rightarrow x_d[n] = x_c(nT) = \frac{1}{T} \delta[n]$$

$$\square y_d[n] = y_c(nT) \quad y_c(t) = \frac{d}{dt} x_c(t) = \frac{\cos(\pi t/T)}{Tt} - \frac{\sin(\pi t/T)}{\pi t^2}$$

$$\square y_d[n] = \begin{cases} \frac{(-1)^n}{nT^2}, & n \neq 0 \\ 0, & n = 0 \end{cases}$$

$$\Rightarrow \therefore h_d[n] = \begin{cases} \frac{(-1)^n}{nT}, & n \neq 0 \\ 0, & n = 0 \end{cases}$$



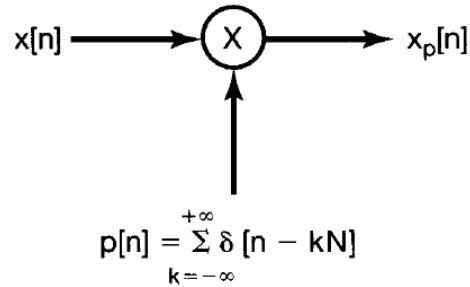
Chapter 7: Sampling

- ❑ Representation of a Continuous-Time Signal by Its Samples: The Sampling Theorem
- ❑ Reconstruction of a Signal from Its Samples Using Interpolation
- ❑ The Effect of Undersampling: Aliasing
- ❑ Discrete-Time Processing of Continuous-Time Signals
- ❑ Sampling of Discrete-Time Signals

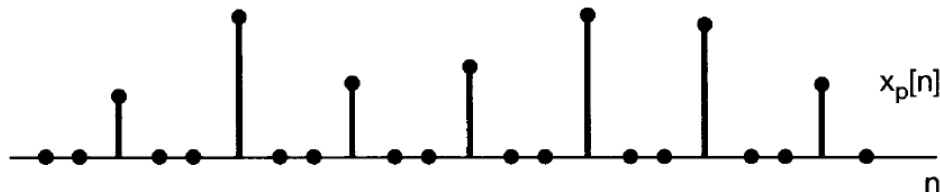
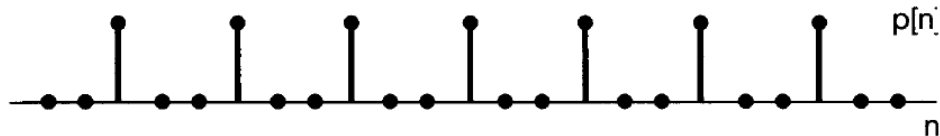
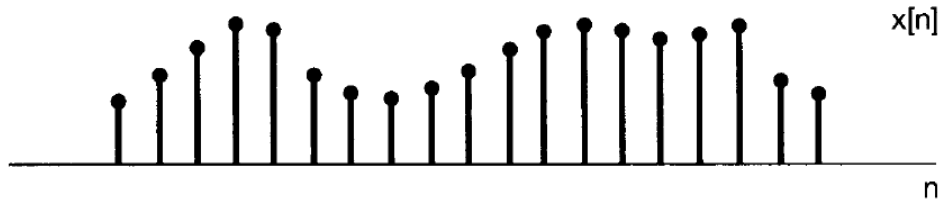


Sampling of Discrete-Time Signals

Impulse train sampling Time domain



N : sampling period



$$x_p[n] = x[n]p[n] = \sum_{k=-\infty}^{\infty} x[kN]\delta[n - kN]$$
$$= \begin{cases} x[n], & \text{if } n \text{ is an integer multiple of } N \\ 0, & \text{otherwise} \end{cases}$$



