Periodic (Uniform) Sampling

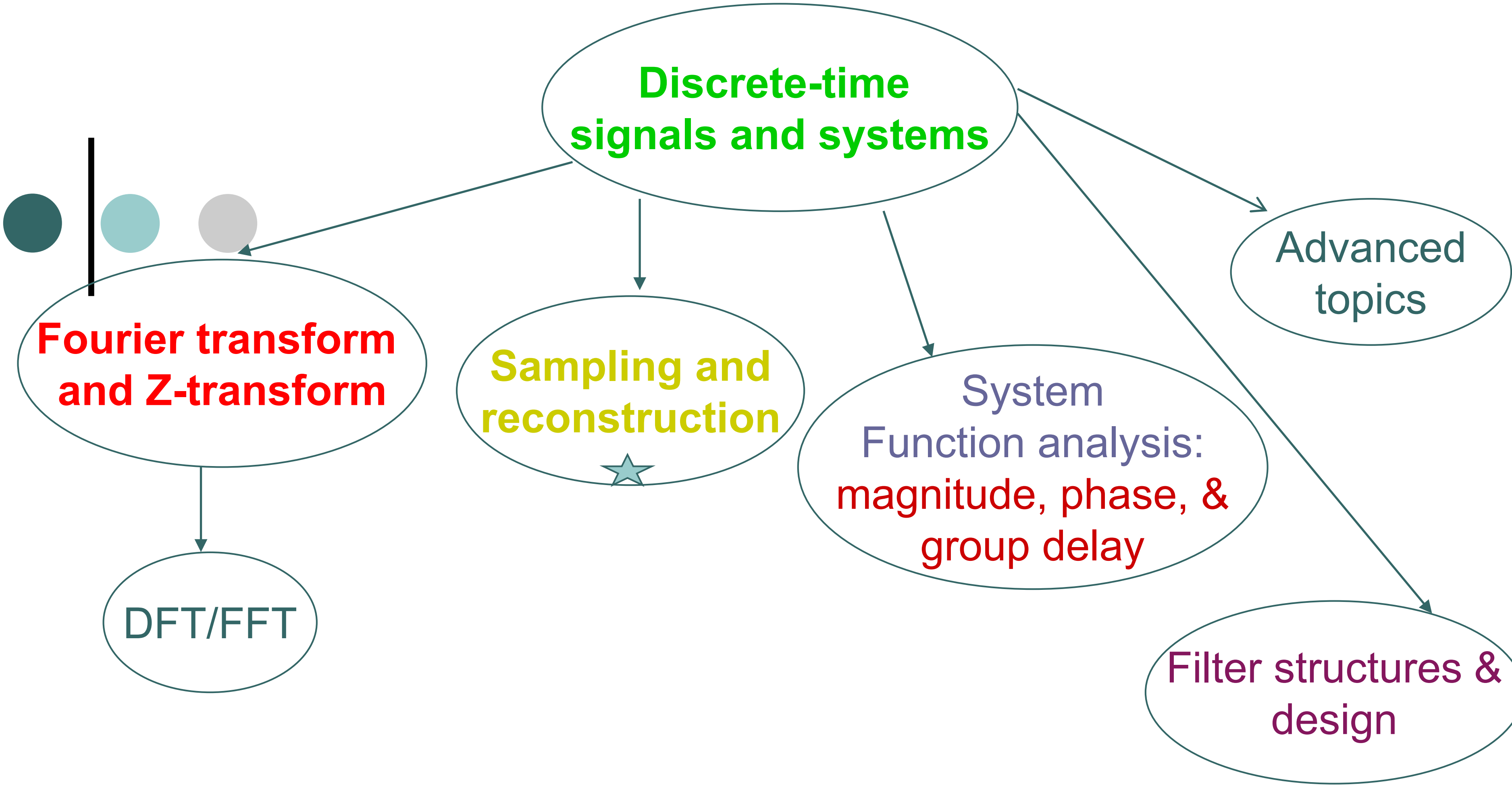
- Introduction
- Sampling: Representation of a CT Signal by Its Samples
- Reconstruction of a Signal from Its Samples
- Aliasing: The Effect of under-sampling
- Examples
- DT Processing of CT Signals
- Real sampling: A/D and D/A Conversion
- Sampling of DT Signals (Down-sampling & up-sampling)
- Quantization (if time permits)
- Summary

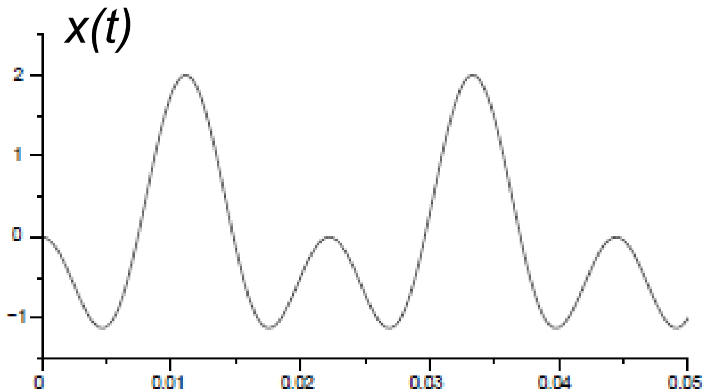
M. Amer
Concordia University
Electrical and Computer Engineering

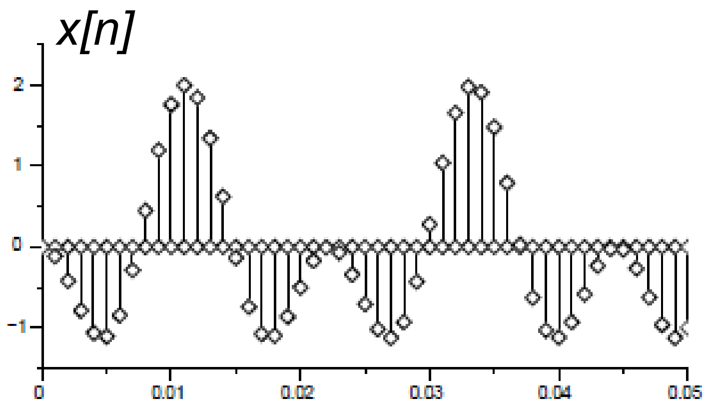
Content and Figures are based on:

- A. Oppenheim, A.S. Willsky and S.H. Nawab, Signals and Systems (S&S), 2nd Edition, Prentice-Hall, 1997
- Oppenheim, Shafer, Discrete-Time Signal Processing (DTSP), 3e,
- Dr. Güner Arslan, 351M Digital Signal Processing, http://signal.ece.utexas.edu/~arslan/courses/dsp
- Dr. Zheng-Hua Tan, Digital Signal Processing III, 2009, http://kom.aau.dk/~zt/cources/DSP/

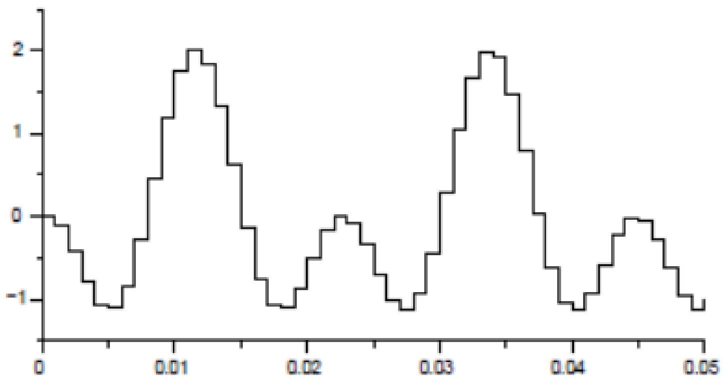
Course at a glance



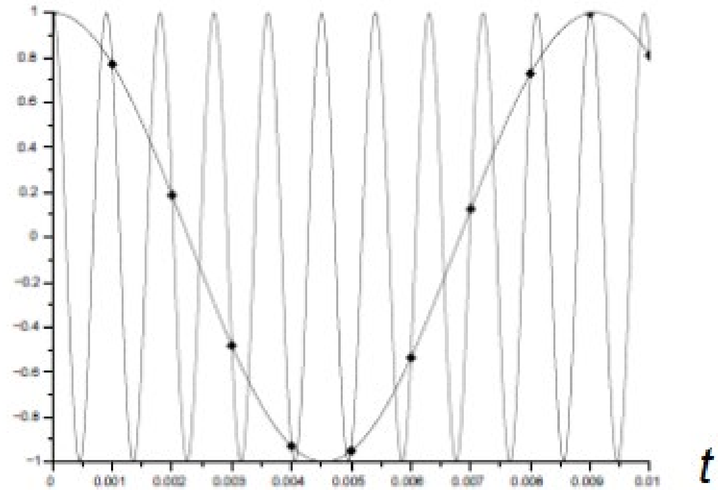




The result of Sampling



The result of reconstruction

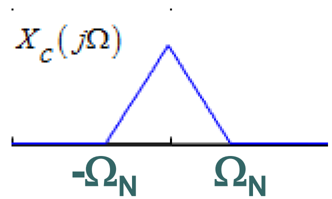


Not good sampling:
2 sine signals; same number of samples

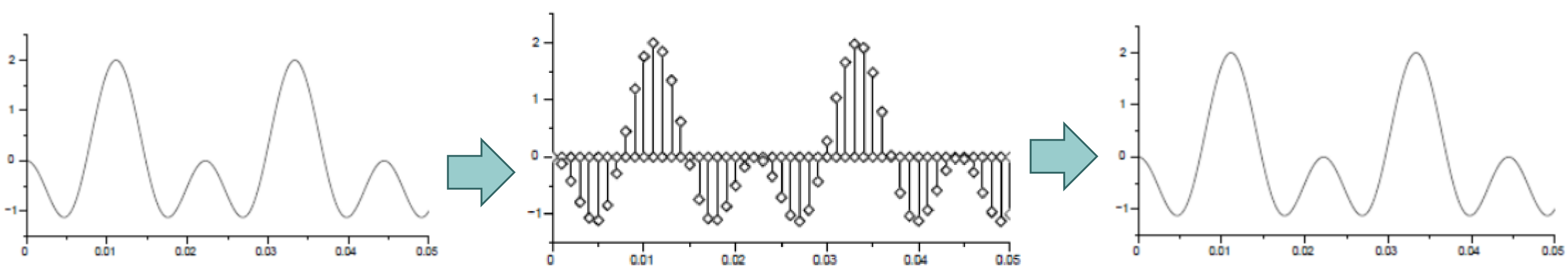
➔ Aliasing (under-sampling):
cannot reconstruct

The Sampling (Nyquist) Theorem

A continuous-time signal **xc(t)**, whose spectral content is **limited to frequencies** smaller than Ω_N , i.e., it is band-limited to $|\Omega| < \Omega_N$



can be **perfectly** recovered from its sampled version *x[n]*, if the **sam** $\Omega_s > 2\Omega_N$ **rate Ω_s is larger than twice the maximum frequency Ω_s**, i.e., if

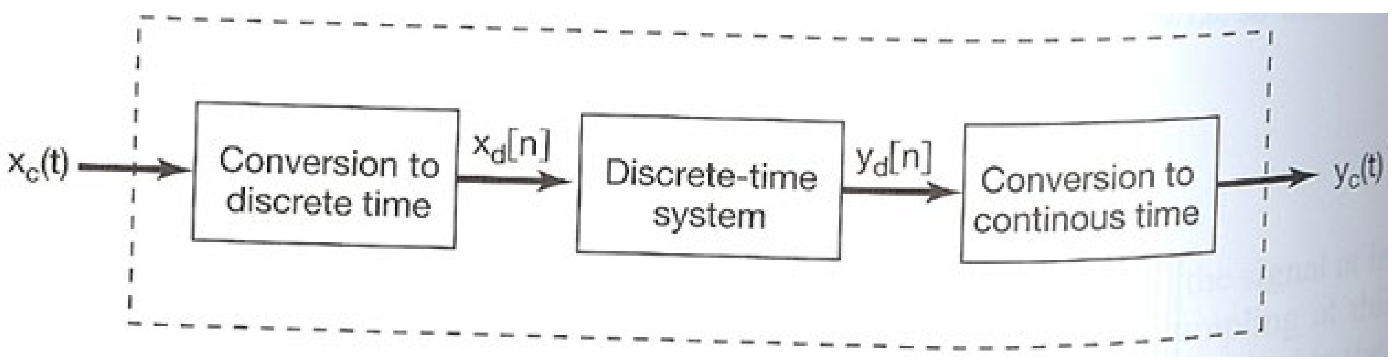


Introduction

- Sampling is an operation that transforms a CT signal $x_c(t)$ into a DT signal $x[n]$

  ➔x[n] gives the values of x(t) read at intervals of T seconds $x[n]=x_c(nT)$

- Why sampling ?
  - With $x[n]$ we can take advantage of DT systems (DSP) technologies to process them



Sampling: Applications

- **Audio sampling:**
  - Human hearing: 20–20,000 Hz range
  - Sampling rate is at
    - 44.1 kHz (CD), 48 kHz (professional audio), or 96kHz
- **Speech sampling:**
  - The energy of human speech: 5Hz - 4 kHz range
  - → Sampling rate: 8 kHz (used by nearly all telephony systems)
- **Video sampling:**
  - Standard-definition television (SDTV): 720x480 pixels (N.A.) or 704x576 pixels (UE)
  - High-definition television (HDTV): 1440x1080

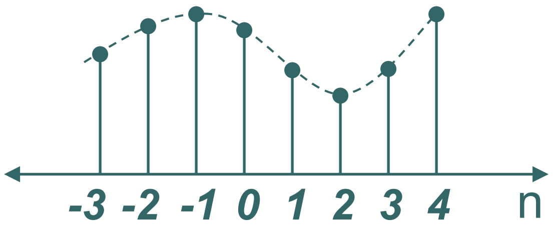
- **Format conversion**: convert signal from one format to another
  - Reduce sample rate to reduce the data rate (Down-sampling)
  - Increase the sampling rate (Up-sampling)
  - Image display when original image size differs from the display size

Introduction

- Most common sampling is periodic:

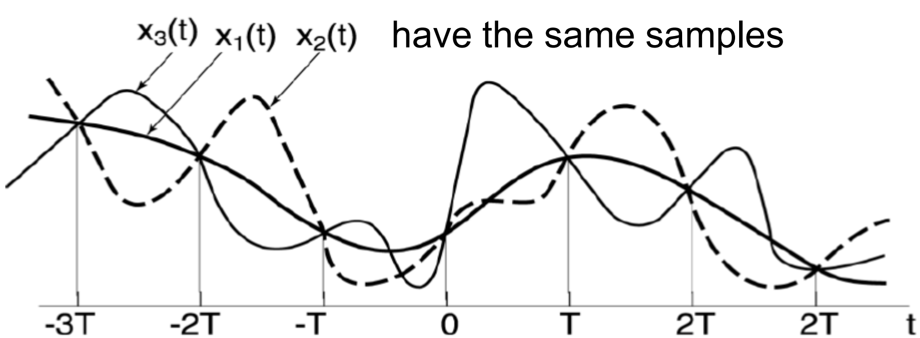
  Taking snap shots of x(t) every T second

$$x[n] = x_c(nT) \quad -\infty < n < \infty$$



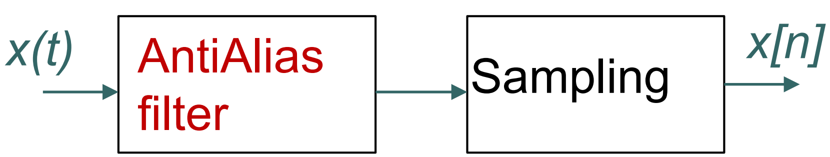
  - *T: sampling period* in *second*
  - $F_s = 1/T$: *sampling frequency* in *samples/s* or *Hz*
  - $\Omega_s = 2\pi F_s$: sampling rate in *radian/s*

- Sampling is, in general, **not reversible**: Given a sampled signal one could fit infinite continuous signals through the samples



Introduction

- Key Questions for Sampling:
  - How do we **determine** ***T*** ? ➔Look at the frequency range of the signal
  - Can we perfectly **reconstruct** the original CT signal x(t) from its samples x[n] ? ➔Nyquist sampling theorem
  - What if the sampling rate Ω_s is lower than the max frequency in *x[n]* ? ➔ Pre-filter (remove some high freq.)

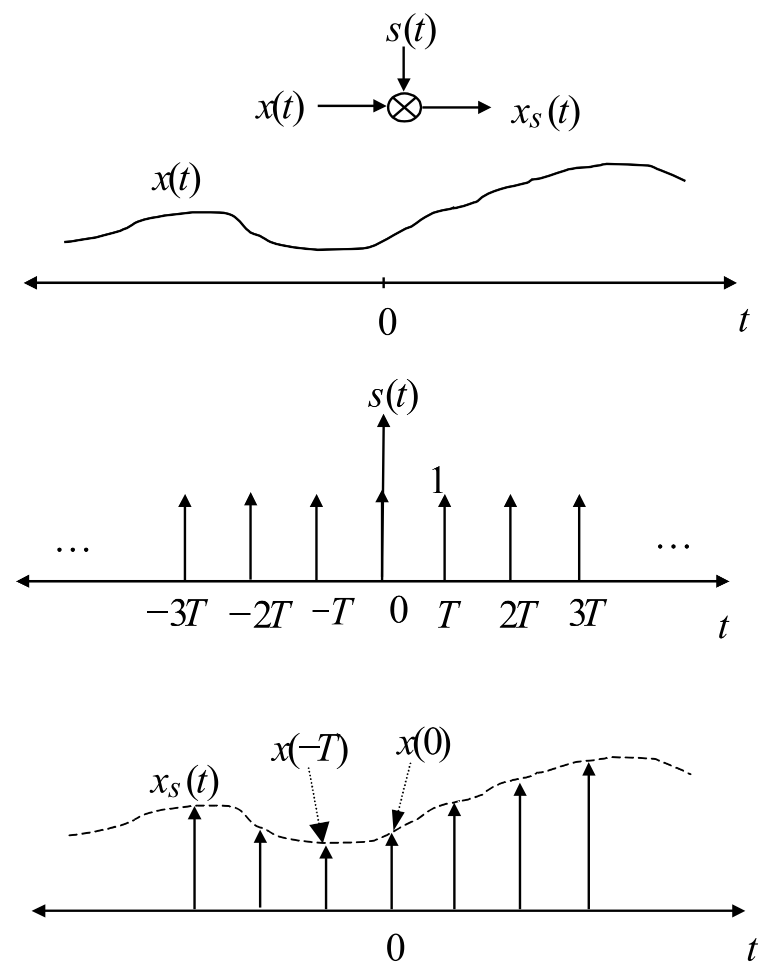


Outline

- Introduction
- Sampling: Representation of a CT Signal by Its Samples
- Reconstruction of a Signal from Its Samples: Interpolation
- Under-sampling: Aliasing
- DT Processing of CT Signals
- A/D and D/A Conversion
- Sampling of DT Signals
- Summary

Ideal Sampling

- <u>Impulse train</u>: an ideal system that samples x(t) at the given instant "*n*"
  - Mathematically convenient to represent in **<u>two stages</u>**



1. Multiply with s(t)= Impulse train modulator **t ➔ nT**

(the *n*th sample is associated with the impulse at *t=nT*)

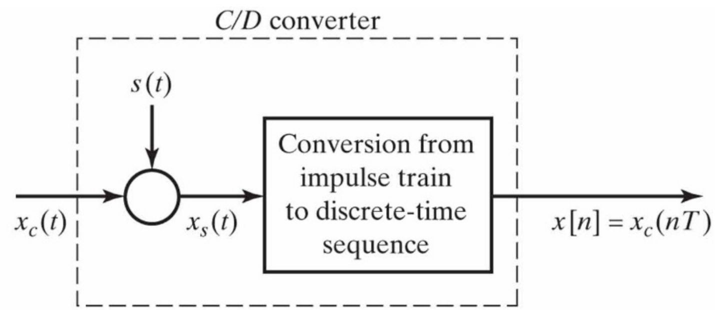
2. Conversion of impulse train to a sequence **nT ➔ n**

(scale the x-axis: divide *t* by *T* to get *n; in frequency domain: multiply by* T)

Impulse-Train Sampling:

Time domain

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

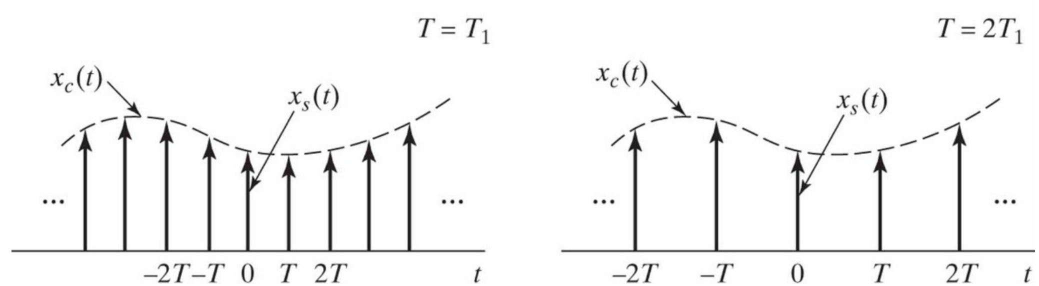


(a)

- $x_s(t) = x_c(t)\, s(t)$

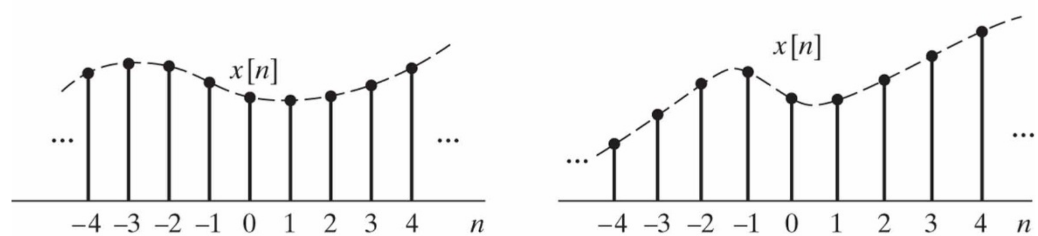
$$= x_c(t) \sum_{n=-\infty}^{\infty} \delta(t - nT)$$

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



(b) $x_s(t)$ for two sampling rates

- $x[n] = x_c(nT), \qquad -\infty < n < \infty$



(c)

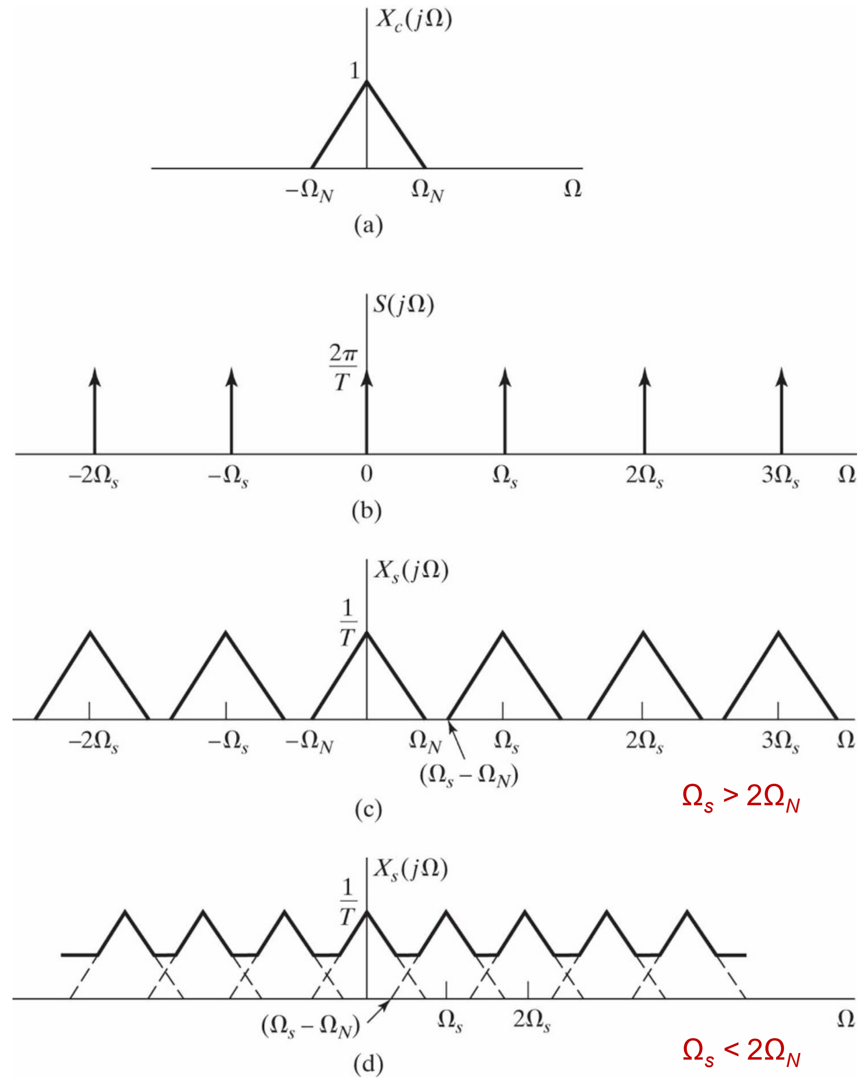
The output sequence for the two different sampling rates