Sampling of Discrete-Time Signals

Impulse train sampling Frequency domain

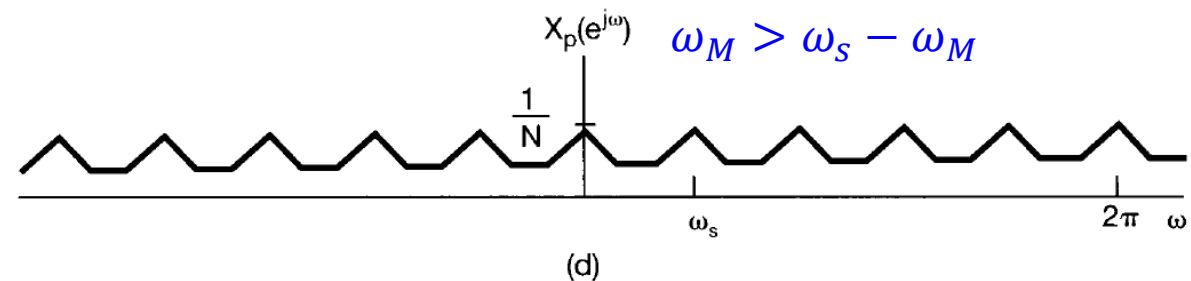
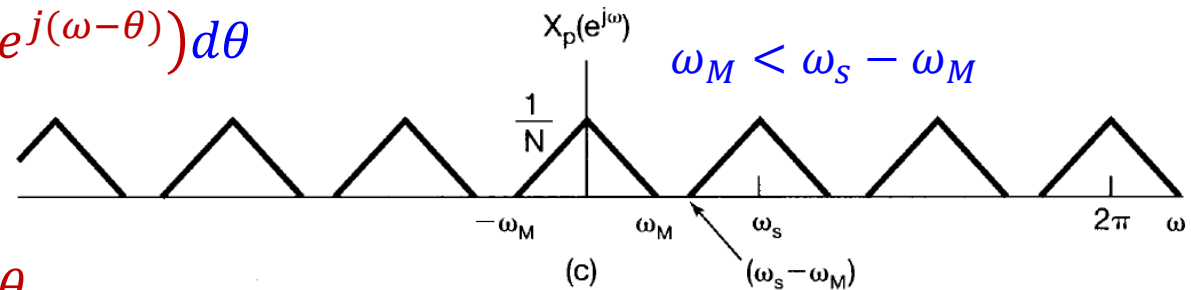
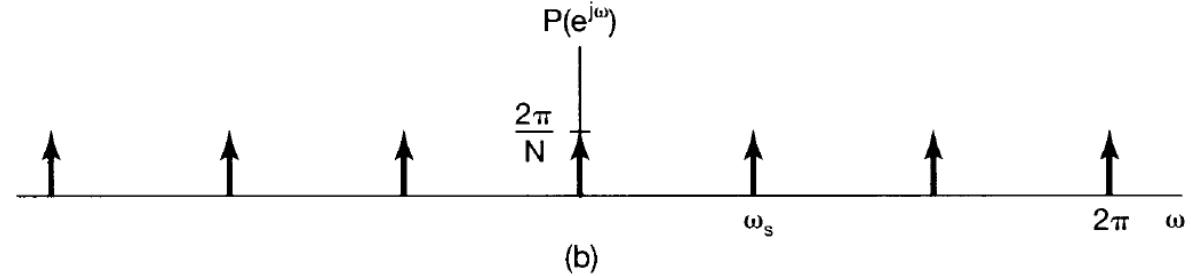
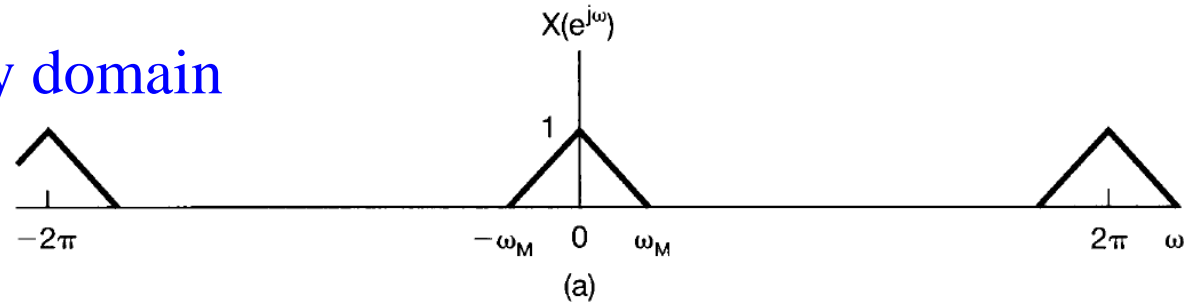
$$P(e^{j\omega}) = \frac{2\pi}{N} \sum_{K=-\infty}^{\infty} \delta(\omega - k\omega_s) \quad \omega_s = \frac{2\pi}{N}$$

$$X_p(e^{j\omega}) = \frac{1}{2\pi} \int_{2\pi} P(e^{j\theta}) X(e^{j(\omega-\theta)}) d\theta$$

$$= \frac{1}{2\pi} \cdot \frac{2\pi}{N} \int_{2\pi} \left[\sum_{k=-\infty}^{\infty} \delta(\theta - k\omega_s) \right] X(e^{j(\omega-\theta)}) d\theta$$

$$= \frac{1}{N} \sum_{k=0}^{N-1} \int_{2\pi} \delta(\theta - k\omega_s) X(e^{j(\omega-\theta)}) d\theta$$

$$X_p(e^{j\omega}) = \frac{1}{N} \sum_{k=0}^{N-1} X(e^{j(\omega-k\omega_s)})$$



Sampling of Discrete

Impulse train sampling Reconstruction of $x[n]$

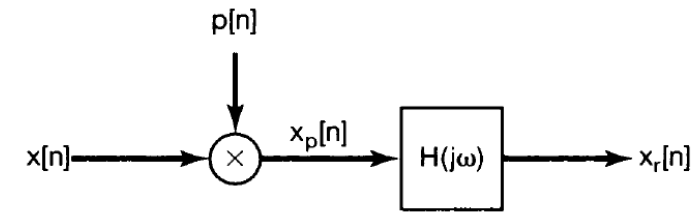
$$x_r[n] = x_p[n] * h[n] \quad \text{Time domain}$$

$$= \left[\sum_{k=-\infty}^{\infty} x[kN] \cdot \delta[n - kN] \right] * h[n]$$

$$= \sum_{k=-\infty}^{\infty} x[kN] [\delta[n - kN] * h[n]]$$

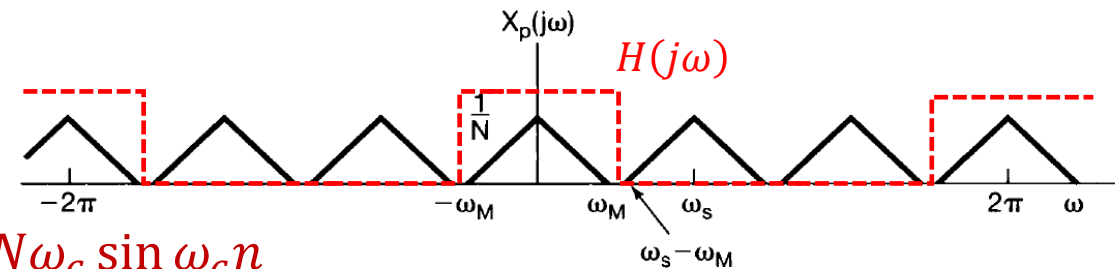
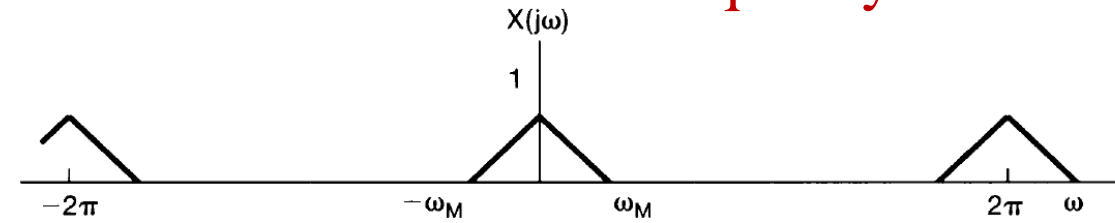
$$= \sum_{k=-\infty}^{\infty} x[kN] h[n - kN]$$

$$x_r[n] = \sum_{k=-\infty}^{\infty} x[kN] \frac{N\omega_c}{\pi} \frac{\sin \omega_c(n - kN)}{\omega_c(n - kN)}$$

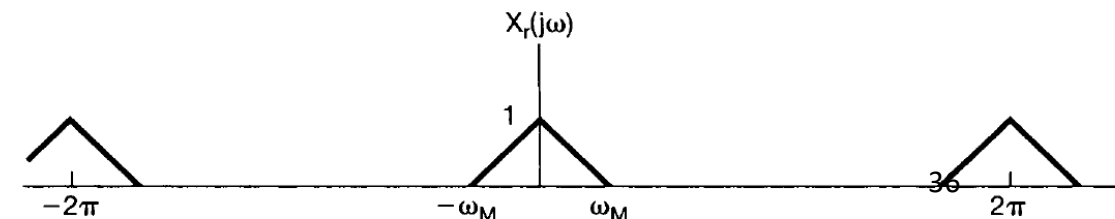
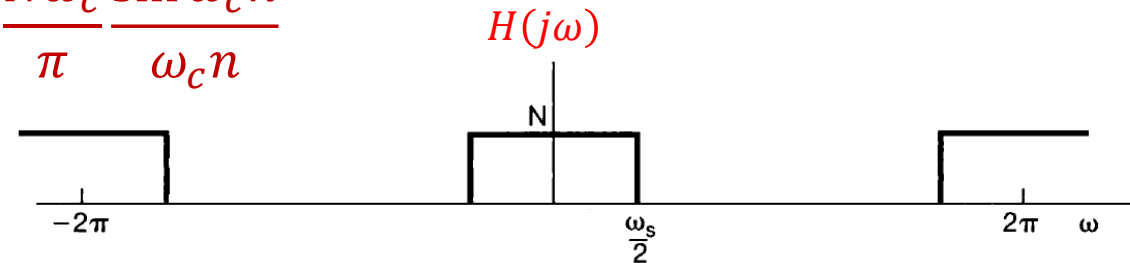