Frequency analysis of Sampling

- $x_s(t) = x_c(t)s(t) = \sum_{n=-\infty}^{\infty} x_c(t)\delta(t - nT)$
- $s(t) \Leftrightarrow S(j\Omega) = \frac{2\pi}{T}\sum_{k=-\infty}^{\infty}\delta(\Omega - k\Omega_s)$
- $X_s(j\Omega) = \frac{1}{2\pi}X_c(j\Omega) * S(j\Omega)$
- $x_s(t) \Leftrightarrow X_s(\Omega) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X_c(\Theta)S(\Theta - \Omega)d\Theta$

$$= \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c(j(\Omega - k\Omega_s))$$

➔Convolution with pulse creates replicas at pulse location, i.e., replicas are periodic at $k\Omega_s$

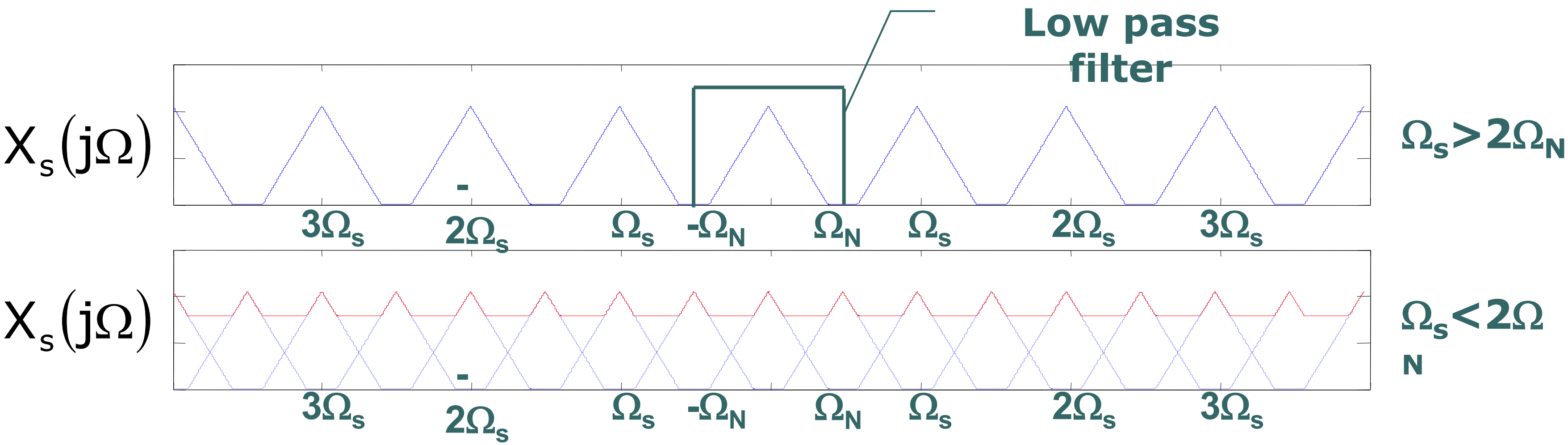


Outline

- Introduction
- Representation of a CT Signal by Its Samples: Sampling
- Reconstruction of a Signal from Its Samples
- Under-sampling: Aliasing
- DT Processing of CT Signals
- A/D and D/A Conversion
- Sampling of DT Signals
- Summary

The Sampling Theorem

- How to recover xc(t) **EXACTLY** from its samples?

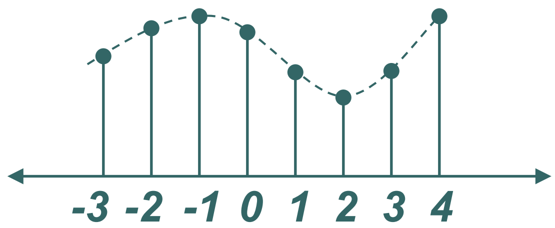


- Let $x_c(t)$ be a band-limited signal: $X_c(j\Omega) = 0 \quad \text{for } |\Omega| \geq \Omega_N$

- $x_c(t)$ is <u>uniquely determined by its samples</u> $x[n] = x_c(nT)$ if

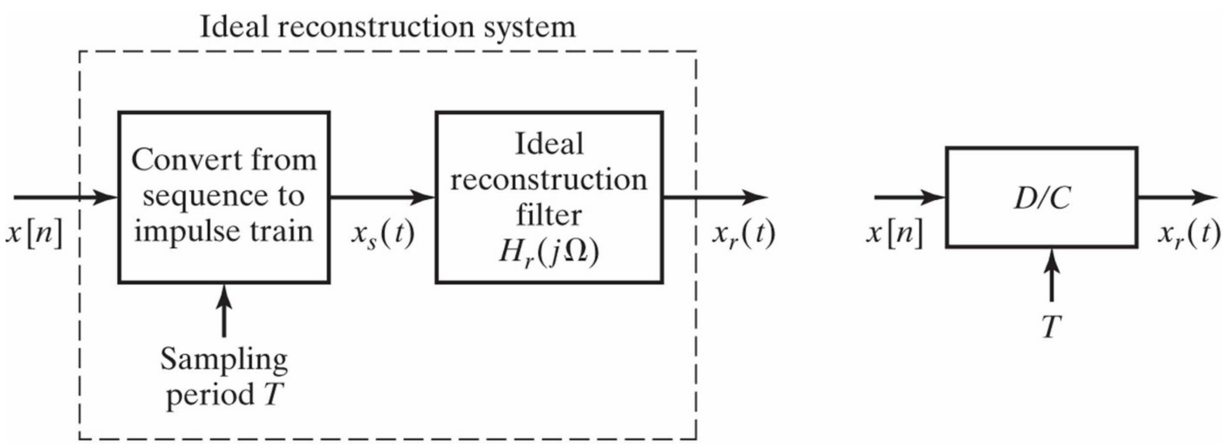
$$\Omega_s = \frac{2\pi}{T} = 2\pi F_s \geq 2\Omega_N$$

Reconstruction Methods



- Reconstruction is interpolation:
  - Connecting *x[n]* samples using interpolation kernels
- Ideal: Band-limited Interpolation

$$X_r(j\Omega) = H_r(j\Omega)X(e^{j\Omega T})$$

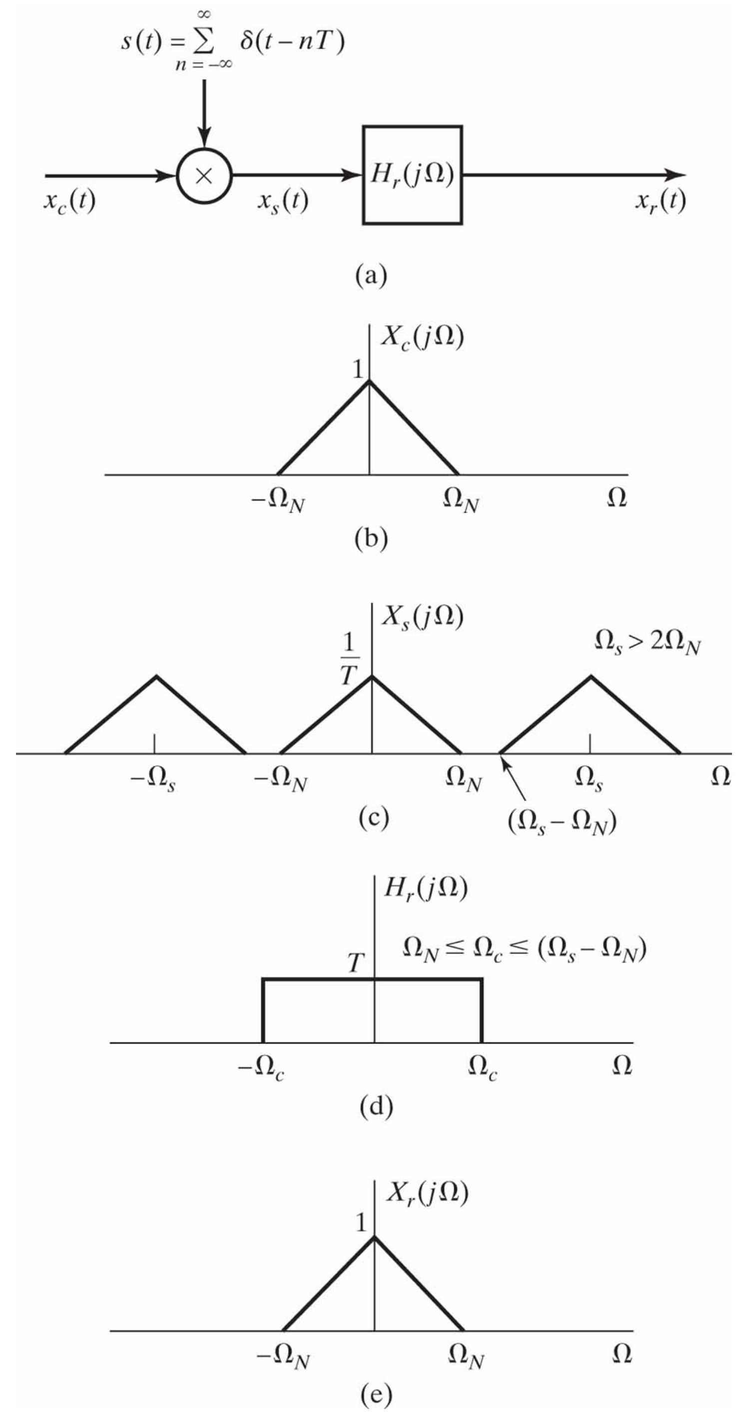


(a)

(b) Equivalent representation of (a) as an ideal D/C converter.

Band-limited Interpolation: Frequency domain

- If there is no overlap between shifted spectra (sampling theorem satisfied), a LPF can recover x(t) from x[n]



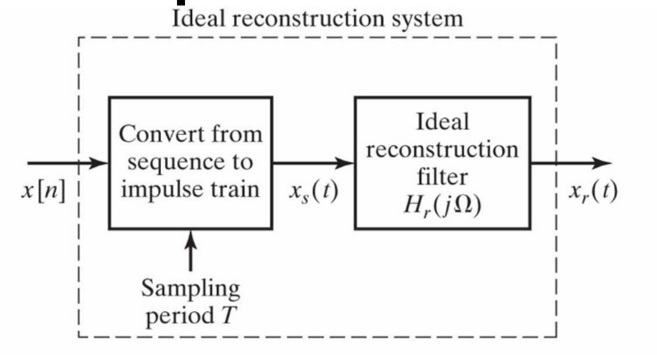
Sampling: Symbols

- Ω_N The *maximum* frequency of $x_c(t)$: the Nyquist Frequency
- $2\Omega_N$ The *minimum* sampling rate that must be exceeded : the Nyquist Rate

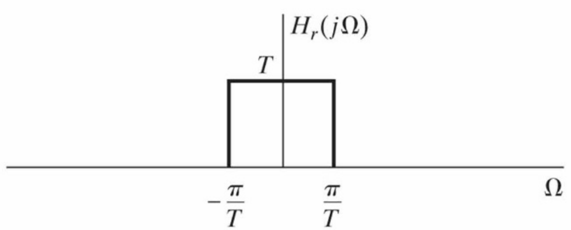
$$\Omega_s = \frac{2\pi}{T} = 2\pi F_s \geq 2\Omega_N$$

- T The Sampling Period
- $[-\frac{\Omega_S}{2}, \frac{\Omega_S}{2}]$ The Nyquist Interval
- $$\Omega_c = \frac{\Omega_s}{2} = \frac{\pi}{T} \Rightarrow \Omega_c > \Omega_N$$ The cutoff frequency

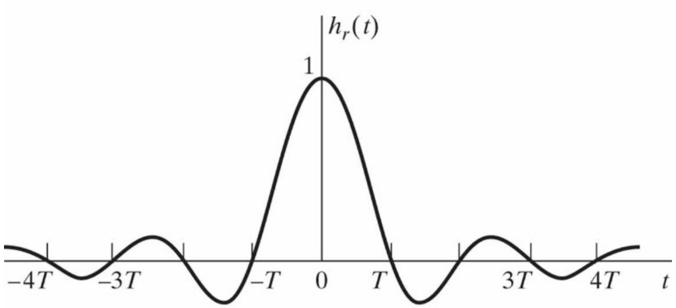
Band-limited Interpolation



(a)



(b)



(c)

$$x_s(t) = x[n]\, s(t); \qquad s(t) = \sum_{n=-\infty}^{+\infty} \delta(t - nT)$$

$$x_r(t) = x_s(t) * h(t) = \sum_{n=-\infty}^{+\infty} x(nT) h_r(t - nT)$$

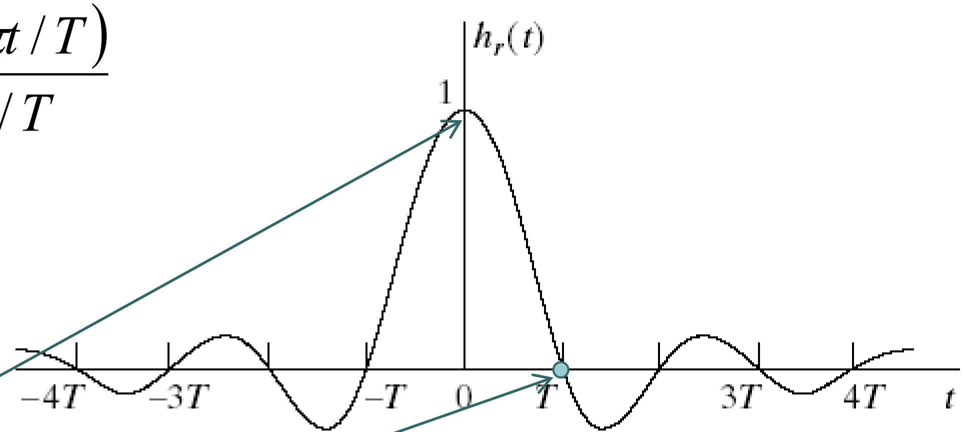
Ideal LPF with cut of frequency of $\Omega_c=\pi/T$ or $F_c=2/T$

$$h_r(t) = \frac{\Omega_c T \sin(\Omega_c t)}{\pi \Omega_c t} = \frac{\sin(\pi t / T)}{\pi t / T}$$

➔normalized sinc function

Normalized sinc Properties

$$h_r(t) = \frac{\sin(\pi t / T)}{\pi t / T}$$



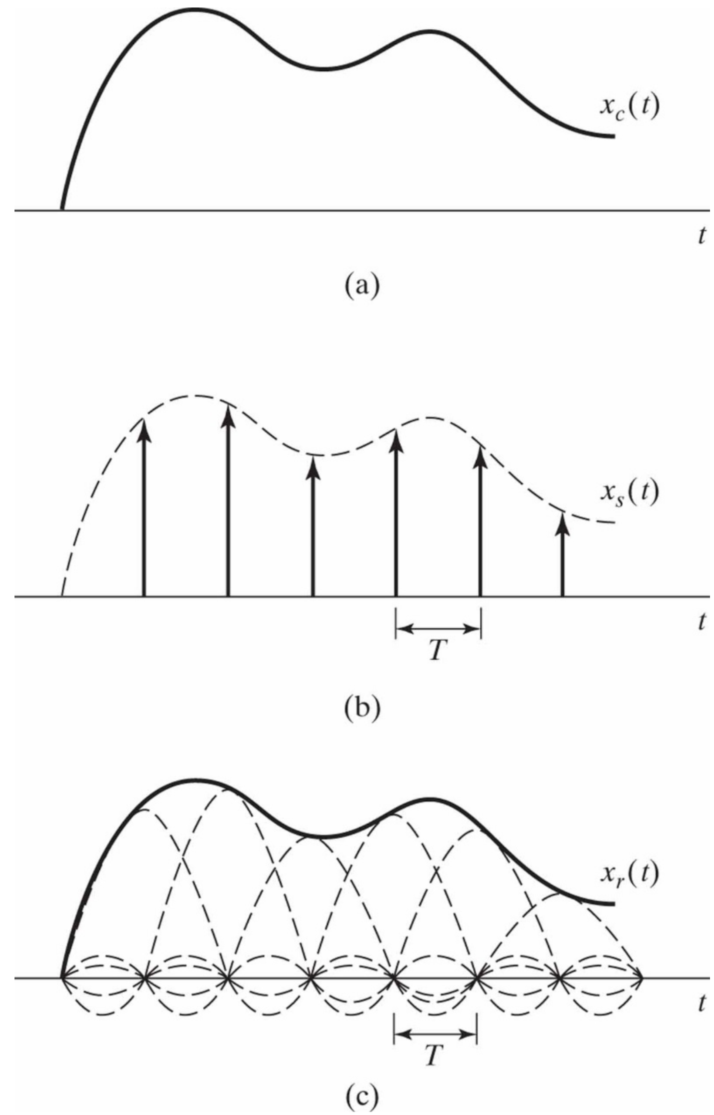
- $\int_{-\infty}^{\infty} \frac{\sin(\pi x)}{\pi x} dx = rect(0)$
- $\int_{-\infty}^{\infty} \left| \frac{\sin(\pi x)}{\pi x} \right| dx = +\infty$

- $sinc(0) = 1,$
- $sinc(k) = 0 \; for \; nonzero \; integer \; k$ **t/T integer**
- $\lim_{a \to 0} \frac{1}{a} sinc(x/a) = \delta(x)$ **T→0**

$\int_{-\infty}^{\infty} |h_r(t)| \, dx = +\infty$ $\int_{-\infty}^{\infty} h_r(t) dt = T = H_r(j0)$

Band-limited Interpolation: Time domain

- Given the samples x[n]
- We can reconstruct $x(t)$ by generating a periodic impulse train with amplitudes that are successive sample values



- This impulse train is then processed through an ideal LPF with gain T and *cutoff frequency* $\Omega_N < \Omega_c < (\Omega_s - \Omega_N)$
- Interpretation of convolution operation:
  - Replacing each pixel by a weighted sum of its neighbors

$$x_r(t) = x_s(t) * h_r(t) = \sum_{n=-\infty}^{+\infty} x(nT)h_r(t - nT)$$

- The resulting output signal $x_r(t)$ will exactly be equal $x_c(t)$

Sinc (ideal) reconstruction

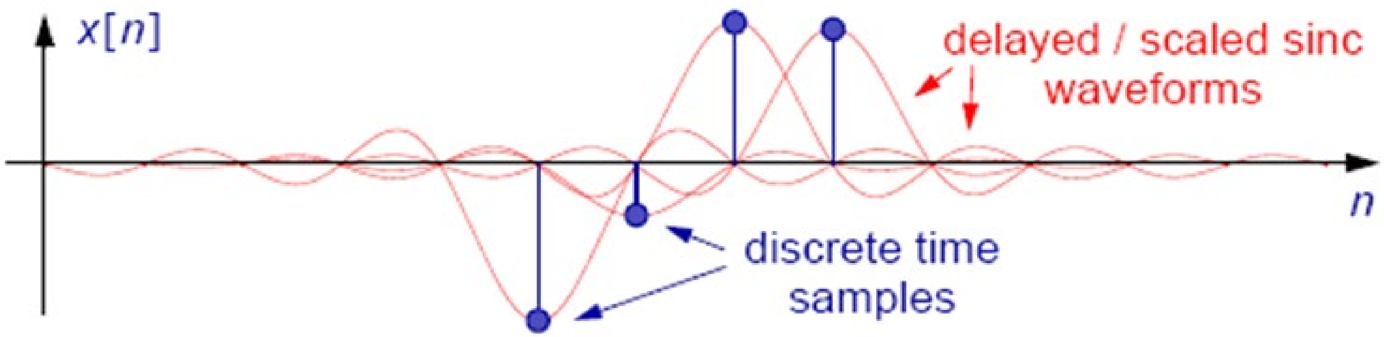
$$x_r(t) = \sum_{n=-\infty}^{\infty} x[n] \frac{\sin[\pi(t - nT)/T]}{\pi(t - nT)/T}$$

$$h_r(t) = \frac{\sin(\pi t / T)}{\pi t / T}$$

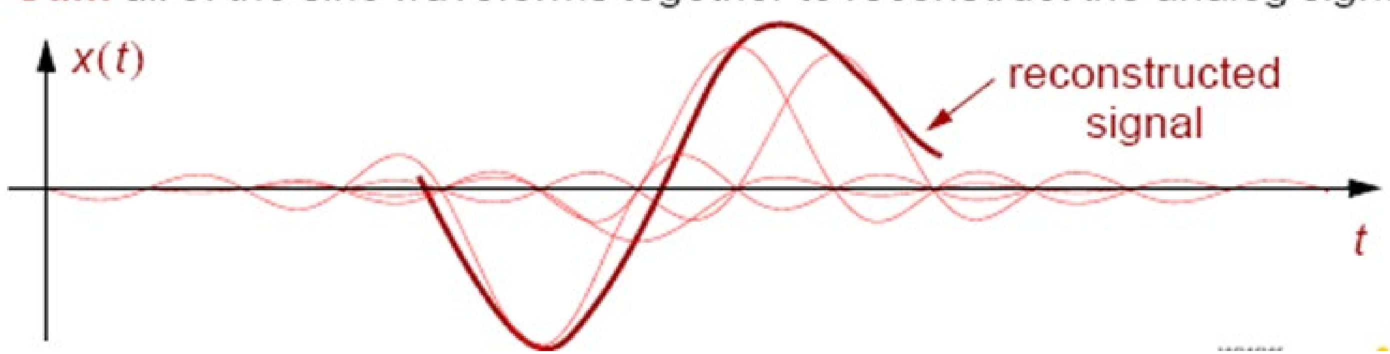
sinc is 1 at $t = 0$

sinc is 0 at $t = nT$

- Centre a sinc function on each discrete time sample
- Scale each sinc function by the value of that sample