(a)

Frequency domain



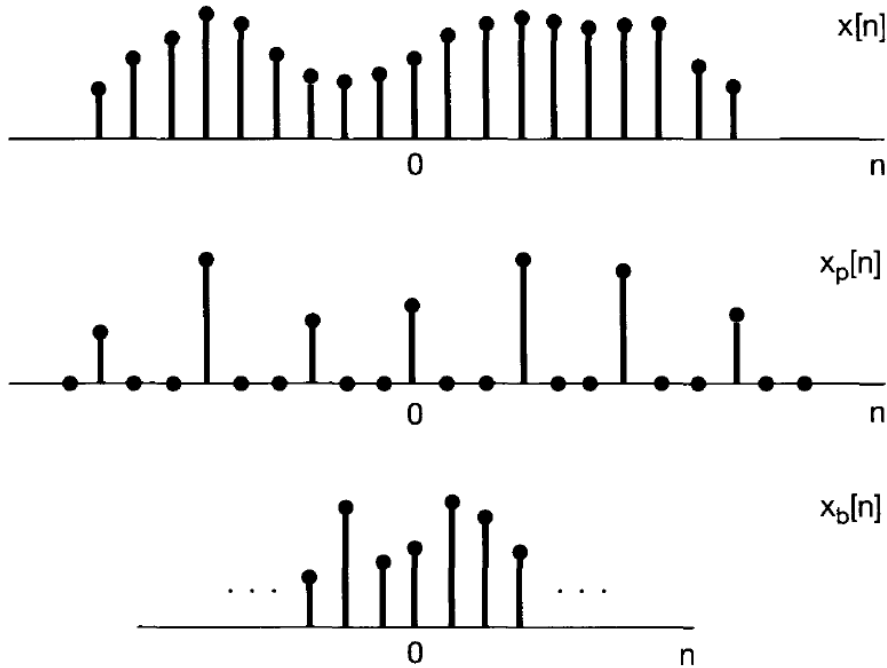
$$h[n] = \frac{N\omega_c}{\pi} \frac{\sin \omega_c n}{\omega_c n}$$



Sampling of Discrete-Time Signals

Decimation (sample rate decrease, SRD)

Time domain



$$x_b[n] = x_p[nN]$$

$$x_b[n] = x[nN]$$

Frequency domain

$$X_b(e^{j\omega}) = \sum_{k=-\infty}^{\infty} x_b[k] e^{-j\omega k}$$

$$X_b(e^{j\omega}) = \sum_{k=-\infty}^{\infty} x_p[kN] e^{-j\omega k} \quad n = kN$$

$$X_b(e^{j\omega}) = \sum_{\substack{n=\text{integer} \\ \text{number of } N}} x_p[n] e^{-j\omega n/N}$$

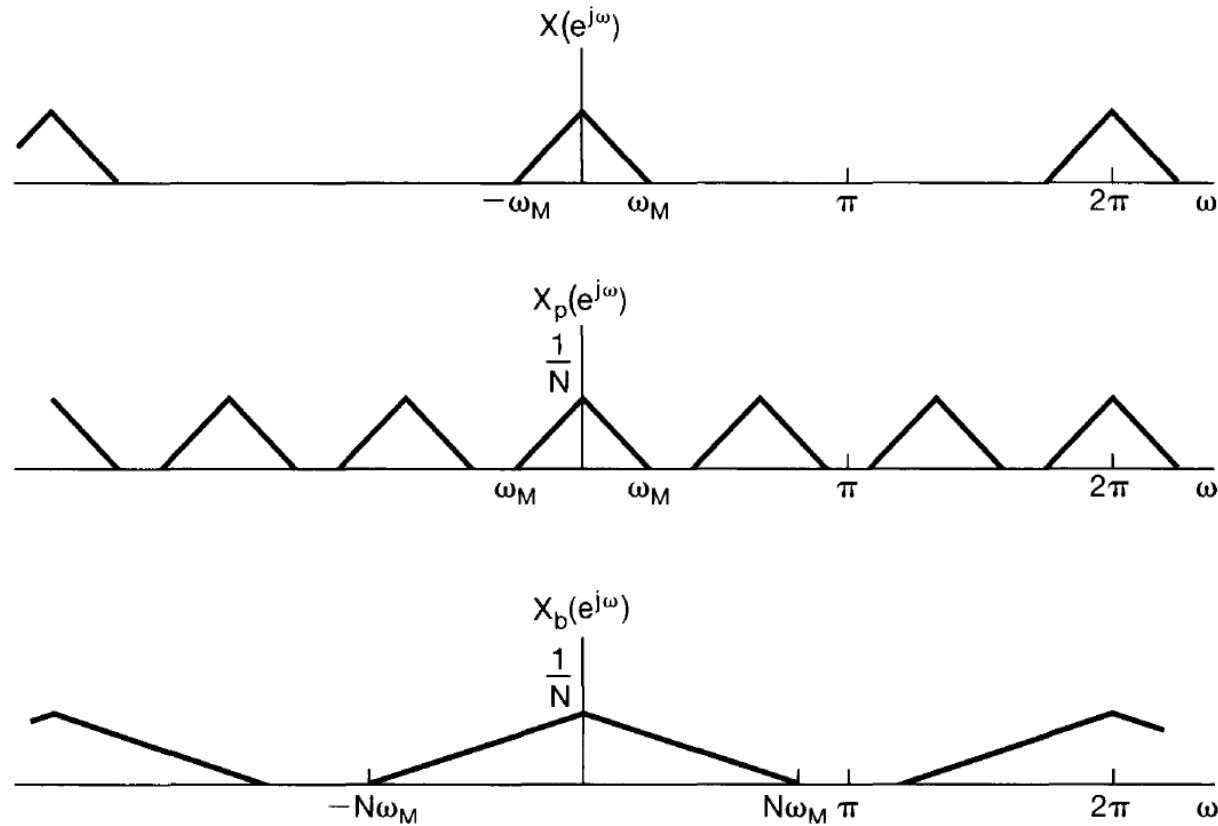
$$X_b(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x_p[n] e^{-j\omega n/N}$$

$$X_b(e^{j\omega}) = X_p(e^{j\omega/N})$$



Sampling of Discrete-Time Signals

Decimation



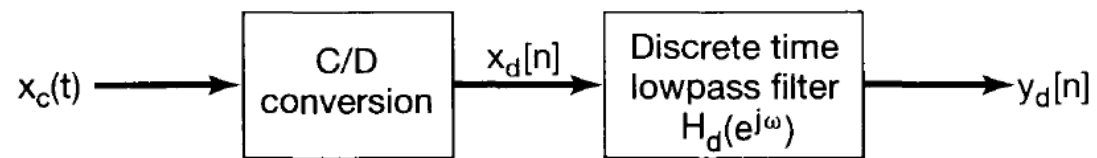
$$x_b[n] = x_p[nN]$$

$$X_b(e^{j\omega}) = X_p(e^{j\omega/N})$$

Low-pass filtering in advance, if $N\omega_M > \pi$

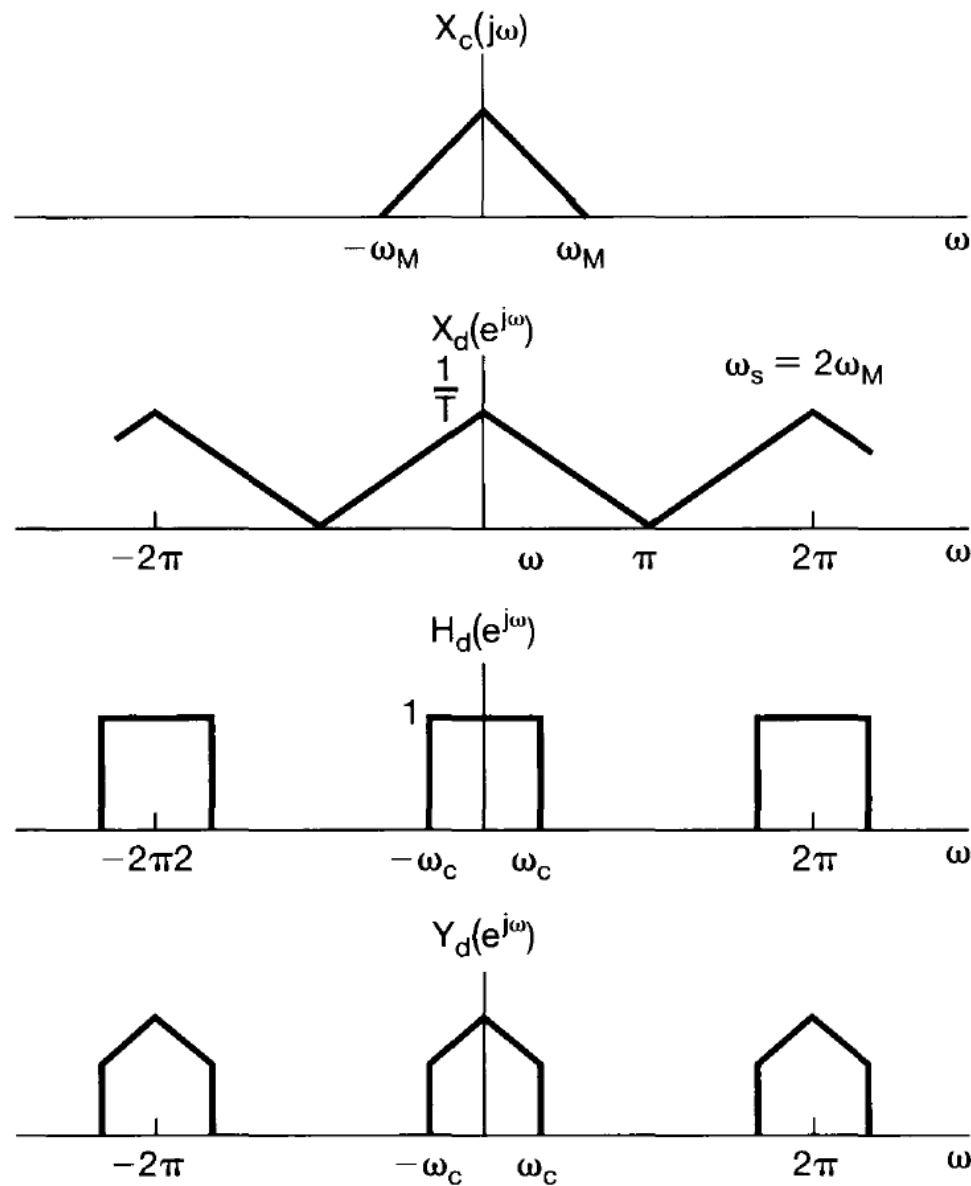


Sampling of



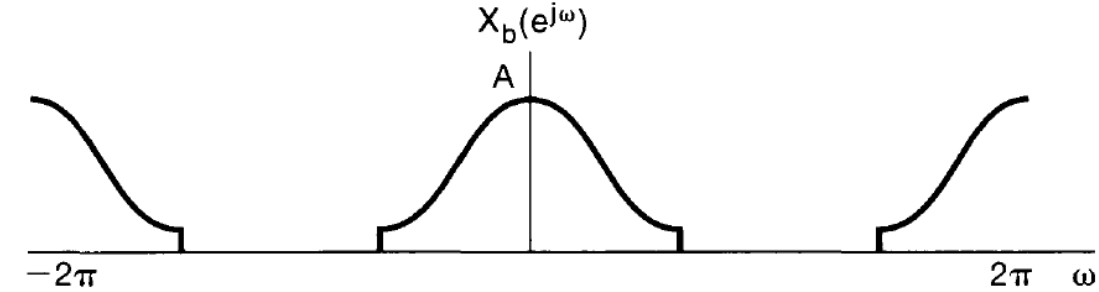
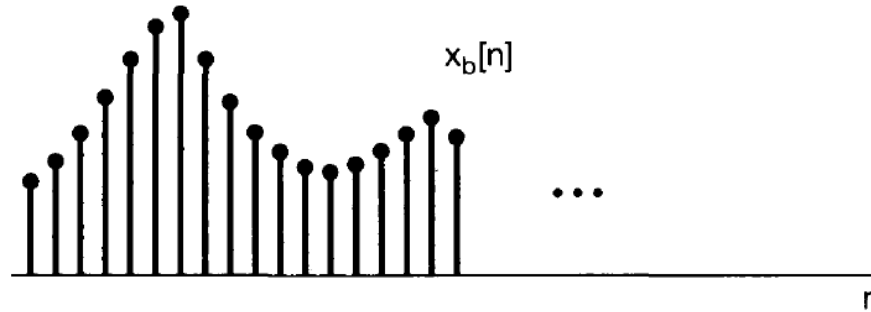
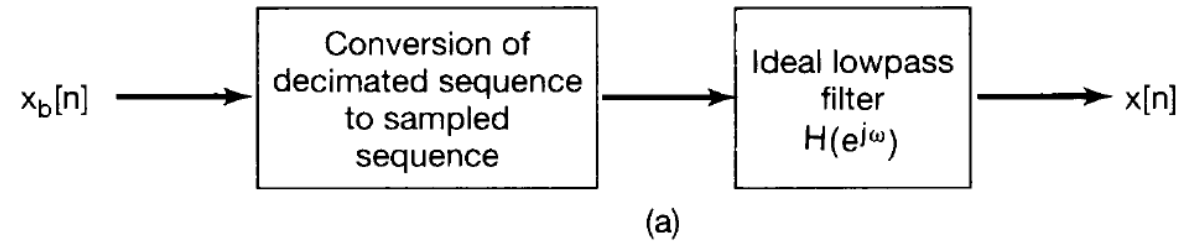
Decimation