- **Sum** all of the sinc waveforms together to reconstruct the analog signal

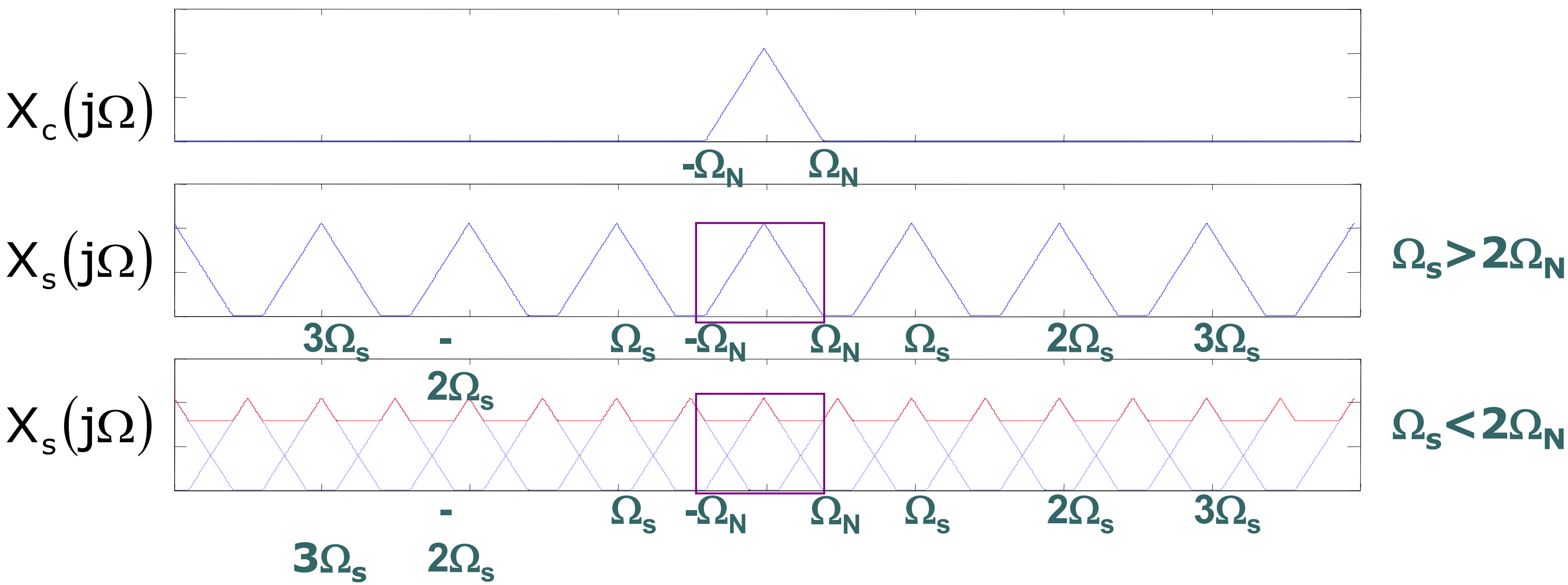


Outline

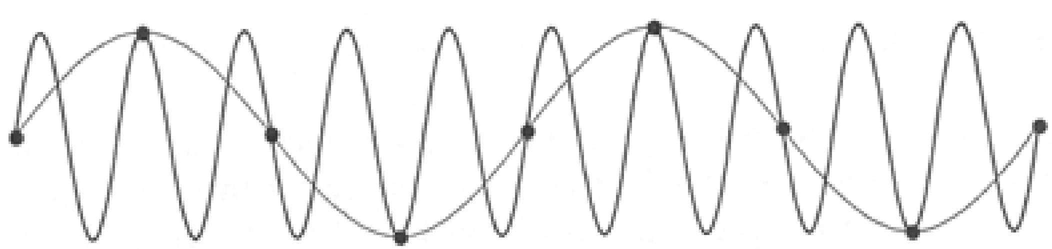
- Introduction
- Sampling
- Reconstruction
- Under-sampling: Aliasing
- DT Processing of CT Signals
- A/D and D/A Conversion
- Sampling of DT Signals
- Summary

The Effect of Undersampling: Aliasing

When $\Omega_s \leq 2\Omega_N$ **➔undersampling**



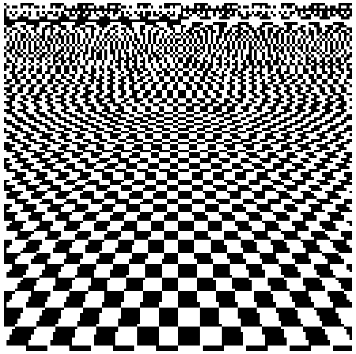
- **Aliasing: overlapping of replicas of Xs in frequency domain**
- **Aliasing: under-sampling in time domain (too few samples)**

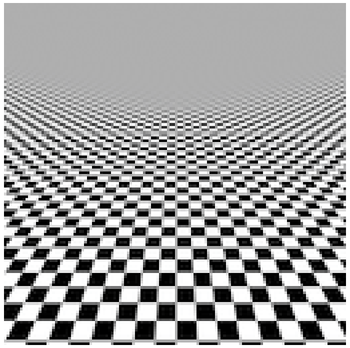


The Effect of Undersampling: Aliasing

- Aliasing is the presence of unwanted components in the reconstructed signal
  - These components were not present when the original signal was sampled
  - Some of the frequencies in the original signal may be lost in the reconstructed signal
- Aliasing occurs because signal frequencies can overlap if the sampling frequency is too low
- Frequencies "fold" around half the sampling frequency

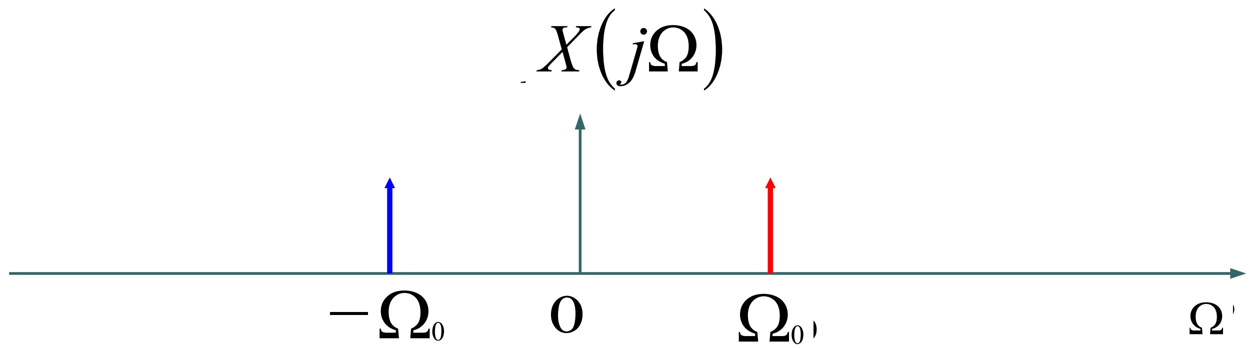
- Avoid aliasing; Use pre-filtering



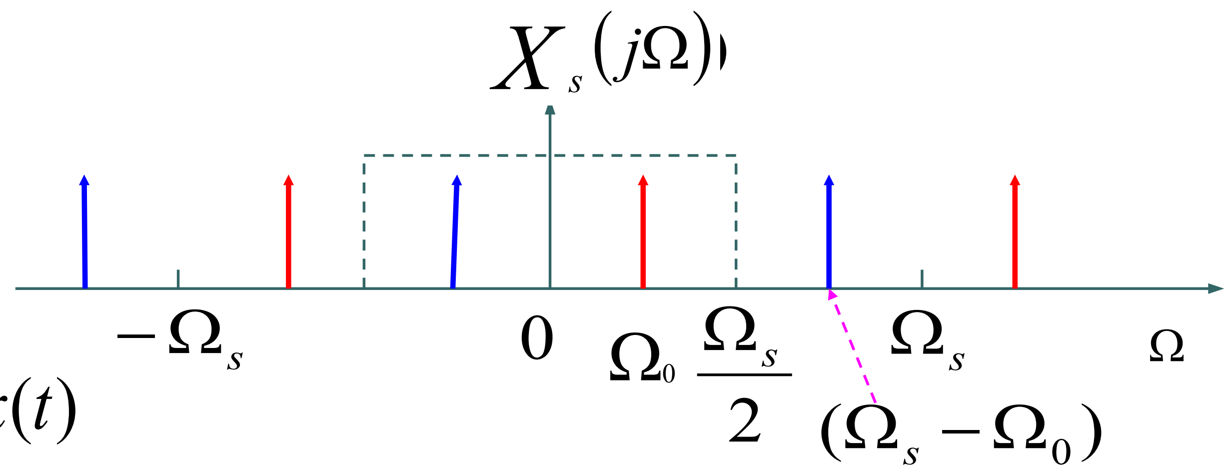


Aliasing: Example

$$x(t) = \cos(\Omega_0 t)$$



$$\Omega_s = 6\Omega_0$$



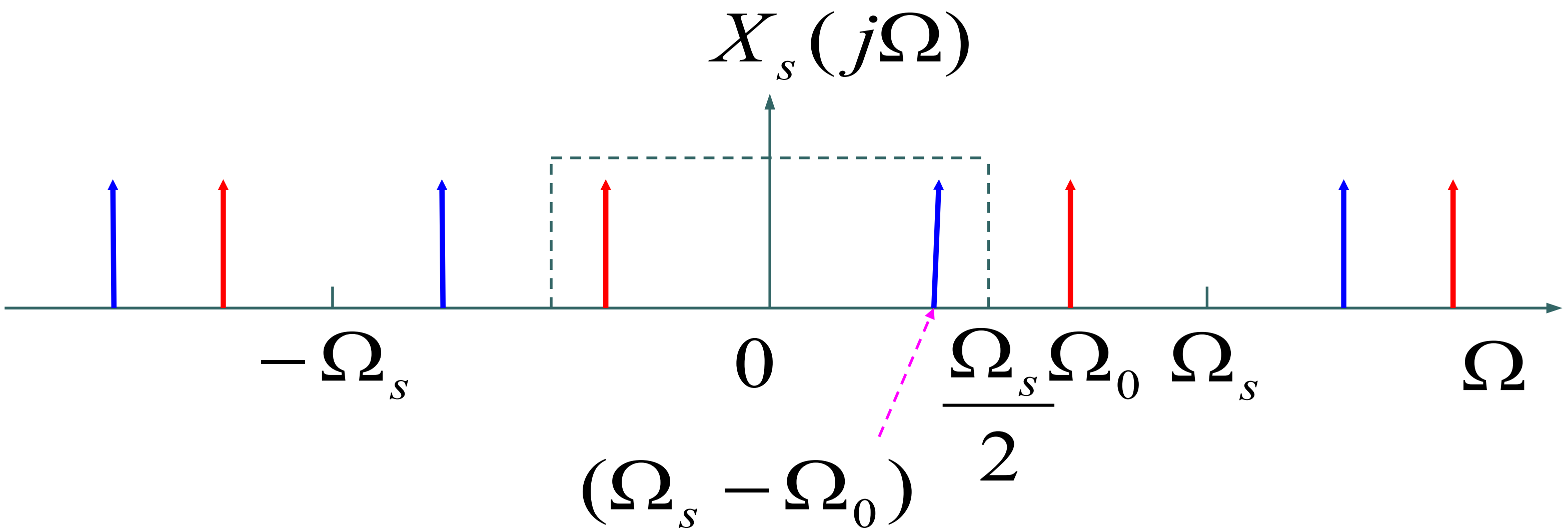
$$x_r(t) = \cos(\Omega_0 t) = x(t)$$

Aliasing: Example

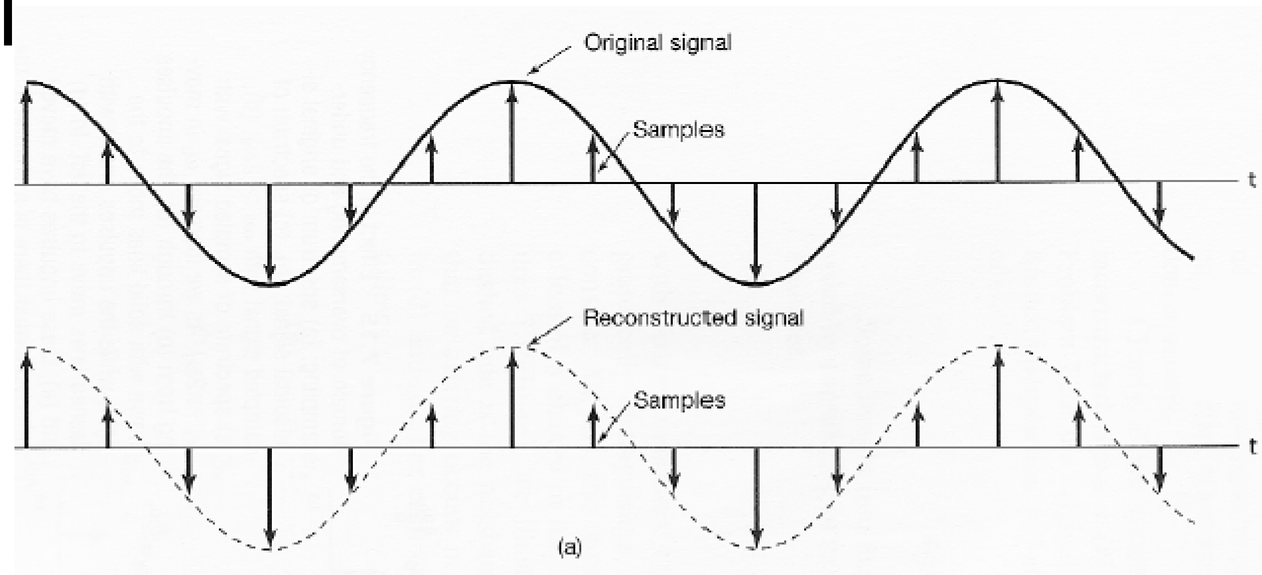
$$\Omega_s = \frac{6\Omega_0}{4}$$

➔Aliasing

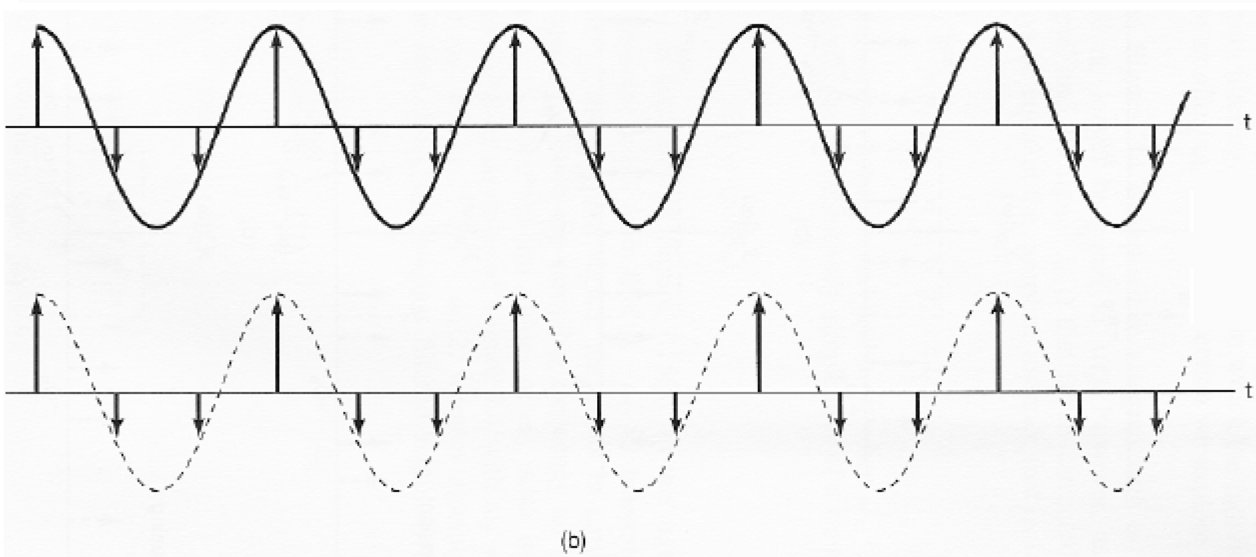
$$x_r(t) = \cos(\Omega_s - \Omega_0)t \neq x(t)$$



Aliasing: Example

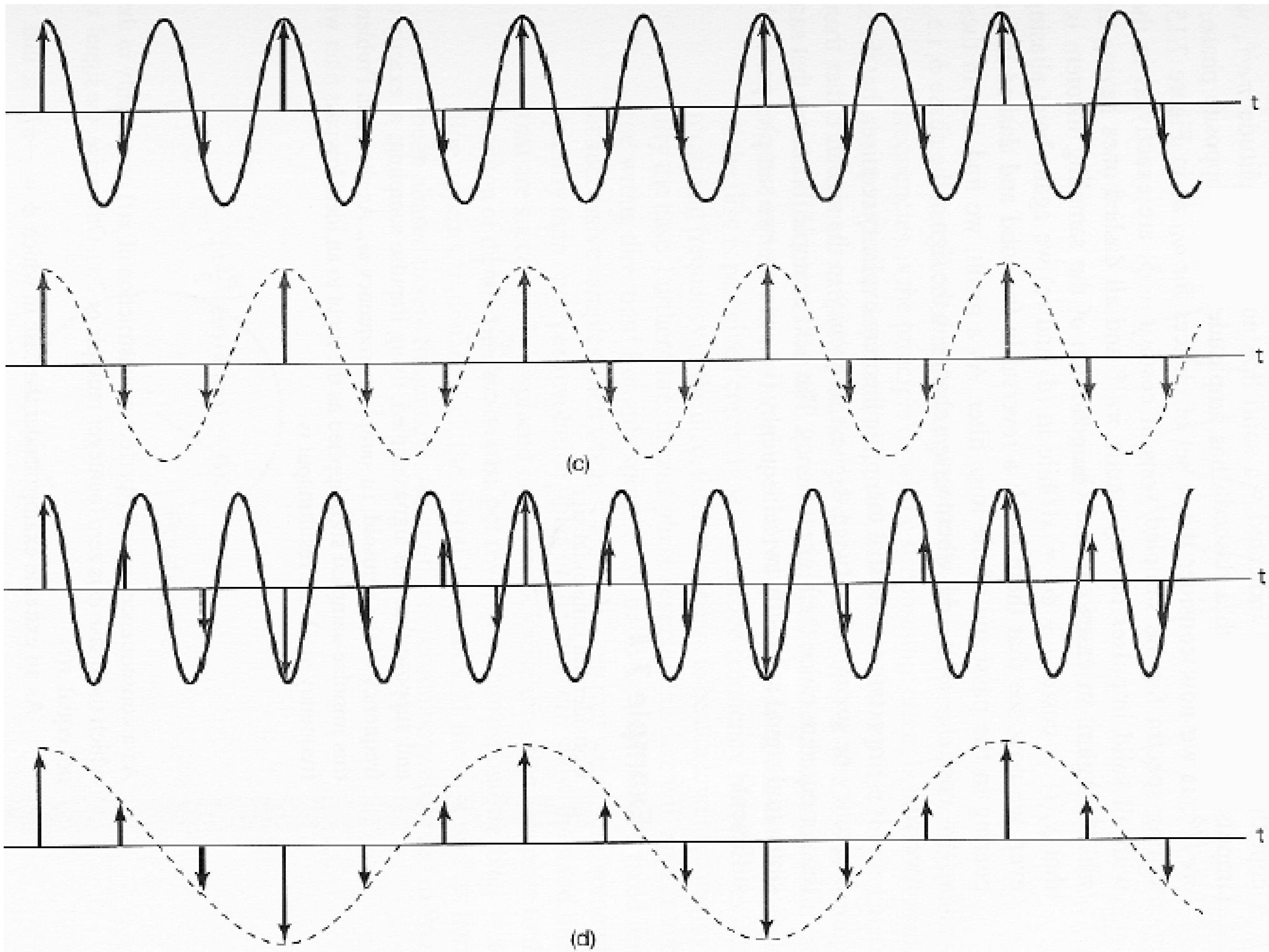


$$\Omega_0 = \frac{\Omega_s}{6}$$



$$\Omega_0 = \frac{2\Omega_s}{6}$$

Aliasing: Example



$$\Omega_0 = \frac{4\Omega_s}{6}$$

$$\Omega_0 = \frac{5\Omega_s}{6}$$

Outline

- Introduction
- Sampling
- Reconstruction
- The Effect of Under-sampling: Aliasing
- Sampling examples
- DT Processing of CT Signals
- A/D and D/A Conversion
- Sampling of DT Signals
- Summary

Example 1:
Sampling of sinusoidal Signals

$$\text{Given}: x_c(t) = \cos(4000\pi t)\ \&\ T = 1/6000$$

$$\text{Find x[n]}\ \&\ X(e^{j\omega})$$

$$1)\ x_c(t) = \cos(4000\pi t) \rightarrow \Omega_0 = 4000\pi$$

$$T = 1/6000 \rightarrow \Omega_s = 2\pi / T = 12000\pi \qquad \therefore \quad \text{no aliasing}$$

$$x[n] = x_c(nT) = \cos(4000\pi nT) = \cos((2\pi/3)n) = \cos(\omega_0 n)$$

$$2)\ x_c(t) \leftrightarrow X_c(j\Omega) = \pi\delta(\Omega - 4000\pi) + \pi\delta(\Omega + 4000\pi)$$

$$Note: \delta(at) = \delta(t) / |a|$$

$$X_s(j\Omega) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c(j(\Omega - k\Omega_s))$$

$$X(e^{j\omega}) = X_s(j\Omega)|_{\Omega=\omega/T} = X_s(j\omega/T) \ \ \text{with normalized frequency} \ \ \omega = \Omega T$$

Example 1: Sampling of sinusoidal Signals

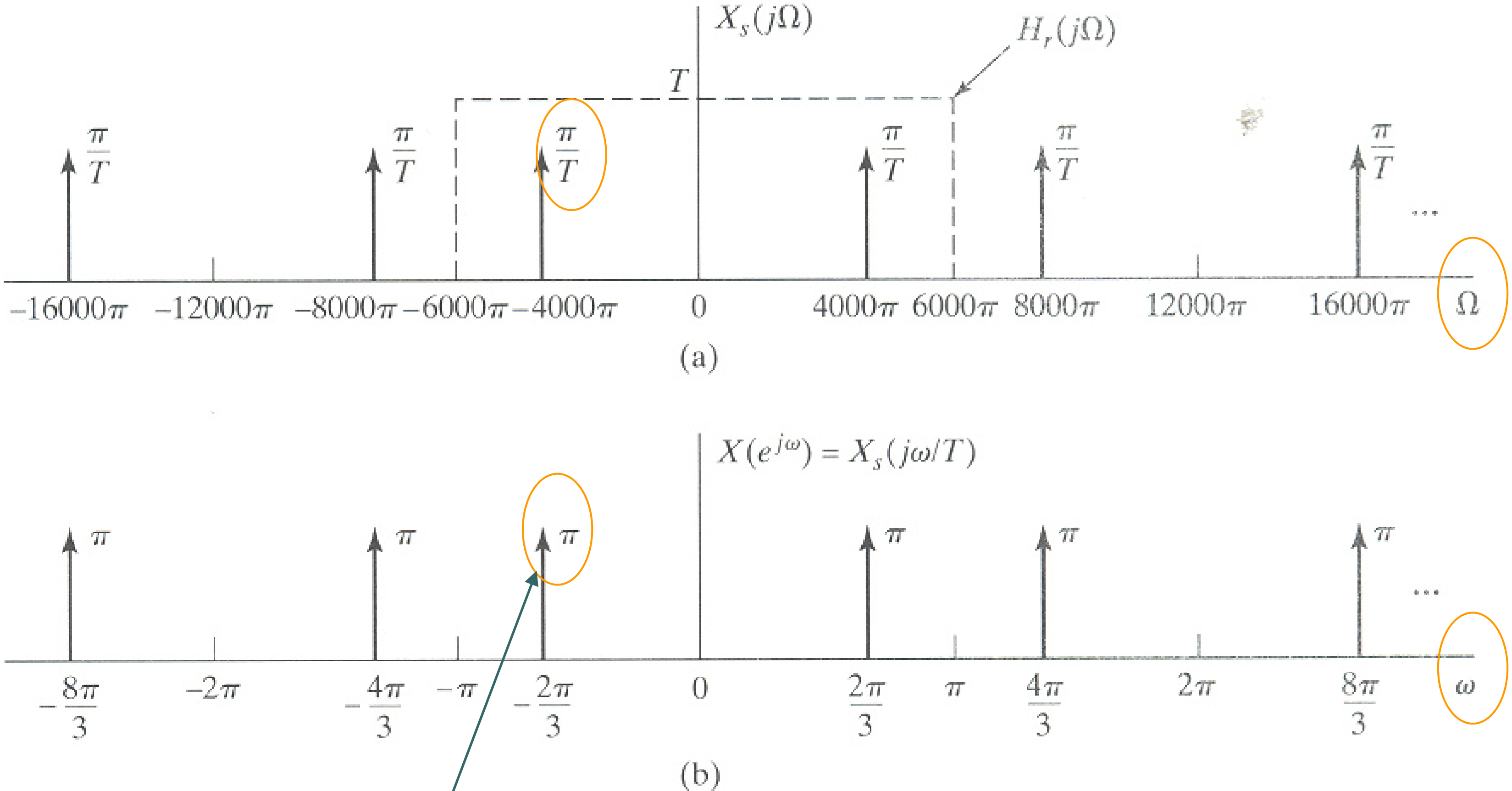


Figure 4.6 Continuous-time (a) and discrete-time (b) Fourier transforms for sampled cosine signal with frequency $\Omega_0 = 4000\pi$ and sampling period $T = 1/6000$.

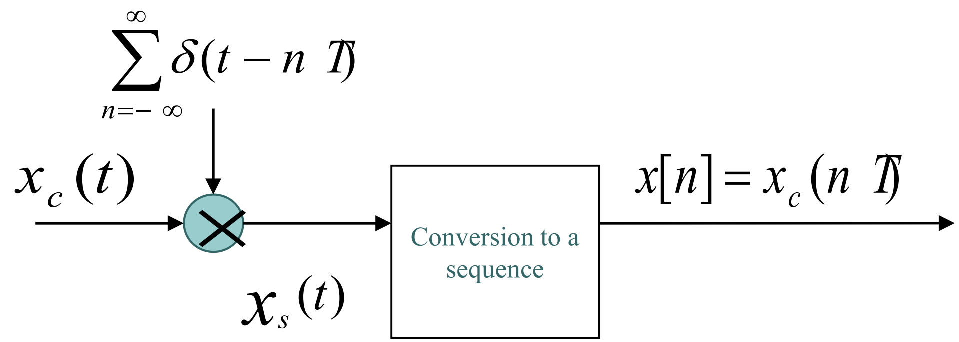
Since $\delta(at) = \delta(t)/|a| ==> \delta(\omega/T) = T\delta(\omega)$

How about

$x_c(t) = \cos(16000\pi t)$

Example 2:Sampling system

For the following system



find the FT of the output signal x[n] if

$$X_c(j\Omega) = \begin{cases} 1 + \dfrac{\Omega}{\Omega_M}, & -\Omega_M < \Omega < 0 \\ 1 - \dfrac{\Omega}{\Omega_M}, & 0 < \Omega < \Omega_M \\ 0, & |\Omega| > \Omega_M \end{cases}$$

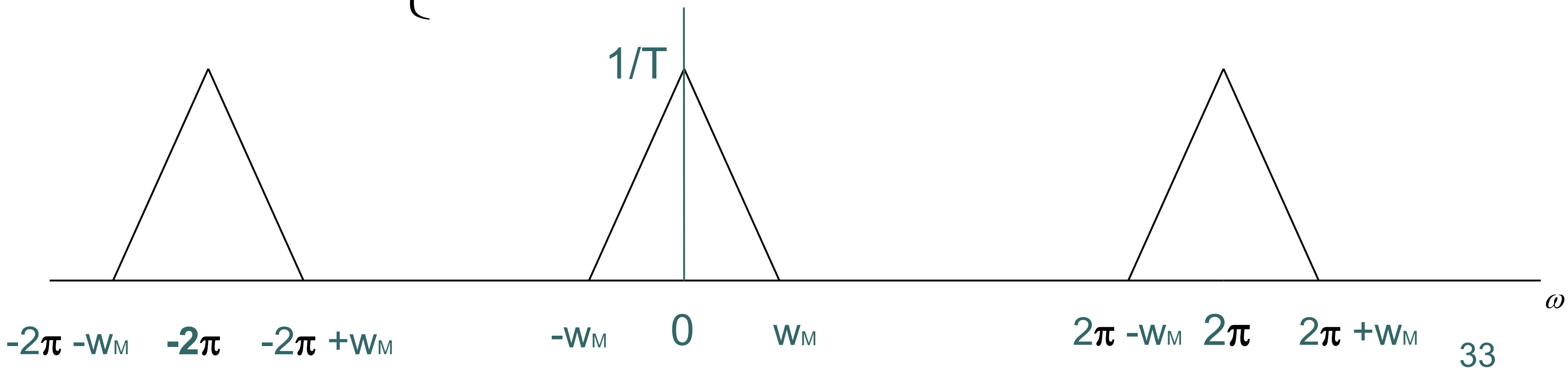
Suppose $\Omega_s > 2\Omega_M$

Example 2:Sampling system

- $$X_s(j\Omega) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c(j(\Omega - k\Omega_s)), \qquad \Omega_s = \frac{2\pi}{T}$$

$$X(e^{jw}) = X_s(\frac{j\omega}{T}) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c(j(\frac{\omega - 2\pi k}{T}))$$

$$X\ (e^{jw}) = \begin{cases} \frac{1}{T}(1+\frac{\omega}{\omega_M}), & -\omega_M < \omega < 0 \\ \frac{1}{T}(1-\frac{\omega}{\omega_M}), & 0 < \omega < \omega_M \\ 0, & |\omega| > \omega_M \end{cases}$$



Example 3: audio sampling

A signal at frequency 50Hz is sampled with F_s=80 Hz.
1- What frequency will be recovered ?
2- Repeat when it is sampled at 120Hz.

Part 1: investigation

$$x(t) = e^{j2\pi F_0 t} = e^{j2\pi 50t}$$

- Data collection:
  - With F0 = 50Hz and sampling with F_s=80 Hz, the signal is undersampled (the sampling theorem is not statisfied)
  - The Nyquist interval is [-40Hz, 40Hz]
  - The samples do not only represent the frequency F = 50Hz but all frequencies

  F± k*F_s=50±m80 , k=0, 1, 2..., i.e. the frequencies
  F_0=50, 50±80, 50±160, 50±240...= 50,130 ,-30,210,-110,290,-190

- Analysis:

$$x_r(t) = e^{j2\pi(-30)t}$$

  ➔Only the frequency -30Hz lies within the Nyquist interval
  ➔Then the recovered signal will be -30Hz (30Hz and phase reversal)
  ➔This signal is the alias of the original signal at 50Hz
  - Notice that 30Hz is just the difference 80Hz – 50Hz

"Source: http://cnx.org/content/m28684/latest/ "

Example 3: audio sampling

Part 2:

- Data collection:
  - Now, the sampling frequency is 120Hz, the sampling theorem is statisfied
- Analysis:
  - Then the original frequency of 50Hz will be recovered
  - But none of other frequencies

$$x_r(t) = e^{j2\pi(50)t}$$

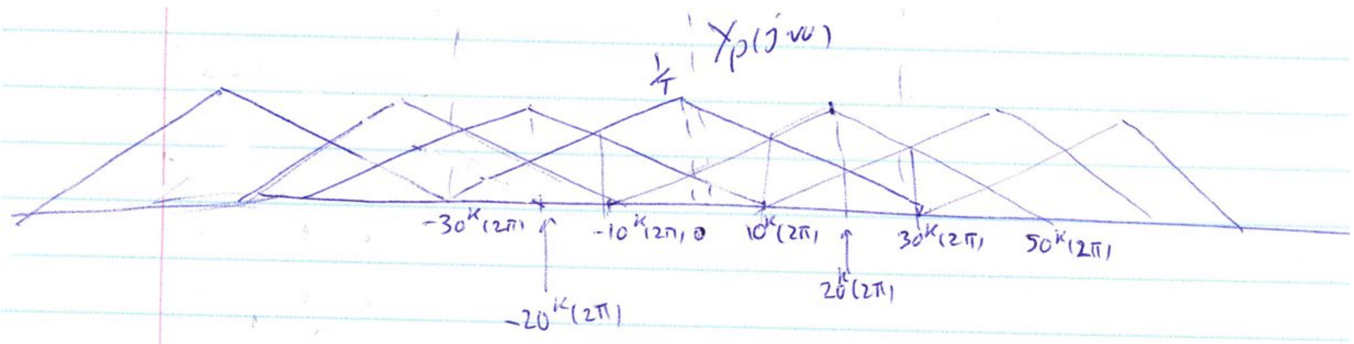
$$F_0=50\pm k*120=50, 170, -70, 290, -190...$$

lie in the Nyquist interval [-60Hz, 60Hz], except the original frequency of 50Hz as already known

Example 4: audio sampling

- A system uses the sampling frequency Fs=20 kHz
  to process audio signal that is frequency-limited at 10 kHz, **but** the anti-aliasing filter still allows frequencies up to 30 khz pass through even at small amplitudes. **What signal will we get back from the samples?**
- For sampling rate Fs=20 kHz, the Nyquist interval is [-10kHz, 10kHz]
  ➔ the audio frequency 0 –10 kHz will be recovered as is
  ➔ The audio frequency from 10 – 20 kHz will be aliased into the frequency range -10 – 0 kHz
  ➔ The audio frequency from 20 – 30 kHz will be aliased into the frequency range 0 – 10 kHz

➔The resulting audio will be **<u>distorted</u>** due to the superposition of the 3 frequency bands caused by the too high Fc of the anti-aliasing filter compared to Fs



Example 5:

A signal $x_p(t)$ is obtained through impulse train

sampling with sampling frequency ω_S: $x(t) = \cos(\frac{\omega_S}{2}t + \phi)$;

$$x_s(t) = \sum_{n=-\infty}^{+\infty} x(nT)\delta(t - nT), \quad T = \frac{2\pi}{\omega_s}$$

(a) Find $g(t)$ such that $x(t) = \cos(\phi)\cos(\frac{\omega_S}{2}t) + g(t)$

Using Trigonometric identities,

$$\cos(\frac{\omega_S}{2}t + \phi) = \cos(\frac{\omega_S}{2}t)\cos(\phi) - \sin(\frac{\omega_S}{2}t)\sin(\phi)$$

$$\Rightarrow g(t) = -\sin(\frac{\omega_S}{2}t)\sin(\phi) \qquad (1)$$

Example 5:

(b) Show that $g(nT) = 0$, for n=0, $\pm 1, \pm 2, \ldots$

By replacing ω_s with $\frac{2\pi}{T}$, and t by nT in the equation (1), we get

$g(nT) = -\sin(\frac{2\pi}{2T} nT)\sin(\phi) = -\sin(n\pi)\sin(\phi)$, the right hand side of the equation is equal to zero for n=0, $\pm 1, \pm 2, \ldots$

(c) Using the results of the previous two parts, show that if $x_p(t)$ is applied as the input to an ideal lowpass filter with cutoff frequence $\frac{\omega_s}{2}$, the resulting output is

$y(t) = \cos(\phi)\cos(\frac{\omega_s}{2} t)$.

Example 5:

From parts (a) and (b), we get

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

$$= \sum_{n=-\infty}^{+\infty} \delta(t-nT)\left\{\cos(\frac{\omega_s}{2}nT)\cos(\phi)+g(NT)\right\}$$

$$= \sum_{n=-\infty}^{+\infty} \delta(t-nT)\cos(\frac{\omega_s}{2}nT)\cos(\phi).$$

When the system is passed through a lowpass filter, we are performing a band-limited interpolation, the result is the signal $y(t)=\cos(\frac{\omega_s}{2}t)\cos(\phi)$.

Example 6:

- Consider Sinusoidal signal

$$x(t) = \cos(\frac{\omega_S}{2}t + \phi)$$

- Suppose that this signal is sampled, using impulse sampling, at exactly twice the frequency of the sinusoid, i.e., at sampling frequency ω_S

- if this impulse-sampled signal is applied as the input to an ideal lowpass filter with cut frequency $\omega_S/2$., the resulting output is:

$$x_r(t) = \cos(\frac{\omega_S}{2}t)\cos(\phi)$$

- As a consequence, we see that perfect reconstruction of $x(t)$ occurs only in the case in which the phase Φ is zero (or an integer multiple of 2π. Otherwise, the signal $x_r(t)$ does not equal $x(t)$.

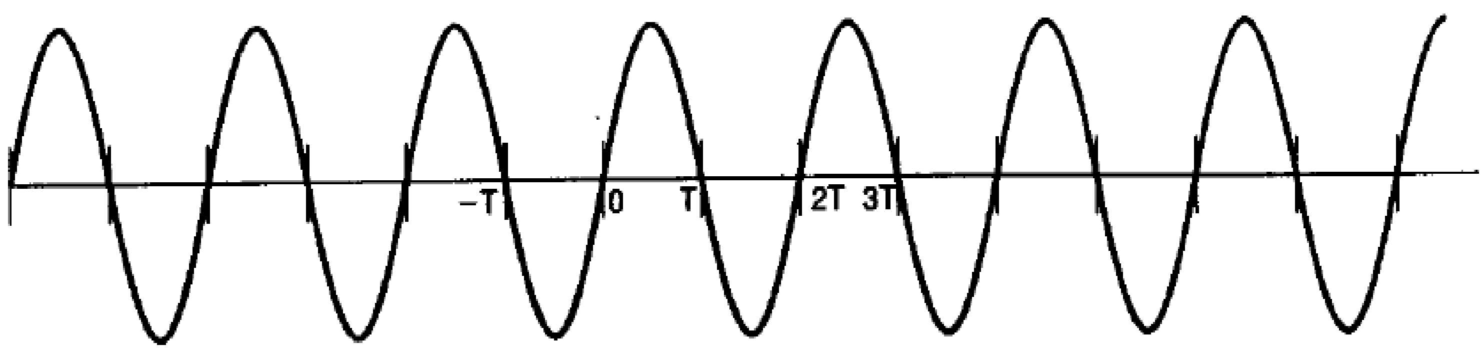
- As an extreme example, consider the case in which $\Phi = -\pi/2$, so that

$$x(t) = \sin(\frac{\omega_S}{2}t)$$

Example 6 :

- The values of the signal at integer multiples of the sampling period $2\pi / \omega_S$ are zero.

- Consequently. sampling at this rate produces a signal that is identically zero, and when this zero input is applied to the ideal lowpass filter, the resulting output $x_r(t)$ is also identically zero.



Example 7

- Consider the following sinusoidal signal with the fundamental frequency F = 4kHz:

  $g(t) = 5\cos(2\pi Ft) = 5\cos(8000\pi t)$.

- The sinusoidal signal is sampled at a sampling rate Fs = 6000 samples/second and reconstructed with an ideas low-pass filter (LPF) with the following transfer function:

  $H_1(jW) = 1/6000 : |W| <= 6000\pi$

  $0 : \text{otherwise}$

i) a) Determine the reconstructed signal y(t). Give details of derivations of Gs(jW).

b) Is the reconstruction perfect? If yes, justify and if no, suggest how can it be achieved. Give details.

(ii) Repeat (i) for a sampling rate f_s of 12 000 samples/s and an ideal LPF with the following transfer function:

$$H_2(\omega) = \begin{cases} 1/12\,000 & |\omega| \leq 12\,000\pi \\ 0 & \text{elsewhere.} \end{cases}$$

Solution

(i) The CTFT $G(\omega)$ of the sinusoidal signal $g(t)$ is given by

$$G(\omega) = 5\pi[\delta(\omega - 8000\pi) + \delta(\omega + 8000\pi)].$$

Using Eq. (9.4), the CTFT $G_s(\omega)$ of the sampled signal with a sampling rate $\omega_s = 2\pi(6000)$ radians/s ($T_s = 1/6000$ s) is expressed as follows:

$$G_s(\omega) = 6000 \sum_{m=-\infty}^{\infty} G(\omega - 2\pi m(6000)) = 6000 \sum_{m=-\infty}^{\infty} G(\omega - 12\,000m\pi).$$

Substituting the value of $G(\omega)$ in the above expression yields

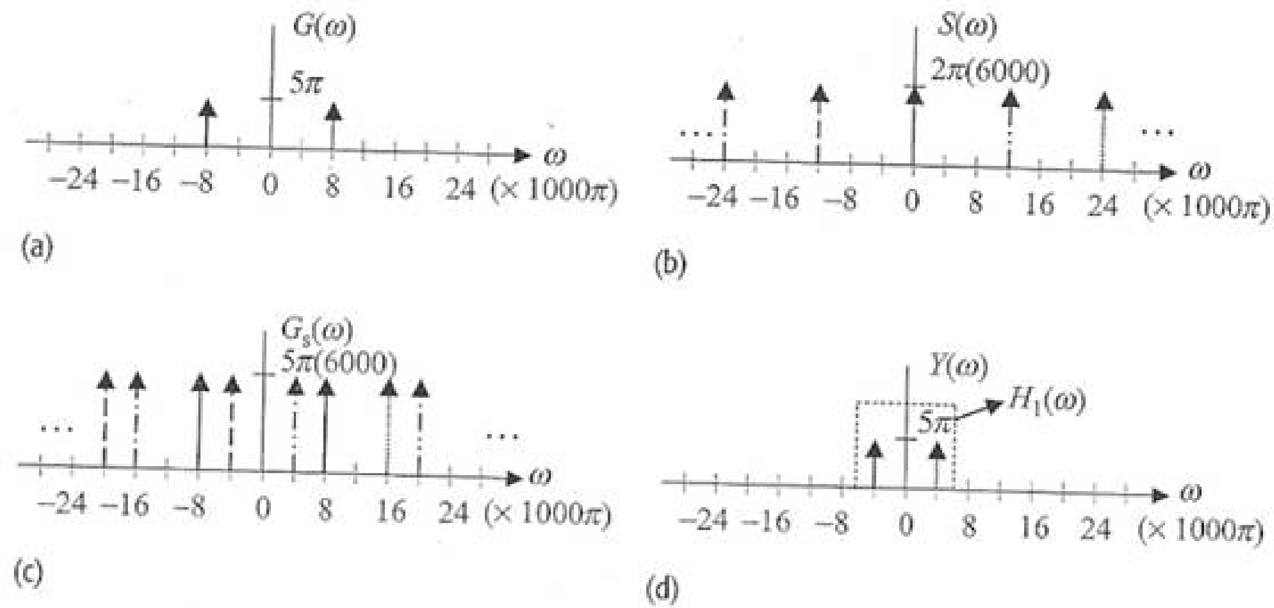
$$\begin{aligned}
G_s(\omega) &= 6000 \sum_{m=-\infty}^{\infty} 5\pi[\delta(\omega - 8000\pi - 12\,000\,m\pi) \\
&\qquad + \delta(\omega + 8000\pi - 12\,000\,m\pi)] \\
&= 6000(5\pi)\Bigg[\cdots + \underbrace{\delta(\omega + 16\,000\pi) + \delta(\omega + 32\,000\pi)}_{m=-2} \\
&\quad + \underbrace{\delta(\omega + 4000\pi) + \delta(\omega + 20\,000\pi)}_{m=-1} \\
&\quad + \underbrace{\delta(\omega - 8000\pi) + \delta(\omega + 8000\pi)}_{m=0} + \underbrace{\delta(\omega - 20\,000\pi) + \delta(\omega - 4000\pi)}_{m=1} \\
&\quad + \underbrace{\delta(\omega - 32\,000\pi) + \delta(\omega - 16\,000\pi)}_{m=2} + \cdots\Bigg].
\end{aligned}$$

When the sampled signal is passed through the ideal LPF with transfer function $H_1(\omega)$, all frequency components $|\omega| > 6000\pi$ radians/s) are eliminated from the output. The CTFT $Y(\omega)$ of the output $y(t)$ of the LPF is given by

$$Y(\omega) = H_1(\omega)G_s(\omega) = \frac{1}{6000} \cdot 6000(5\pi)[\delta(\omega + 4000\pi) + \delta(\omega - 4000\pi)].$$

Calculating the inverse CTFT, the reconstructed signal is given by $y(t) = 5\cos(4000\pi t)$.

Fig. 9.6. Sampling and reconstruction of a sinusoidal signal $g(t) = 5\cos(8000\pi t)$ at a sampling rate of 6000 samples/s. CTFTs of: (a) the sinusoidal signal $g(t)$; (b) the impulse train $s(t)$; (c) the sampled signal $g_s(t)$; and (d) the signal reconstructed with an ideal LPF $H_1(\omega)$ with a cut-off frequency of 6000π radians/s.



I

The graphical representation of the sampling and reconstruction of the sinusoidal signal in the frequency domain is illustrated in Fig. 9.6. The CTFTs of the sinusoidal signal $g(t)$ and the impulse train $s(t)$ are plotted, respectively, in Fig. 9.6(a) and Fig. 9.6(b). Since the CTFT $S(\omega)$ of $s(t)$ consists of several impulses, the CTFT $G_s(\omega)$ of the sampled signal $g_s(t)$ is obtained by convolving the CTFT $G(\omega)$ of the sinusoidal signal $g(t)$ separately with each impulse in $G_s(\omega)$ and then applying the principle of superposition. To emphasize the results of individual convolutions, a different pattern is used in Fig. 9.6(b) for each impulse in $S(\omega)$. For example, the impulse $\delta(\omega)$ located at origin in $S(\omega)$ is shown in Fig. 9.6(b) by a solid line. Convolving $G(\omega)$ with $\delta(\omega)$ results in two impulses located at $\omega = \pm 8000\pi$, which are shown in Fig. 9.6(c) by solid lines. Similarly for the other impulses in $S(\omega)$.

The output $y(t)$ is obtained by applying $G_s(\omega)$ to the input of an ideal LPF with a cut-off frequency of 6000π radians/s. Clearly, only the two impulses at $\omega = \pm 4000\pi$, corresponding to the sinusoidal signal $\cos(4000\pi t)$, lie within the pass band of the lowpass filter. The remaining impulses are eliminated from the output. This results in an output, $y(t) = \cos(4000\pi t)$, which is different from the original signal.

(ii) The CTFT $G_s(\omega)$ of the sampled signal with $\omega_s = 2\pi(12\,000)$ radians/s ($T_s = 1/12\,000$ s) is given by

$$G_s(\omega) = 12\,000 \sum_{m=-\infty}^{\infty} G(\omega - 2\pi m(12\,000))$$

$$= 12\,000 \sum_{m=-\infty}^{\infty} G(\omega - 24\,000 m\pi).$$

Substituting the value of the CTFT $G(\omega) = 5\pi[\delta(\omega - 8000\pi) + \delta(\omega + 8000\pi)]$ in the above equation, we obtain

$$\begin{aligned}
G_s(\omega) &= 12\,000 \sum_{m=-\infty}^{\infty} 5\pi[\delta(\omega - 8000\pi - 24\,000 m\pi) \\
&\quad + \delta(\omega + 8000\pi - 24\,000 m\pi)] \\
&= 12\,000(5\pi)\Bigg[\cdots + \underbrace{\delta(\omega + 40\,000\pi) + \delta(\omega + 56\,000\pi)}_{m=-2} \\
&\quad + \underbrace{\delta(\omega + 16\,000\pi) + \delta(\omega + 32\,000\pi)}_{m=-1} \\
&\quad + \underbrace{\delta(\omega - 8000\pi) + \delta(\omega + 8000\pi)}_{m=0} + \underbrace{\delta(\omega - 32\,000\pi) + \delta(\omega - 16\,000\pi)}_{m=1} \\
&\quad + \underbrace{\delta(\omega - 56\,000\pi) + \delta(\omega - 40\,000\pi)}_{m=2} + \cdots\Bigg].
\end{aligned}$$

To reconstruct the original sinusoidal signal, the sampled signal is passed through an ideal LPF $H_2(\omega)$. The frequency components outside the pass-band range $|\omega| \leq 12\,000\pi$ radians/s are eliminated from the ouput. The CTFT $Y(\omega)$ of the output $y(t)$ of the LPF is therefore given by

$$Y(\omega) = 5\pi[\delta(\omega + 8000\pi) + \delta(\omega - 8000\pi)],$$

which results in the reconstructed signal

$$y(t) = 5\cos(8000\pi t).$$

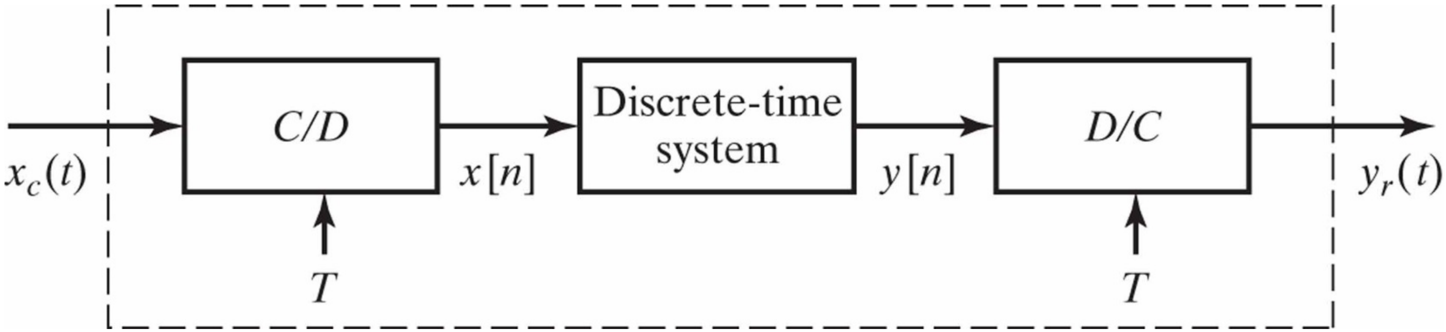
The graphical interpretation of the aforementioned sampling and reconstruction process is illustrated in Fig. 9.7.