- Prevent aliasing by LPF in front of SRD \Rightarrow Decimator

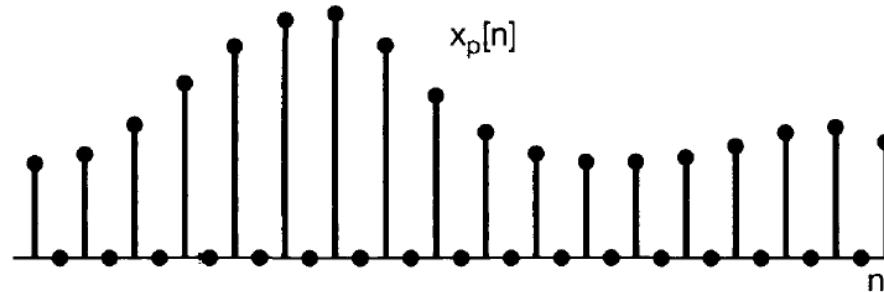


S

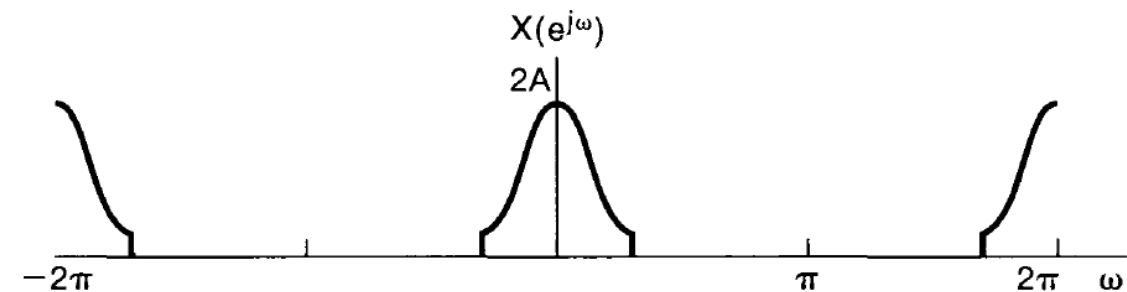
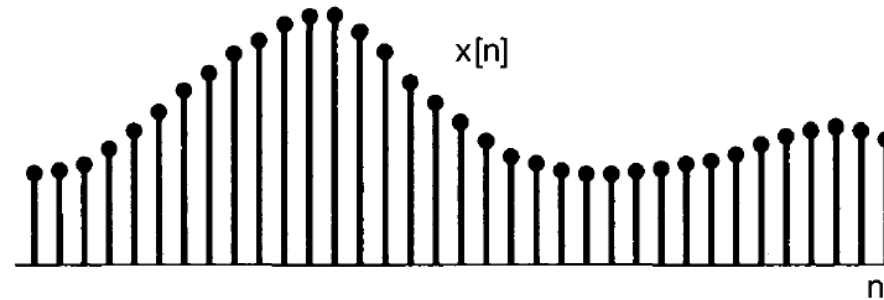
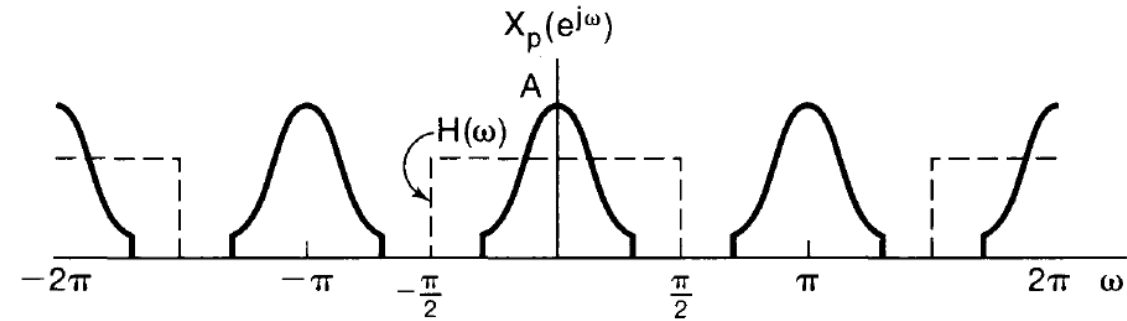
Interpolation (SRI)