As the signal g(t) is a sinusoidal signal with frequency 4 kHz, the Nyquist sampling rate is 8 kHz. In part (i), the sampling rate (6 kHz) is lower than the Nyquist rate, and consequently the reconstructed signal is different from the original signal due to the aliasing effect. In part (ii), the sampling rate is higher than the Nyquist rate, and as a result the original sinusoidal signal is accurately reconstructed.

Outline

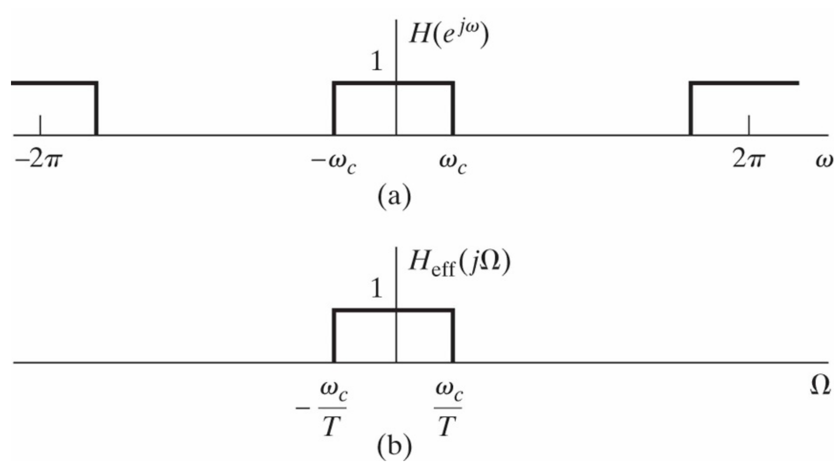
- Introduction
- Representation
- Reconstruction
- The Effect of Under-sampling: Aliasing
- Examples
- DT Processing of CT Signals
  - DT processing: Effective CT Frequency Response
  - DT from CT: Impulse invariance
- CT processing: Effective DT Frequency Response
- A/D and D/A Conversion
- Sampling of DT Signals
- Summary

DT Processing of CT Signals



- Reason: take advantage of the vast variety of general- or special-purpose DT signal processing devices
- Overall system is equivalent to a CT system: Input and output are CT
- The CT system depends on DT system & Sampling rate
- What is the equivalent (effective) frequency response of the overall system? Find the relation
    1. between $x_c(t)$ and $x[n]$
    2. Next between y[n] and x[n]
    3. Finally between $y_r(t)$ and y[n]

$$H_{eff}(j\Omega) = \begin{cases} H(e^{j\omega}) & |\omega| < \pi;\ \omega = \Omega T \\ 0 & \text{otherwise} \end{cases}$$



Effective Frequency Response

- Input CT to DT $x[n] = x_c(nT)$ $X(e^{j\omega}) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c\left(j\left(\frac{\omega}{T} - \frac{2\pi k}{T}\right)\right)$
- Assume a DT LTI system

$$Y(e^{j\omega}) = H(e^{j\omega})X(e^{j\omega}) = \frac{1}{T}H(e^{j\omega})\sum_{k=-\infty}^{\infty} X_c\left(j\left(\frac{\omega}{T} - \frac{2\pi k}{T}\right)\right)$$

- Output DT to CT

$$y_r(t) = \sum_{n=-\infty}^{\infty} y[n]\frac{\sin[\pi(t-nT)/T]}{\pi(t-nT)/T} \qquad Y_r(j\Omega) = \begin{cases} TY(e^{j\Omega T}) & |\Omega| < \pi/T \\ 0 & \text{otherwise} \end{cases}$$

- Output frequency response

$$Y_r(j\Omega) = \begin{cases} H(e^{j\Omega T})X_c(j\Omega) & |\Omega| < \pi/T \\ 0 & \text{otherwise} \end{cases}$$

- Effective Frequency Response

$$Y_r(j\Omega) = H_{eff}(j\Omega)X_c(j\Omega) \qquad H_{eff}(j\Omega) = \begin{cases} H(e^{j\Omega T}) & |\Omega| < \pi/T \\ 0 & \text{otherwise} \end{cases}$$

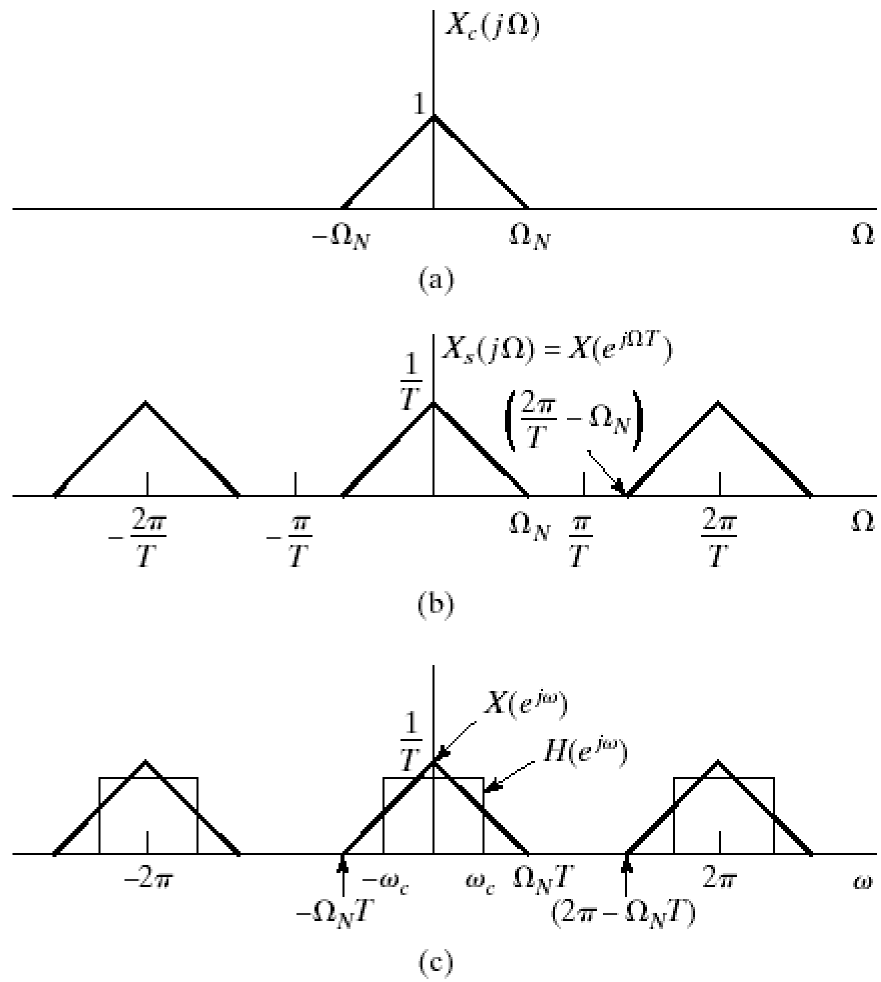
Example 4.3:

Ideal LPF implemented as a DT system

CT input signal

Sampled CT input signal

Apply DT LPF



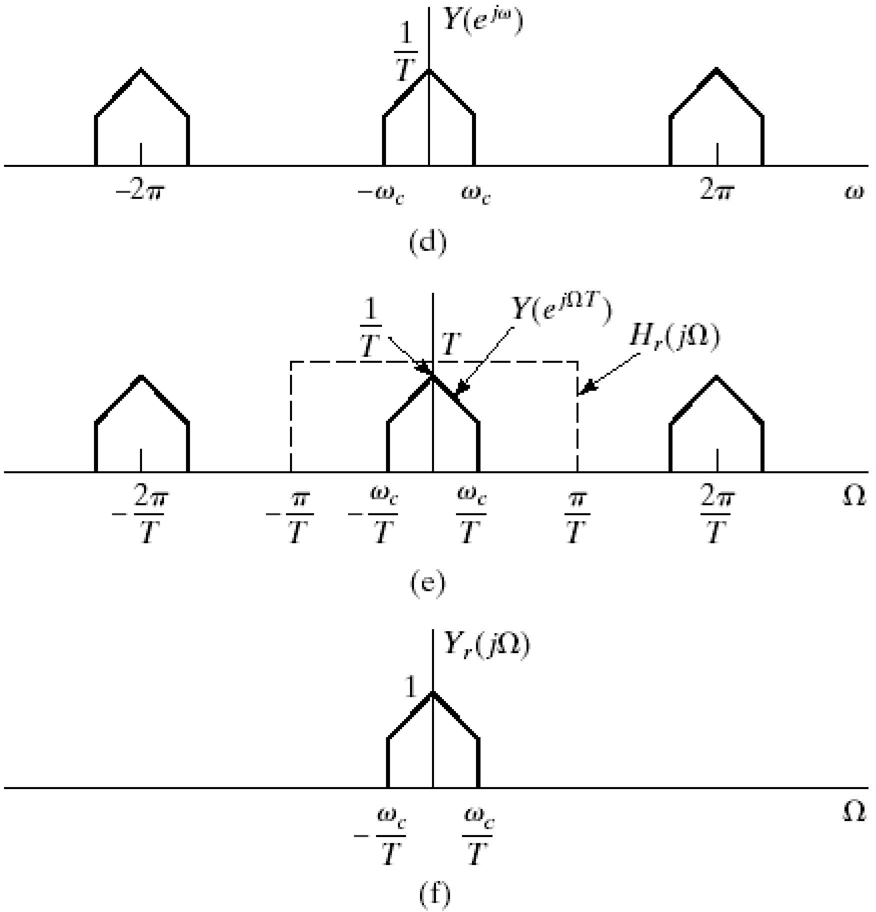
Example 4.3:

Ideal LPF implemented as a DT system

Signal after DT LPF is applied

Application of reconstruction filter

Output CT signal after reconstruction



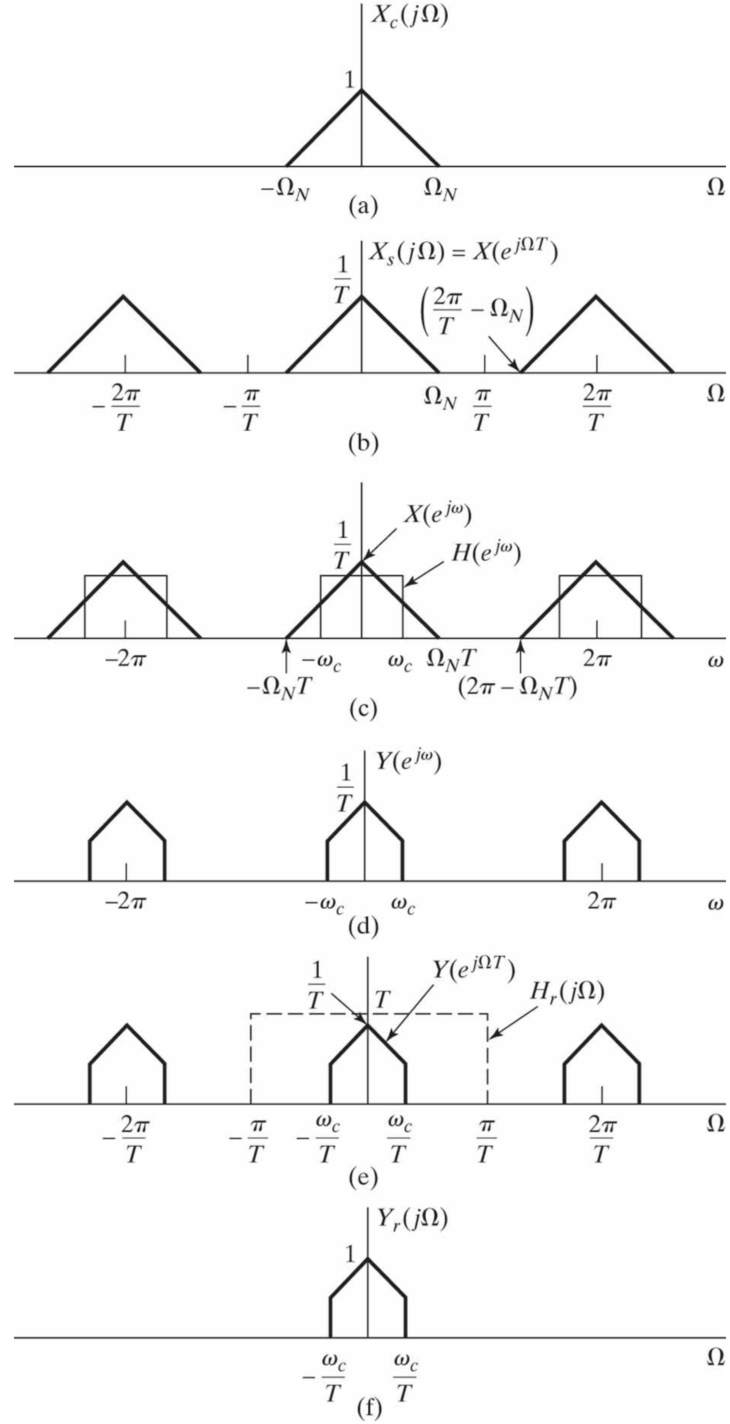
Example 4.3: Summary Ideal LPF implemented as a DT system

(a) FT of a bandlimited input signal.
(b) FT of sampled input plotted as a function of CT frequency Ω.
(c) FT $X(e^{j\omega})$ of sequence of samples and frequency response $H(e^{j\omega})$ of DT system plotted versus ω.

(d) FT of output of DT system.

(e) FT of output of DT system and frequency response of ideal reconstruction filter plotted versus Ω.

(f) FT of output.



Freq. response of an integrator

- H(s) of an ideal integrator is:

$$H(s)=\frac{1}{s} \quad \text{and} \quad H(j\omega)=\frac{1}{j\omega}=\frac{-j}{\omega}=\frac{1}{\omega}e^{-j\pi/2}$$

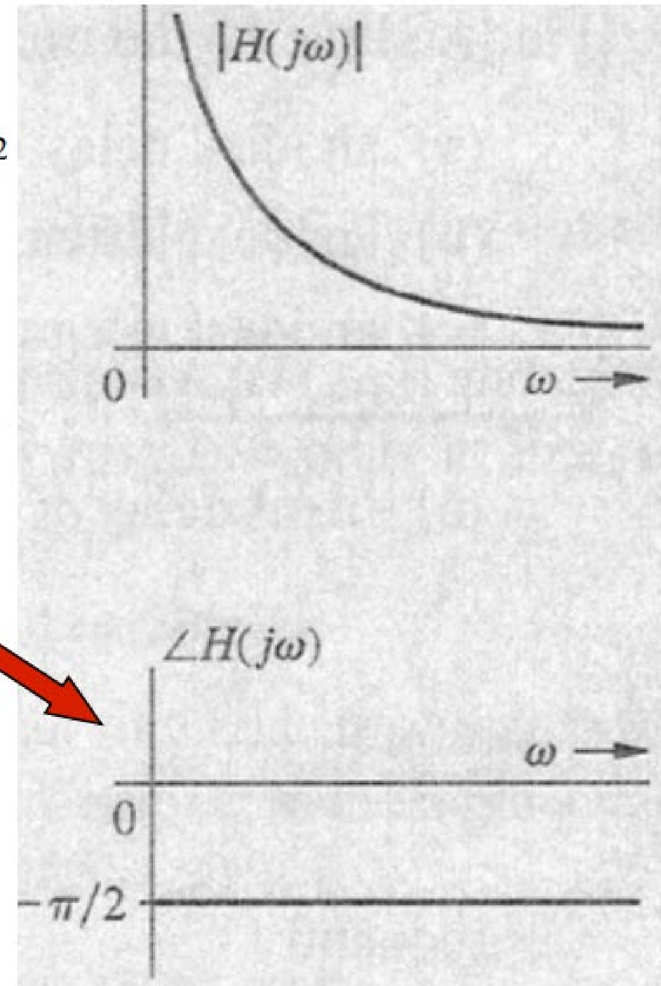
- Therefore

$$|H(j\omega)|=\frac{1}{\omega} \quad \text{and} \quad \angle H(j\omega)=-\frac{\pi}{2}$$

- This agrees with:

$$\int \cos\omega t\, dt=\frac{1}{\omega}\sin\omega t=\frac{1}{\omega}\cos(\omega t-\pi/2)$$

- That's why integrator is a nice component to work with – it suppresses high frequency component (i.e. noise!).
  - ➔An integrator acts like a LPF



Freq. response of a differentiator

- H(s) of an ideal differentiator is:

$$H(s) = s \quad \text{and} \quad H(j\omega) = j\omega = \omega e^{j\pi/2}$$

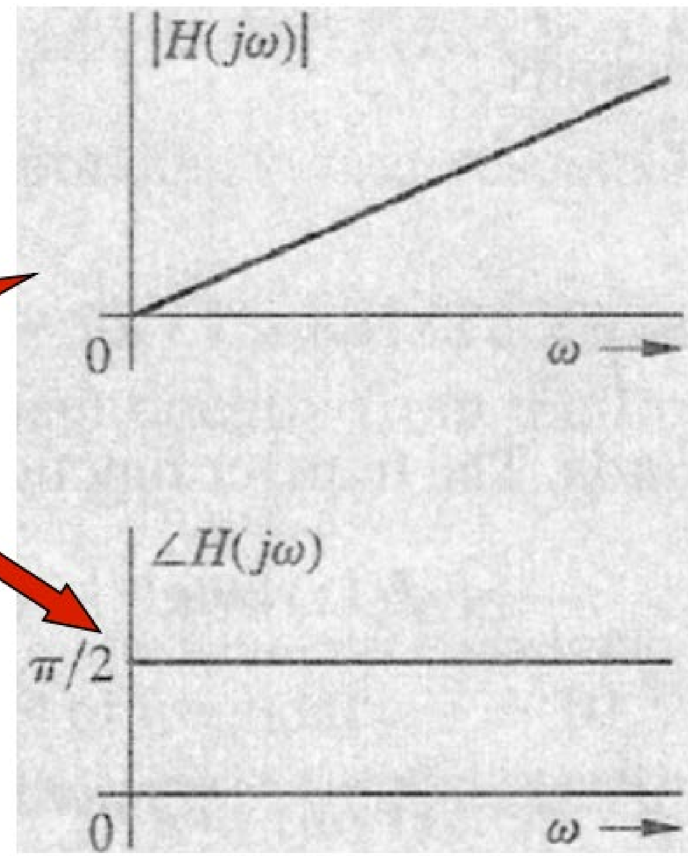
- Therefore

$$|H(j\omega)| = \omega \quad \text{and} \quad \angle H(j\omega) = \frac{\pi}{2}$$

- This agrees with:

$$\frac{d}{dt}(\cos\omega t) = -\omega \sin\omega t = \omega\cos(\omega t + \pi/2)$$

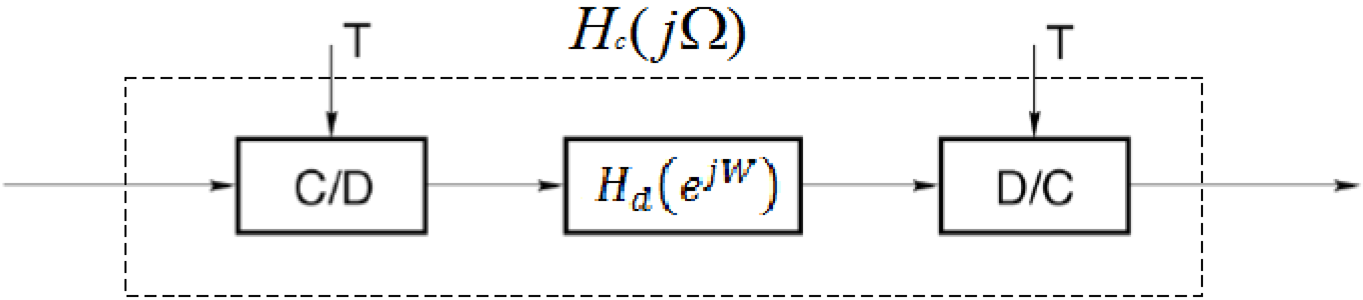
- That's why differentiator is not a nice component to work with – it amplifies high frequency component (i.e. noise!).



➔A differentiator acts like a HPF

Example 4.4: Digital Differentiator

Construction of Band-limited Digital Differentiator



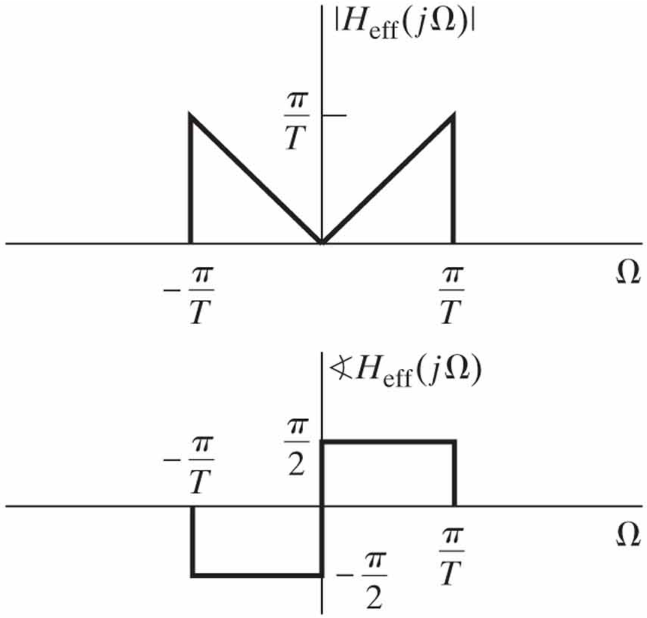
Desired: $H_c(j\Omega) = \begin{cases} j\Omega & , \ |\Omega| < \Omega_c \\ 0 & , \ |\Omega| > \Omega_c \end{cases}$

Set $\Omega_s = 2\Omega_c \Rightarrow T = \dfrac{2\Pi}{\Omega_s} = \dfrac{\Pi}{\Omega_c}$. Assume $\Omega_M < \Omega_c$ (Nyquist rate met)

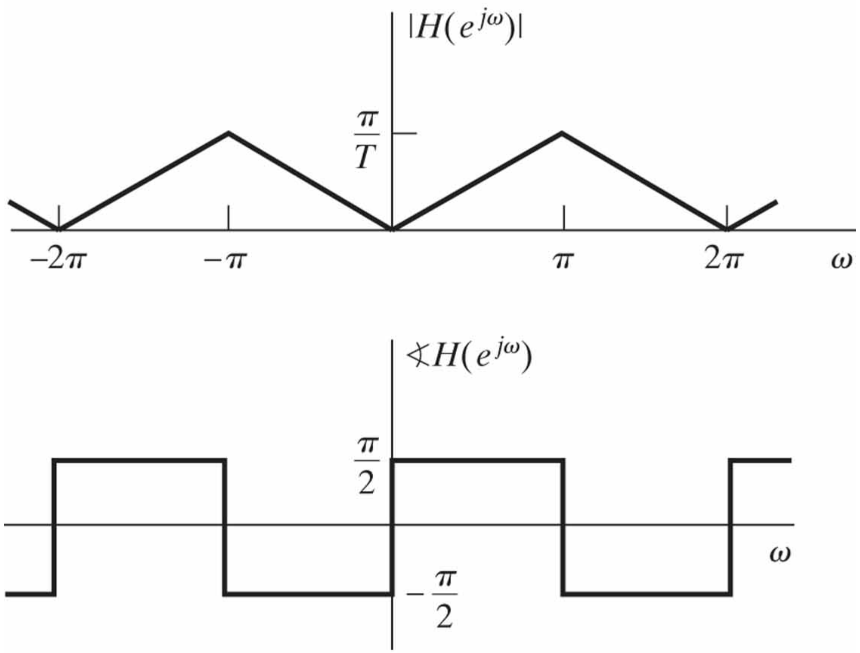
Choice for $H_d(e^{jW})$:

$$H_d(e^{jW}) = \begin{cases} H_c(j(\omega/T)), & |\omega| < \Pi \\ \text{periodic,} & , |\omega| \geq \Pi \end{cases}$$

$$= j\left(\frac{\omega}{T}\right) \ , |\omega| < \Pi$$



(a)



(b)

Example 4.4: Digital Differentiator

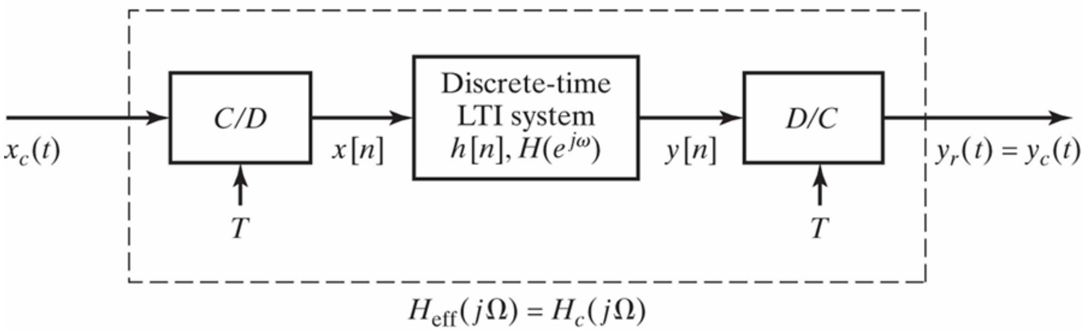
a) Frequency response of a CT ideal bandlimited differentiator $H_c(j\Omega) = j\Omega$, $|\Omega| < \pi/T$

b) Frequency response of a DT filter to implement a CT bandlimited differentiator

Outline

- Introduction
- Sampling
- Reconstruction
- The Effect of Under-sampling: Aliasing
- DT Processing of CT Signals
  - DT processing: Effective CT Frequency Response
  - DT from CT: Impulse invariance
- CT processing: Effective DT Frequency Response
- A/D and D/A Conversion
- Sampling of DT Signals
- Summary

DT from CT: Impulse Invariance



- Given a CT system $H_c(j\Omega)$
- Find a DT system response $H(e^{j\omega})$ such that the effective response $H_{eff}(j\Omega)=H_c(j\Omega)$
- Impulse Invariance: The sampling of the CT impulse response h(t) to produce the DT impulse response h[n]
  - $h[n] = Th_c(nT)$; $h[n]$ the impulse-invariant version of the CT system
- If h(t) is appropriately band-limited, i.e., $H_c(j\Omega) = 0$ for $|\Omega| \geq \pi/T$, then $H\left(e^{j\omega}\right) = H_c(j\,\omega/T)$ with $|\omega| < \pi,\ or\,|\Omega| < \frac{\pi}{T}$
  - $H(e^{j\omega})$ is $H_c(j\Omega)$ with linearly-scaled frequency i.e., $\omega = \Omega T$
- If the CT filter has poles at $s = s_k$, these poles are translated to poles at $z = e^{s_k T}$; T is sampling period
- If the CT filter is causal and stable, then the DT filter will be causal and stable as well

Example: Impulse Invariance

- Ideal low-pass DT filter by impulse invariance $H_c(j\Omega) = \begin{cases} 1 & |\Omega| < \Omega_c \\ 0 & \text{else} \end{cases}$

- The impulse response of CT system is $h_c(t) = \dfrac{\sin(\Omega_c t)}{\pi t}$

- Obtain DT impulse response via impulse invariance

$$h[n] = T h_c(nT) = T\,\frac{\sin(\Omega_c nT)}{\pi nT} = \frac{\sin(\omega_c n)}{\pi n}$$

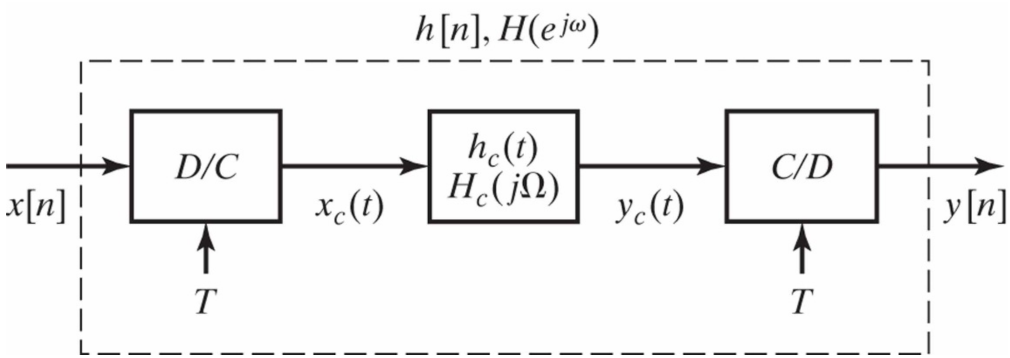
- The frequency response of the DT system is

$$H_c(e^{j\omega}) = \begin{cases} 1 & |\omega| < \omega_c \\ 0 & \omega_c < |\omega| \le \pi \end{cases}$$

Outline

- Introduction
- Sampling
- Reconstruction
- The Effect of Under-sampling: Aliasing
- DT Processing of CT Signals
  - DT processing: Effective CT Frequency Response
  - DT from CT: Impulse invariance
- CT processing of DT signals: Effective DT Frequency Response
- A/D and D/A Conversion
- Sampling of Discrete-Time Signals
- Summary

CT processing of DT signals



$$x[n] \rightarrow x_s(t) \rightarrow x_c(t) \rightarrow y_c(t) \rightarrow y[n]$$

$$x_c(t) = \sum_{n=-\infty}^{+\infty} x(nT) h_r(t-nT) = \sum_{n=-\infty}^{\infty} x[n] \frac{\sin\left[\pi(t-nT)/T\right]}{\pi(t-nT)/T}$$

$$X_c(j\Omega) = T X(e^{j\omega})$$

$$Y_c(j\Omega) = H_c(j\Omega) X_c(j\Omega)$$

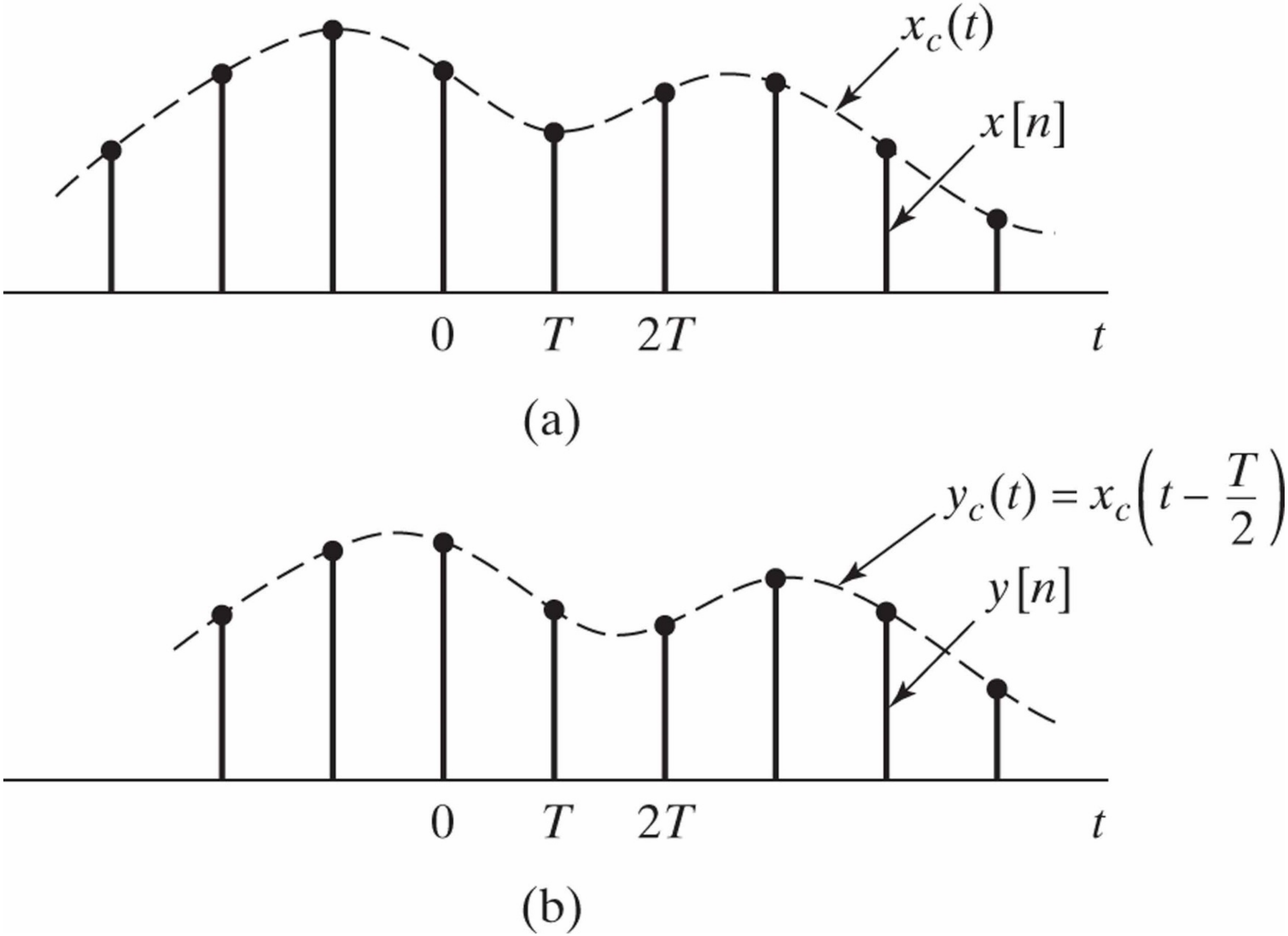
$$Y(e^{j\omega}) = \frac{1}{T} Y_c\left(j\frac{\omega}{T}\right)$$

$$H(e^{j\omega}) = \frac{Y(e^{j\omega})}{X(e^{j\omega})} = H_c\left(j\frac{\omega}{T}\right)$$

$$H\left(e^{j\omega}\right) = \begin{cases} H_c\left(j\Omega\right) & |\omega| < \pi;\ \Omega = \omega/T \\ 0 & \text{otherwise} \end{cases}$$

$$y_c(t) = \sum_{n=-\infty}^{\infty} y[n] \frac{\sin\left[\pi(t-nT)/T\right]}{\pi(t-nT)/T}$$

CT processing of the DT sequence *x[n]* can produce a new sequence *y[n]* with a “half-sample” delay



Example 1: DT System for Arbitrary Delay

$$y[n] = x[n-\Delta]; \quad \Delta \text{ real or integer}$$

$$H\left(e^{j\omega}\right) = e^{j\omega\Delta}; \quad |\omega| < \pi; \qquad H(z) = z^{\Delta}$$

- For integer delay values, this DT system is meaningful:
  - the samples y(n) are equal to the delayed samples of x(n)

- For real delay values,
- ➢ y(n) would lie somewhere between the known samples of x(n)
- ➢ The unknown y(n) would then have to be obtained by interpolation from the known x(n)

$$y[n] = x[n-\Delta] = \sum_{k=-\infty}^{\infty} x[n]\,\mathrm{sinc}(n-\Delta-k);$$

$$\text{N integer; } \Delta \text{ real; T} = 1 \text{ sampling period}$$

- Conclusion: producing a fractional delay requires reconstruction of the DT signal and shifted resampling of the resulting CT
  - T=1 to simplify notation

Ex. 1°: Noninteger delay system $H(e^{j\omega}) = e^{-j\omega\Delta}$, Δ: fraction

$$H_c(j\Omega) = H(e^{j\Omega T}) = e^{-j\Omega\Delta T}$$

$$y_c(t) = x_c(t - \Delta T)$$

$$y(n) = y_c(nT) = x_c(nT - \Delta T)$$

Note that $$x_c(t) = \sum_{k=-\infty}^{\infty} x(k) \frac{\sin[\pi(t-kT)/T]}{\pi(t-kT)/T}$$

Therefore $$y(n) = x_c(nT - \Delta T)$$

$$= \sum_{k=-\infty}^{\infty} x(k) \frac{\sin[\pi(t-\Delta T-kT)/T]}{\pi(t-\Delta T-kT)/T}\Bigg|_{t=nT}$$

$$= x(n) * h(n)$$

where $$h(n) = \frac{\sin\pi(n-\Delta)}{\pi(n-\Delta)} \qquad -\infty < n < \infty$$

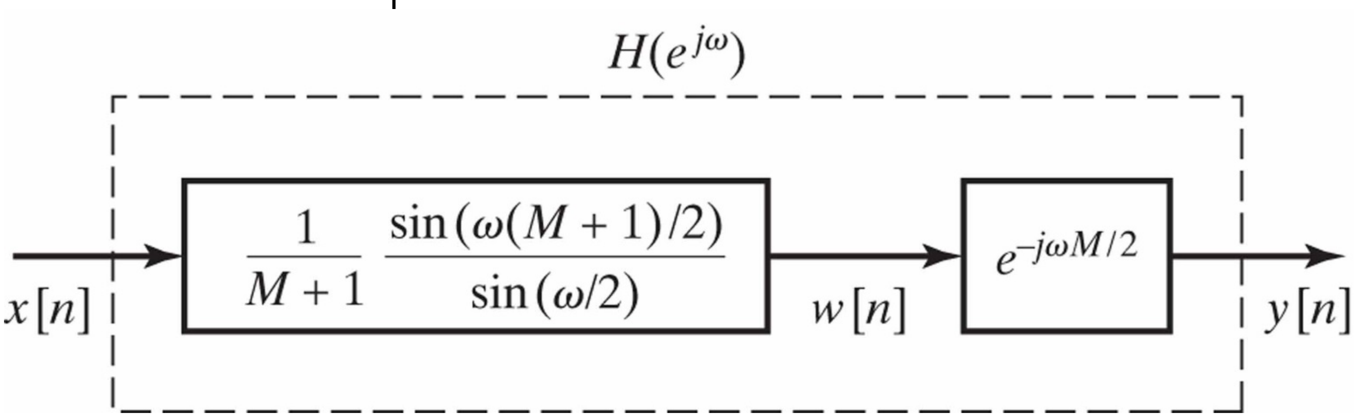
Example 2: DT Moving Average System with fractional delay

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

$$H(e^{j\omega}) = \frac{1}{M+1} \sum_{k=0}^{M} e^{-j\omega k}$$

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

$$h[n] = \begin{cases} \dfrac{1}{M+1}, & M \le n \le 0 \\ 0, & \text{otherwise} \end{cases}$$



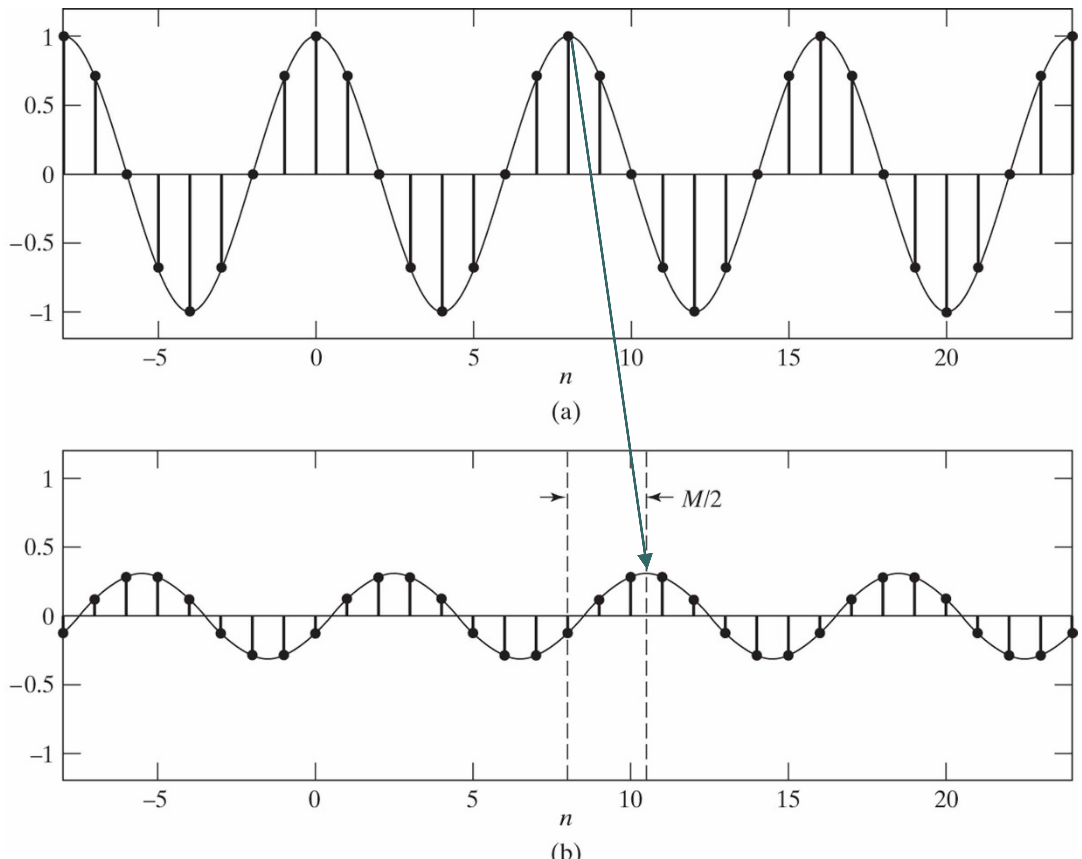
The moving-average system represented as a cascade of two systems

Example 2: DT Moving Average System with fractional delay

6-point moving-average filter

- Assume M=5 ➔ $H\left(e^{j\omega}\right)$ causes a delay of 2.5 samples
- Assume $x[n] = \cos(0.25\pi n) = \cos(\frac{\pi}{4}n)$ ➔

$$y[n] = \frac{1}{2} e^{j\frac{\pi}{4}n} H\left(e^{j\frac{\pi}{2}}\right) + \frac{1}{2} e^{-j\frac{\pi}{4}n} H\left(e^{j\frac{\pi}{4}}\right) = 0.308 \cos\left(\frac{\pi}{4}(n-2.5)\right)$$



Summary

- DT processing: Effective CT Frequency Response

$$H_{eff}(j\Omega) = \begin{cases} H(e^{j\omega}) & |\omega| < \pi;\ \omega = \Omega T \\ 0 & \text{otherwise} \end{cases}$$

- DT from CT: Impulse invariance
  - Sampling of $h_c(t)$

$$h[n] = T h_c(nT)$$

- CT processing: Effective DT Frequency Response

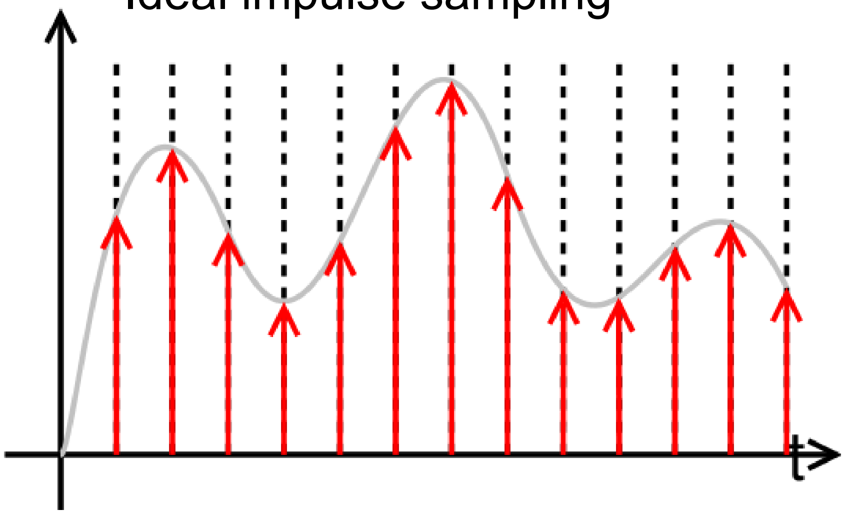
$$H(e^{j\omega}) = \begin{cases} H_c(j\Omega) & |\omega| < \pi;\ \omega = \Omega T \\ 0 & \text{otherwise} \end{cases}$$

Outline

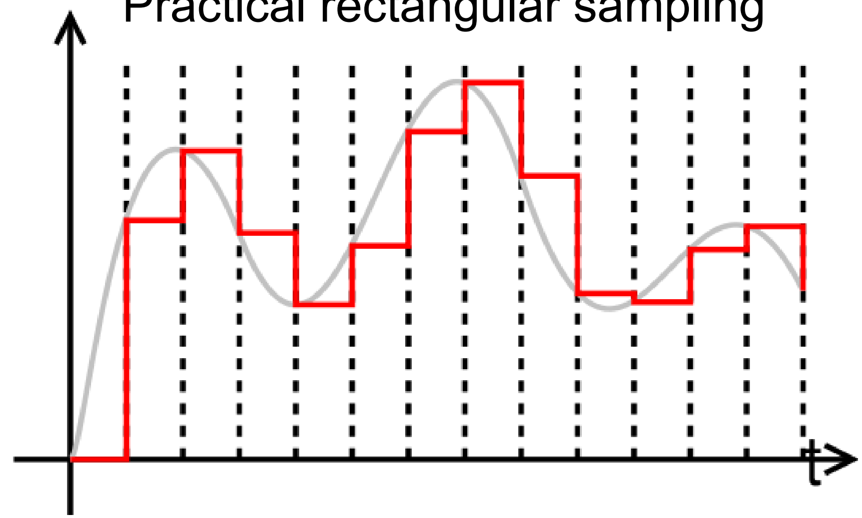
- Introduction
- Sampling
- Reconstruction
- The Effect of Under-sampling: Aliasing
- Examples
- DT Processing of CT Signals
- A/D and D/A Conversion:
  - Practical sampling & reconstruction
- Sampling of DT Signals
- Summary

A/D and D/A Conversion

Ideal impulse sampling



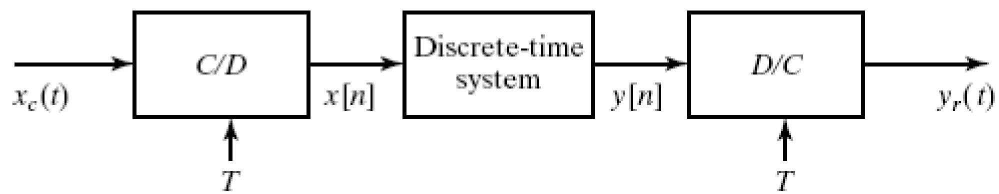
Practical rectangular sampling



- A non-ideal system that samples *x(t)* at a given instant and holds that value until the next instant, at which a sample should be taken

A/D and D/A Conversion

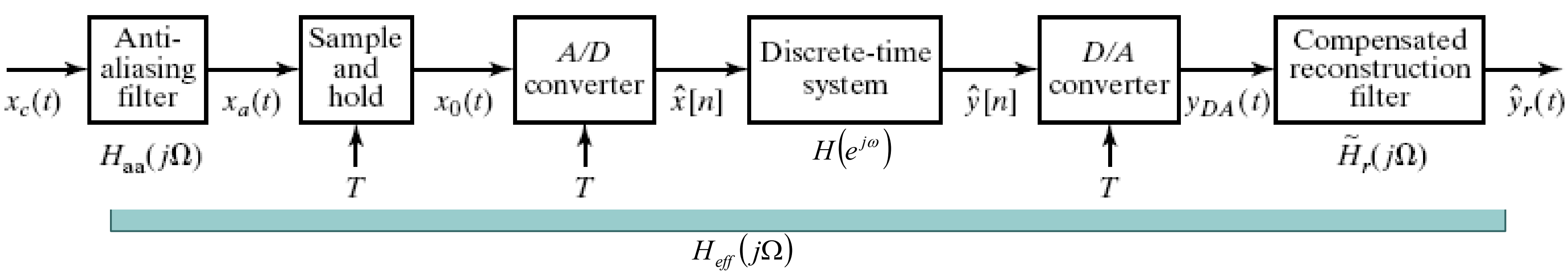
- Up to this point we assumed ideal D/C and C/D conversion



- In practice, CT signals are not perfectly band-limited

  ➔D/C and C/D converters can only be approximated with D/A and A/D converters

- A more realistic model for DSP



- In the following we discuss each of these blocks

AntiAliasing filter (AAF)

- AAF is desirable

1. to minimize sampling rate: Minimizes amount of data to process
2. to reduce noise: no point of sampling high frequencies that are not of interest (e.g., noise)