- Prevent mirrors by LPF after SRI \Rightarrow Interpolator



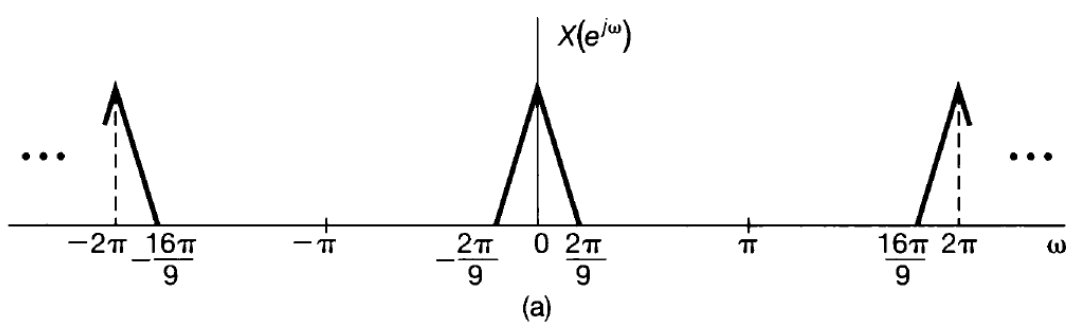
Upsampling by a factor of 2



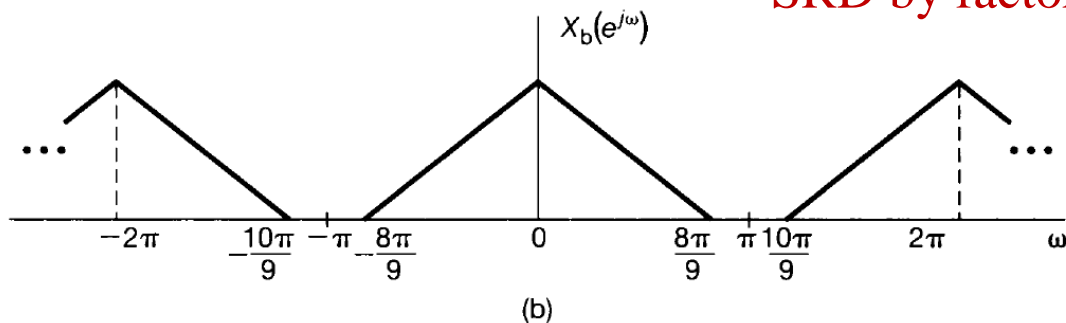
Sampling

als

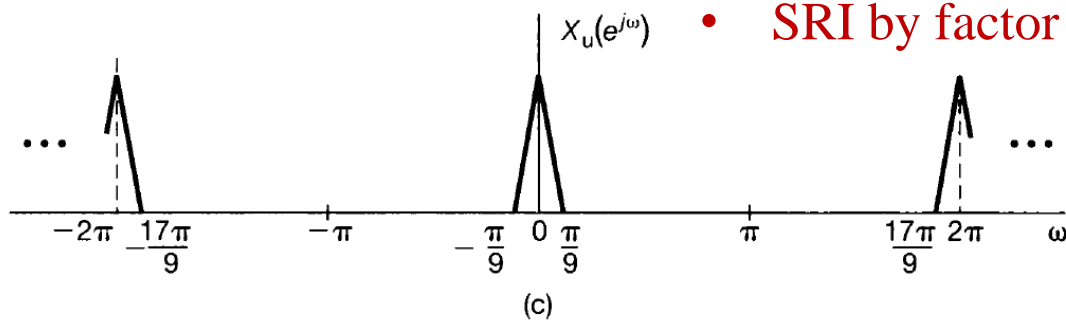
Interpolation (SRI)



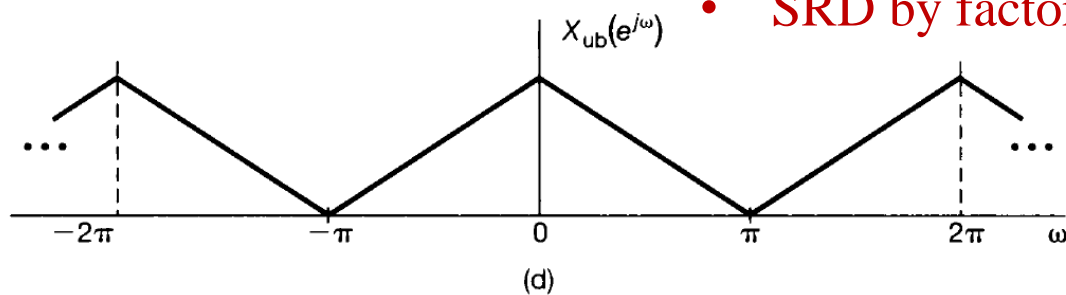
- SRD by factor of 4



- SRI by factor of 2



- SRD by factor of 9



- Overall, SRD by factor of 4.5