- An ideal anti-aliasing filter AAF

$$H_{aa}(j\Omega) = \begin{cases} 1 & |\Omega| < \Omega_c < \pi / T \\ 0 & |\Omega| > \Omega_c \end{cases}$$

- The effective response via DT LPF is

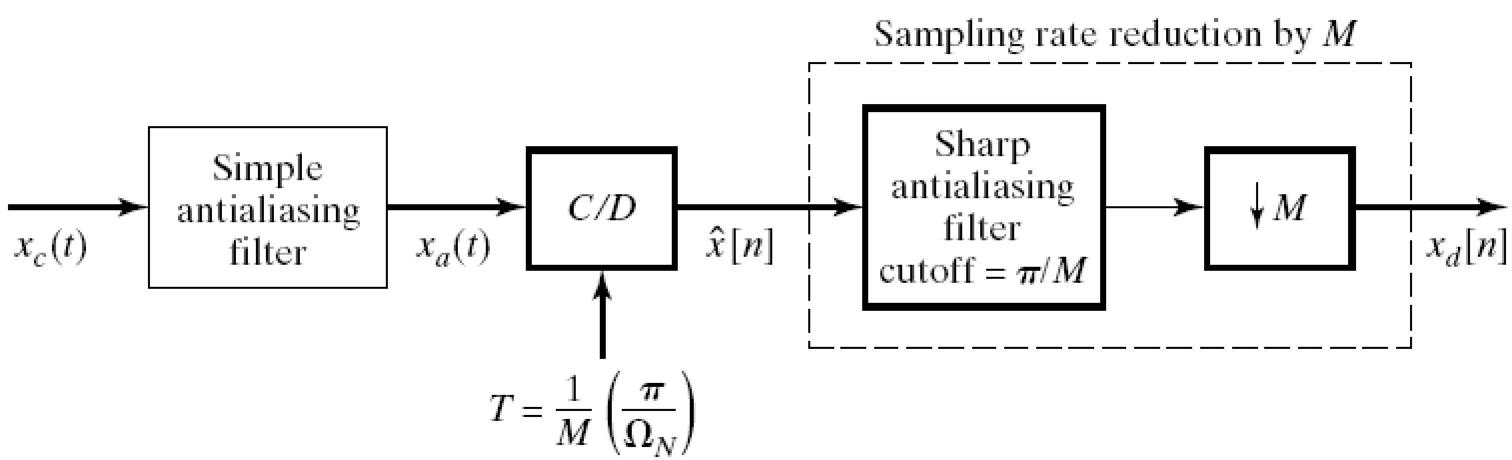
$$H_{eff}(j\Omega) = \begin{cases} H(e^{j\Omega T}) & |\Omega| < \Omega_c \\ 0 & |\Omega| > \Omega_c \end{cases}$$

- In practice an ideal AAF is not possible; hence $H_{eff}(j\Omega) \approx H_{aa}(j\Omega) H(e^{j\Omega T})$

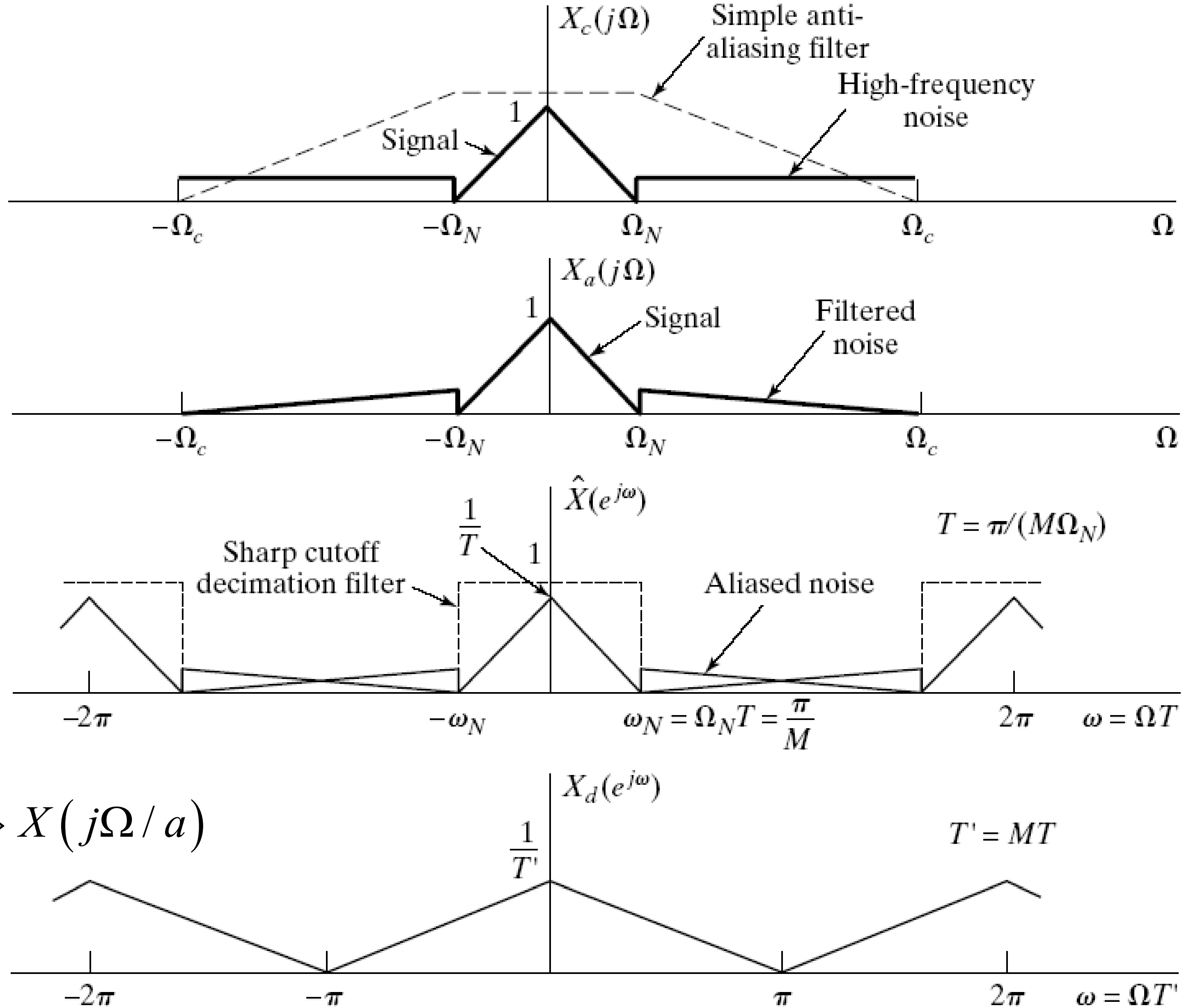
➔ $H_{aa}(j\Omega)$ is a s<u>harp-cutoff analog filters, which are expensive</u>

Simplifying AAF : Oversampled A/D Conversion

1. have a simple (gradual cutoff) analog anti-aliasing filter
2. use higher than required sampling rate $\Omega_s = 2M\Omega_N \Rightarrow \mathrm{T} = \frac{1}{M}\frac{\pi}{\Omega_N}$
3. implement sharp DT anti-aliasing filter $\omega_c = \frac{\pi}{M}$
4. downsample to desired sampling rate $\Omega'_s = 2M\Omega_N/\mathrm{M} = 2\Omega_N$



Oversampled A/D Conversion: simplifying AAF



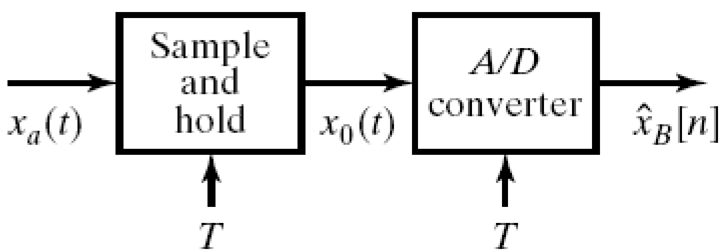
$$x(at) \Leftrightarrow X\left(j\Omega / a\right)$$

A/D Conversion

- Two steps: sampling in time ***t*** and quantization of the amplitude ***x***
- Sampling ➔ $x[n] = x(nT)$
- Quantization: map amplitude values into a set of discrete values ➔$x'[n] = Q(x[n])$
  - Quantization error: $e[n] = x[n]-x'[n]$

A/D Conversion

- Ideal C/D converters convert CT signals into <u>infinite-precision</u> DT signals
- onverters as the cascade of



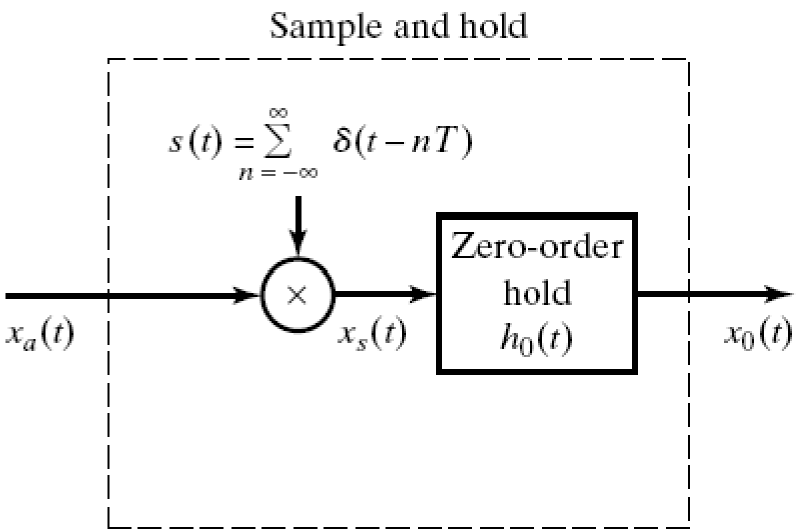
- The sample-and-hold device holds current/voltage constant
- The A/D converter converts current/voltage into <u>finite-precisions</u> number
- The ideal <u>sample-and-hold</u> device has the output

$$x_0(t) = \sum_{n=-\infty}^{\infty} x[n]\; h_0(t - nT)$$

$$h_0(t) = \begin{cases} 1, & 0 < t < T \\ 0, & \text{else} \end{cases}$$

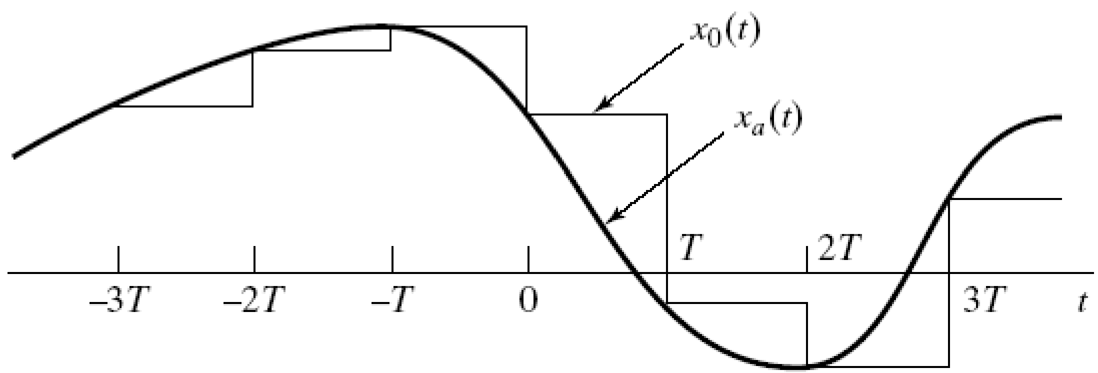
$$H_0(j\Omega) = e^{-j\Omega T/2}\left[\frac{2\sin(\Omega T/2)}{\Omega}\right]$$

A/D Conversion : Ideal Sample and Hold



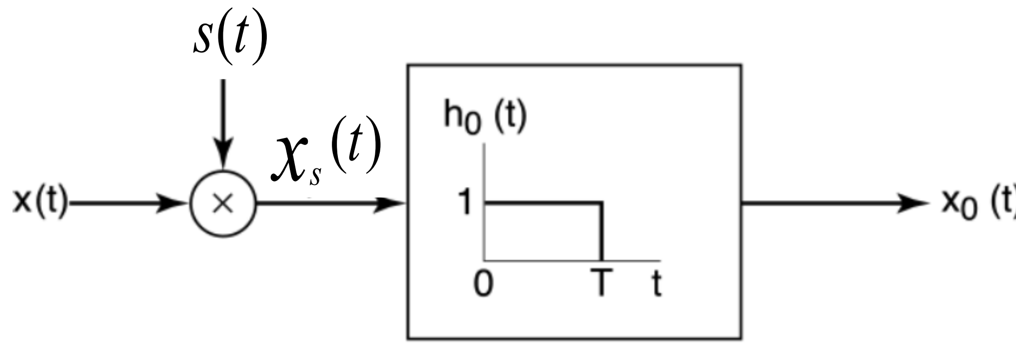
$$x_0(t) = \sum_{n=-\infty}^{\infty} x[n] h_0(t - nT)$$

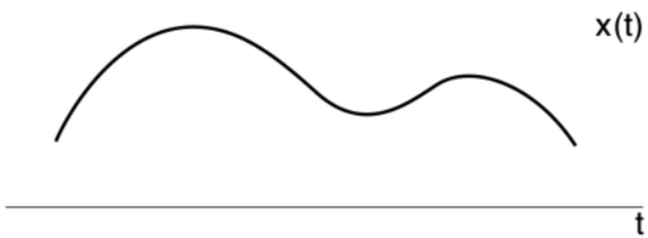
- Time-domain:



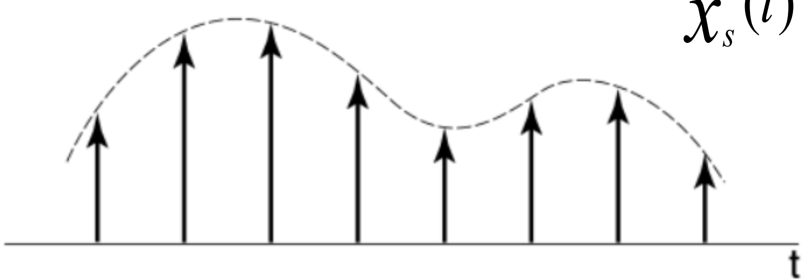
A/D Conversion: Sampling with Zero-Order Hold

- Zero-order hold: impulse-train sampling followed by an LTI system with a rectangular impulse response

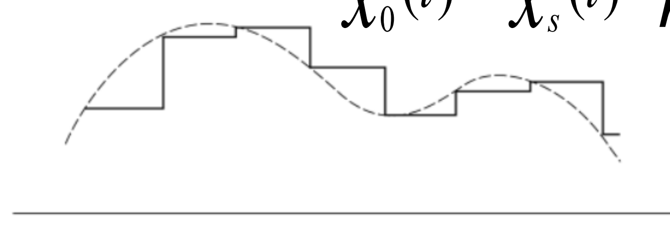




$x_s(t)$



$$x_0(t) = x_s(t) * h_0(t)$$



Outline

- Introduction
- Sampling
- Reconstruction
- The Effect of Under-sampling: Aliasing
- Examples
- DT Processing of CT Signals
- A/D and D/A Conversion
  - A/D conversion
  - D/A conversion
- Sampling of DT Signals
- Summary

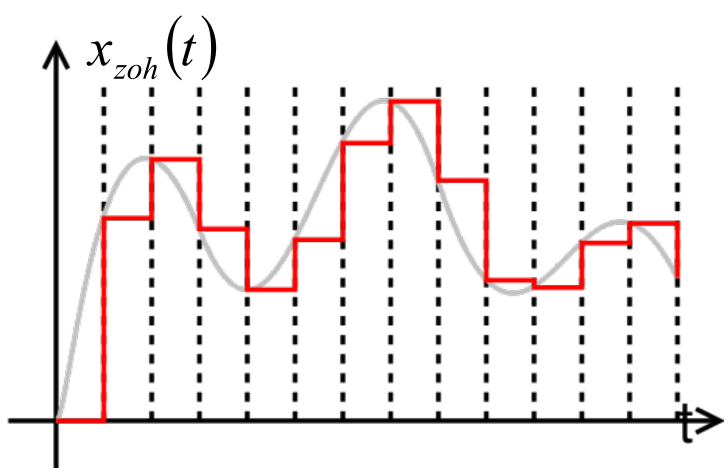
D/A conversion: Reconstruction Methods

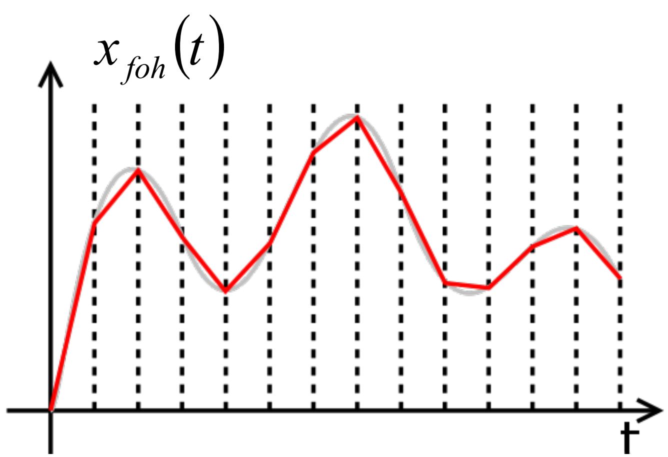
- Reconstruction: connecting samples x[n] using interpolation kernels
  a) Zero-Order Hold: D/A Conversion

     e.g. scanned images

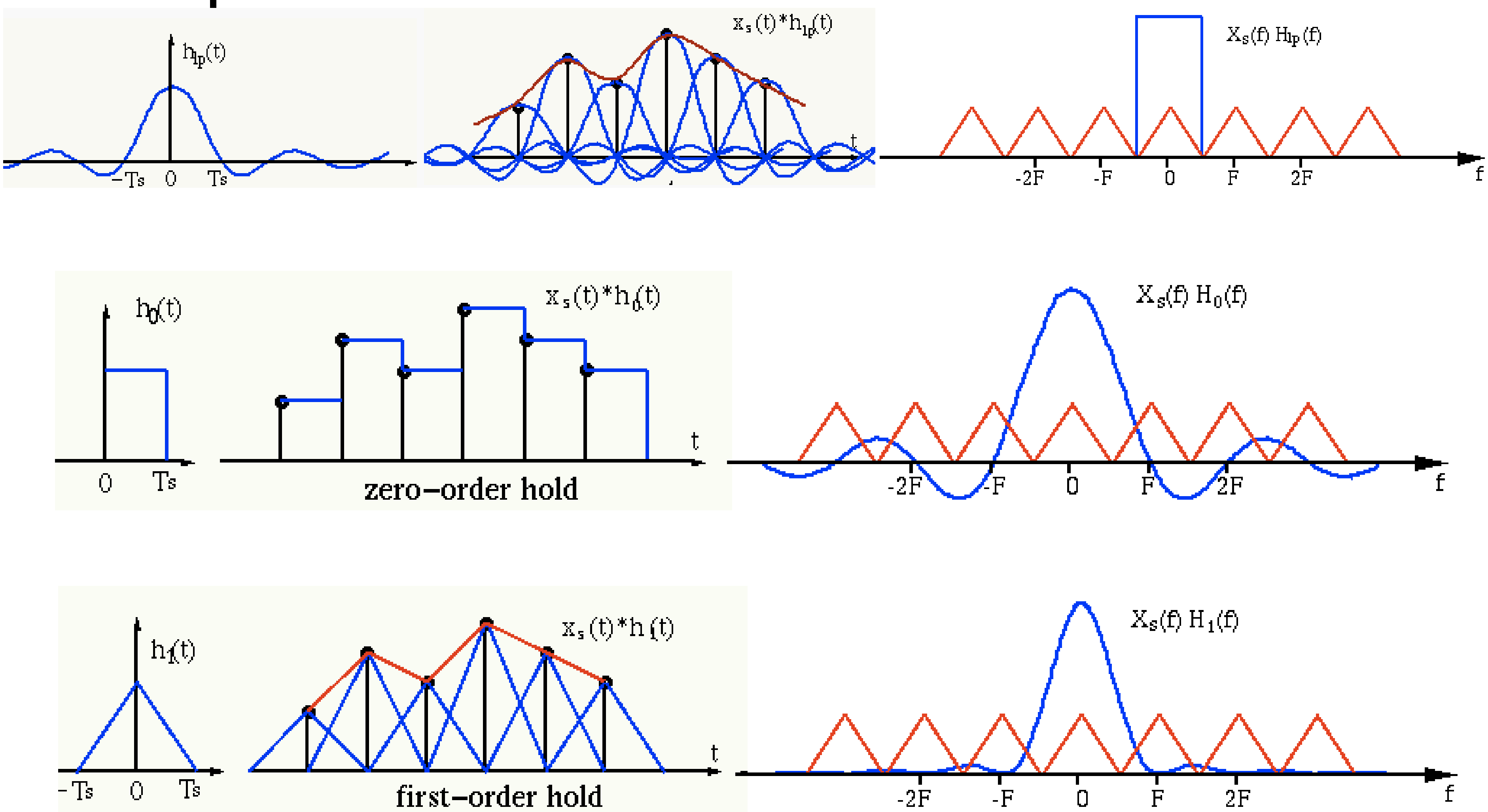
  b) First-Order Hold: D/A Conversion

     Linear interpolation: commonly used in plotting





Ideal, zero-order hold, and first-order hold reconstruction



D/A Conversion: Zero-Order Hold:

- Perfect reconstruction requires filtering with ideal LPF

$$X_r(j\Omega) = X\left(e^{j\Omega T}\right)H_r(j\Omega)$$

$$X\left(e^{j\Omega T}\right): \text{DTFT of sampled signal}$$

$$X_r(j\Omega): \text{FT of reconstructed signal}$$

- The ideal reconstruction filter

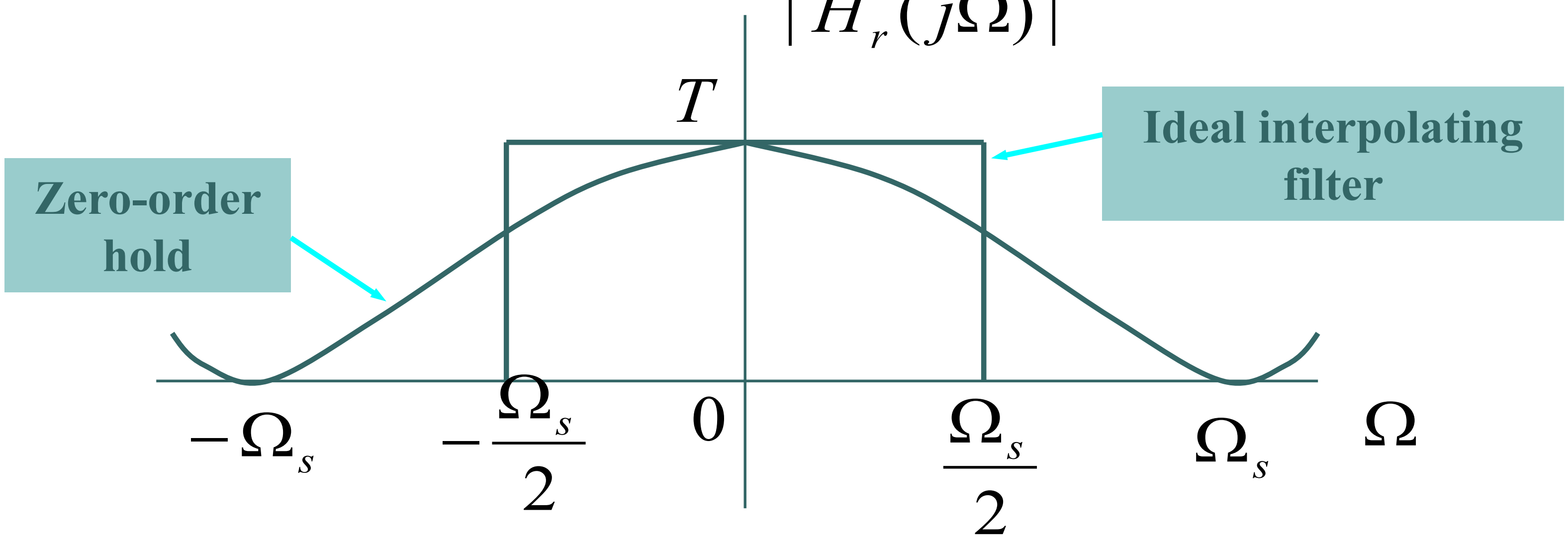
$$H_r(j\Omega) = \begin{cases} T & |\Omega| < \pi / T \\ 0 & |\Omega| > \pi / T \end{cases}$$

- The time domain reconstructed signal is

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

- In practice we cannot implement an ideal reconstruction filter

D/A Conversion: Reconstruction with Zero-Order Hold



D/A Conversion

- The practical D/A converter: Digital to analog converter + Analog LPF
- It takes a binary code and converts it into CT output

$$x_{DA}(t) = \sum_{n=-\infty}^{\infty} X_m \, \hat{x}_B[n] h_0(t-nT) = \sum_{n=-\infty}^{\infty} \hat{x}[n] h_0(t-nT)$$

- Using the additive noise model for quantization

$$x_{DA}(t) = x_0(t) + e_0(t) = \sum_{n=-\infty}^{\infty} x[n] h_0(t-nT) + \sum_{n=-\infty}^{\infty} e[n] h_0(t-nT)$$

- The signal component in frequency domain can be written as

Note: *$x[n]=x_a(nT)$*

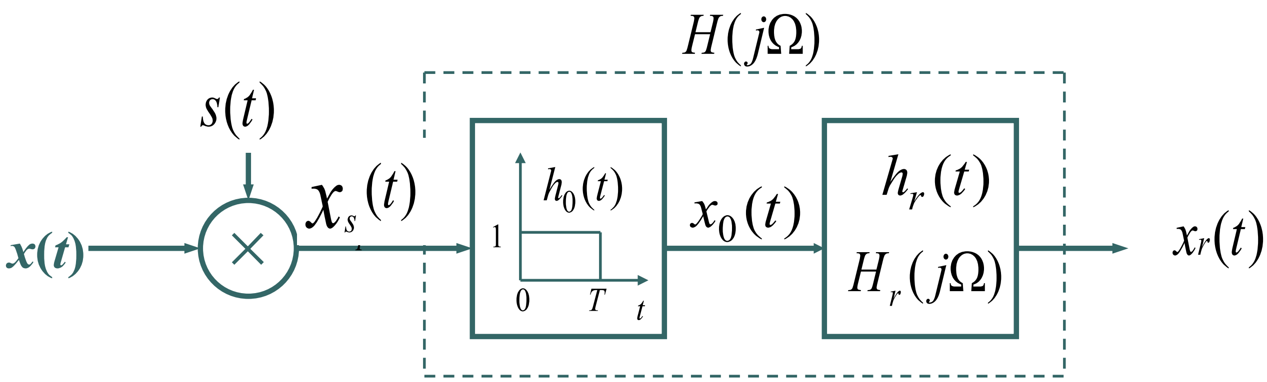
$$\mathrm{X}_0(j\Omega) = \mathrm{X}(e^{j\Omega T}) \mathrm{H}_0(j\Omega) \qquad X(e^{j\Omega T}) = \frac{1}{T} \sum_{k=-\infty}^{\infty} X_a(j(\Omega - k\Omega_s))$$

➔ To recover the desired signal component we need <u>a compensated reconstruction filter</u> of the form to get *$x_a(t)$ back*

$$\tilde{\mathrm{H}}_r(j\Omega) = \frac{\mathrm{H}_r(j\Omega)}{\mathrm{H}_0(j\Omega)}$$

D/A Conversion: Reconstruction with Zero-Order Hold

- Cascade of a zero-order hold with a reconstruction filter



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

$$H_r(j\Omega) = \frac{e^{j\Omega T/2} H(j\Omega)}{\dfrac{2\sin(\Omega T/2)}{\Omega}}$$

Compensated Reconstruction Filter

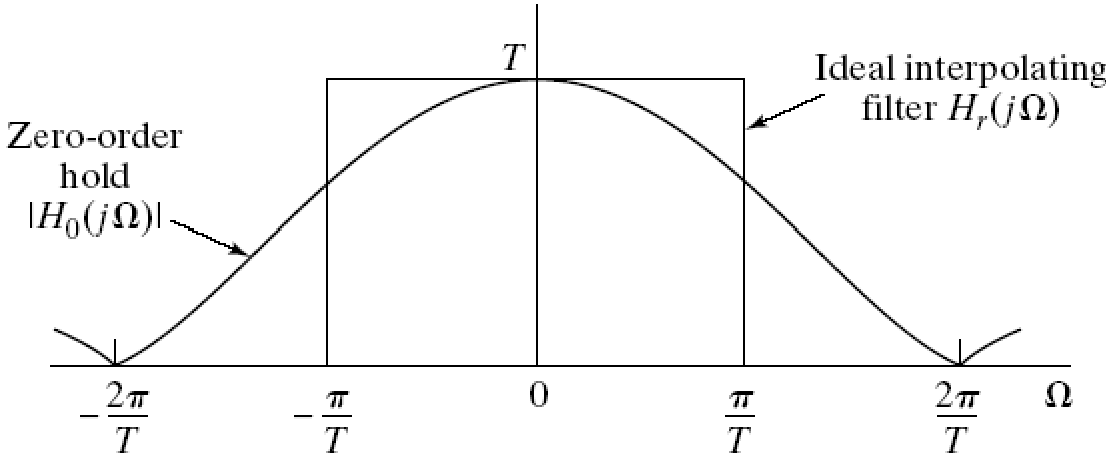
$$\tilde{H}_r(j\Omega) = \frac{H_r(j\Omega)}{H_0(j\Omega)}$$

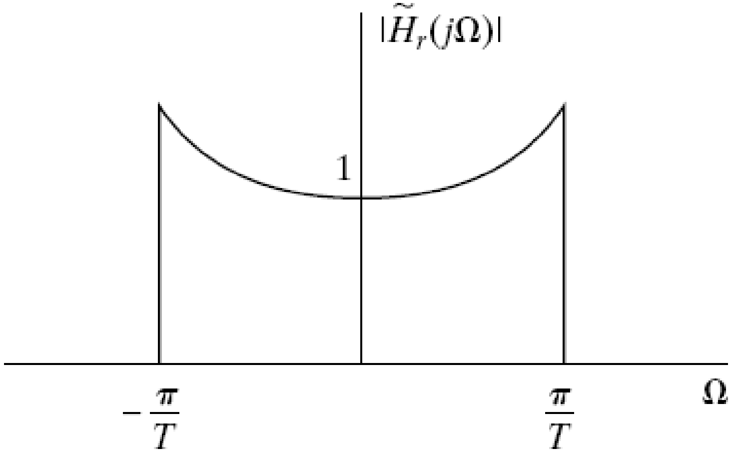
- The frequency response of <u>zero-order hold</u> is

$$H_0(j\Omega) = \frac{2\sin(\Omega T/2)}{\Omega} e^{-j\Omega T/2}$$

- Therefore the compensated reconstruction filter should be

$$\tilde{H}_r(j\Omega) = \begin{cases} \dfrac{\Omega T/2}{\sin(\Omega T/2)} e^{j\Omega T/2} & |\Omega| < \pi/T \\ 0 & |\Omega| > \pi/T \end{cases}$$

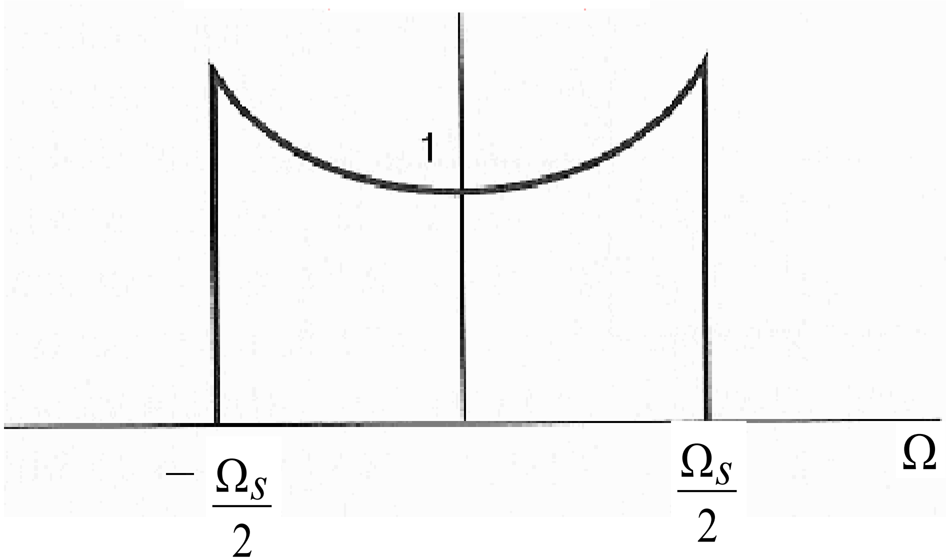




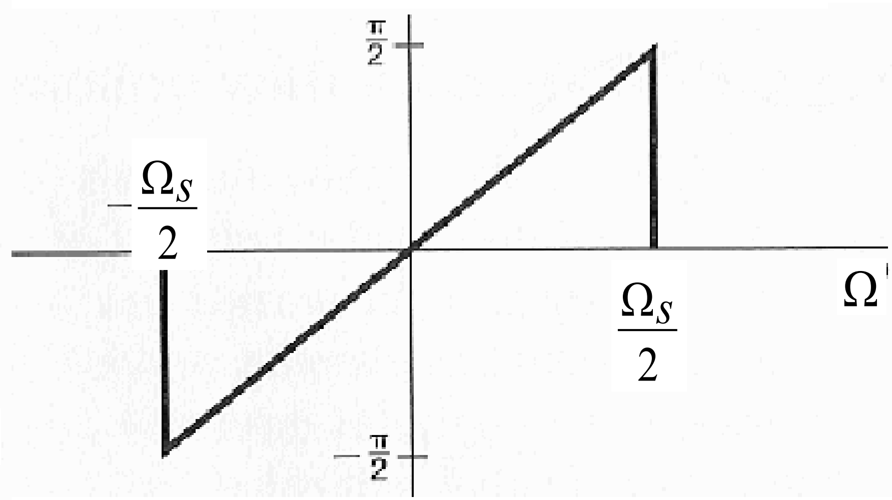
D/A Conversion: Reconstruction with Zero-Order Hold

Reconstruction filter

$|H_r(j\Omega)|$



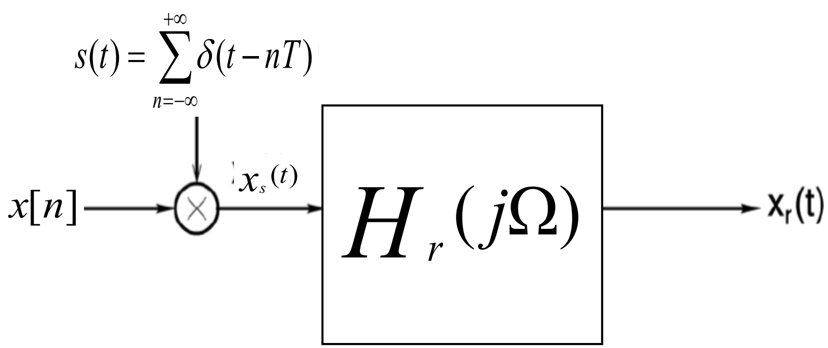
$|H_r(j\Omega)|$

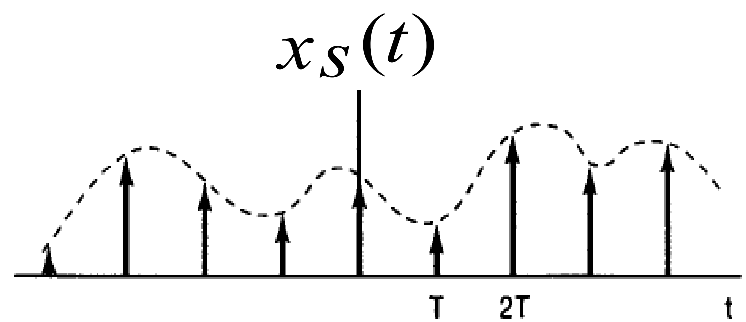


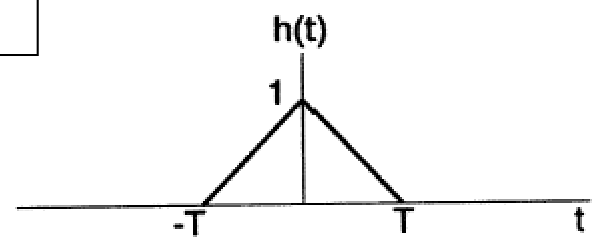
Magnitude and phase for the reconstruction filter for a zero-order hold

First-Order Hold: Linear interpolation

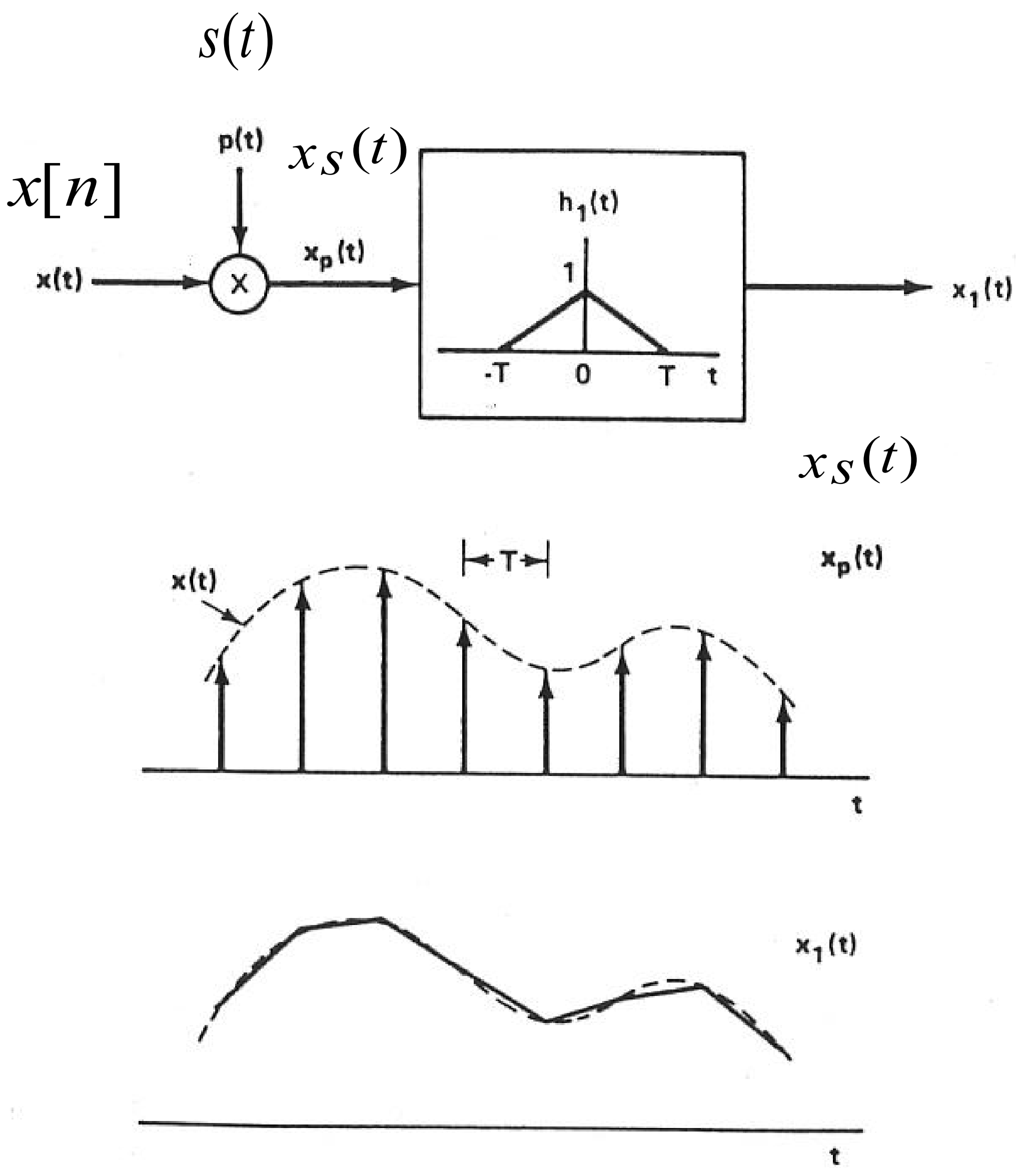
Impulse-train sampling followed by convolution with a triangular impulse response







First-Order Hold: Linear interpolation

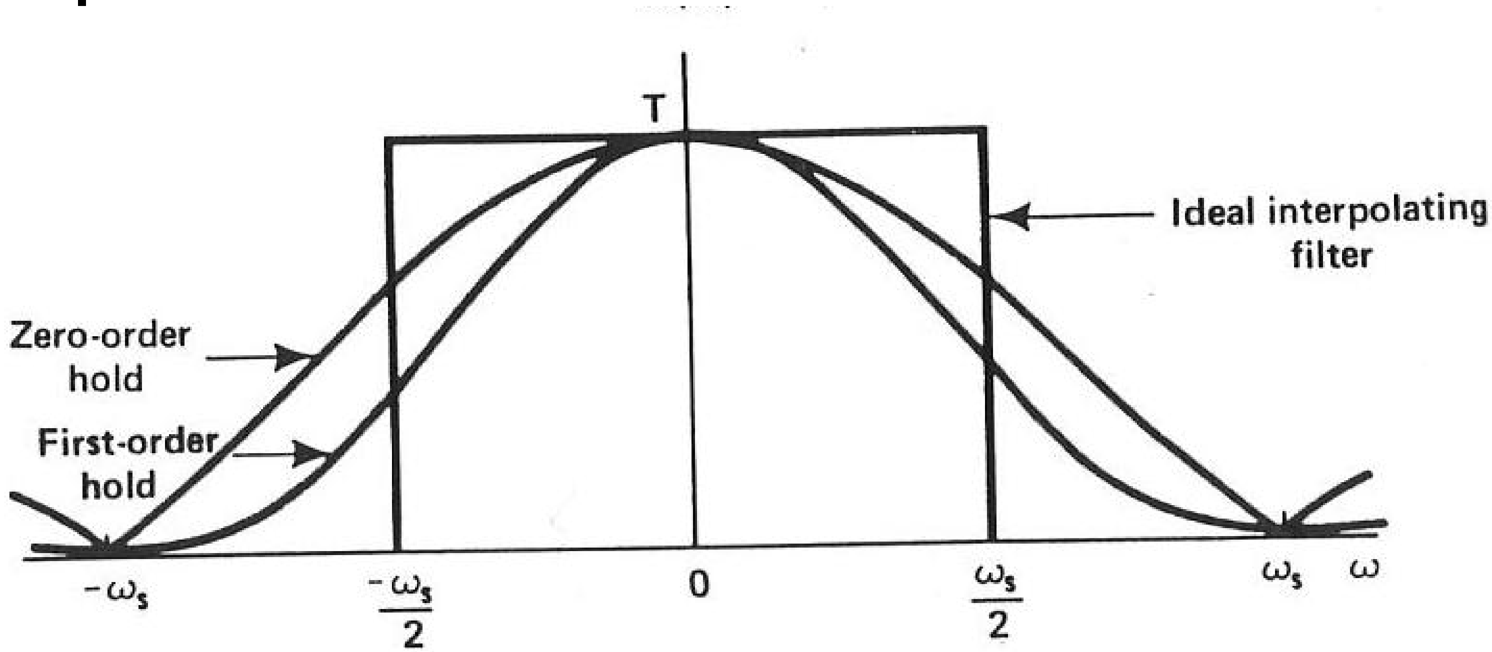


First-order versus zero-order hold

- First-order hold filter: the signal is reconstructed as a piecewise linear approximation to the original signal xc(t)
  - h(t) is a triangle

- Zero-order hold filter converts a DT signal to a CT signal by holding each sample value for one sample interval
  - h(t) is a square

Comparison of frequency responses of ideal lowpass, zero-order hold, and first-order hold reconstruction filters



- "zero-order" since the CT signal is a zeroth order polynomial between the sampling points
- "first-order" since the CT signal is a first order polynomial between the sampling points

Outline

- Introduction
- Sampling
- Reconstruction
- The Effect of Under-sampling: Aliasing
- DT Processing of CT Signals
- A/D and D/A Conversion
- Sampling of DT Signals:
  - Up-Sampling (more samples)
  - Down-sampling (less samples)
- Summary

Up & Down Sampling: Applications

- **Sampling-rate conversion**: Given a digital signal, change its sampling rate (i.e., the number of samples per second)
  - image display when original image size differs from the display size
  - converting speech, audio, image, video from one format to another
  - reduce sample rate to reduce the data rate
- Down-sampling: reduce the sampling rate
- Up-Sampling: increase the sampling rate

- Audio CD $\longleftrightarrow$ DVD
- Film $\longleftrightarrow$ TV Signal
- Underwater signal $\longleftrightarrow$ Upsampling

Sampling of DT Signals:
Changing Sampling Rate (Integer Factor)

- A CT signal can be represented by its samples as

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

- Some applications require us to change the sampling rate
  - Or to obtain a new DT representation of the same CT signal of the form $x'[n] = x_c(nT')$ where $T \neq T'$ **←Changing the time axis**

- The problem is to get x'[n] given x[n]

- One way of accomplishing this is to
  - Reconstruct the CT signal from x[n]
  - Re-sample the CT signal using new rate to get x'[n]
  - This requires analog processing which is often undesired

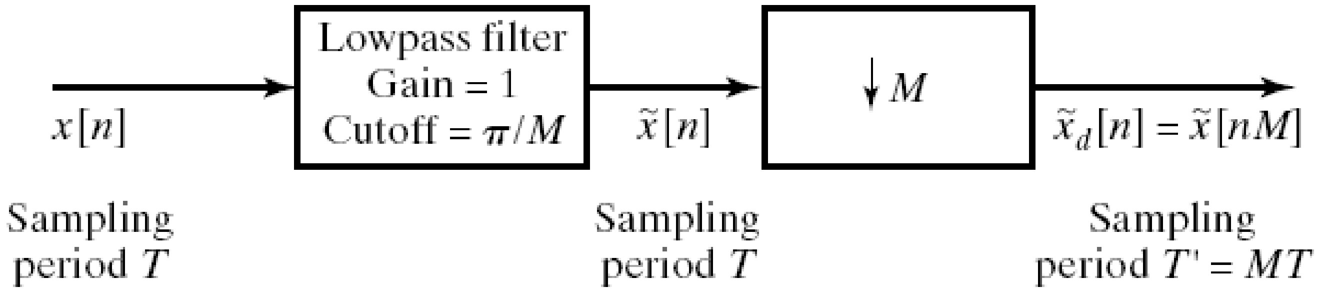
Sampling of DT Signals

Decreasing the Sampling Rate by Integer Factor: Down-Sampling/Decimation

- We reduce the sampling rate of a sequence by "sampling" it

$$x_d[n] = x[nM] = x_c(nMT)$$

- This is accomplished with a sampling rate compressor



- There will be no aliasing if $\Omega'_s = \dfrac{2\pi}{T'} = \dfrac{2\pi}{MT} > 2\Omega_N \implies \Omega'_s = \dfrac{\pi}{MT} > \Omega_N$

- We may obtain $x_d[n]$ by reconstructing *x[n]* and re-sampling with T'=MT

Sampling of Discrete-Time Signals

Frequency domain analysis of Down Sampling

- Recall the DTFT of $x[n]=x_c(nT)$

$$X\left(e^{j\omega}\right)=\frac{1}{T}\sum_{k=-\infty}^{\infty}X_c\left(j\left(\frac{\omega}{T}-\frac{2\pi k}{T}\right)\right)$$

- The DTFT of the down-sampled signal can similarly written as

$$X_d\left(e^{j\omega}\right)=\frac{1}{T'}\sum_{r=-\infty}^{\infty}X_c\left(j\left(\frac{\omega}{T'}-\frac{2\pi r}{T'}\right)\right)=\frac{1}{MT}\sum_{r=-\infty}^{\infty}X_c\left(j\left(\frac{\omega}{MT}-\frac{2\pi r}{MT}\right)\right)$$

- With *r=i+kM*,

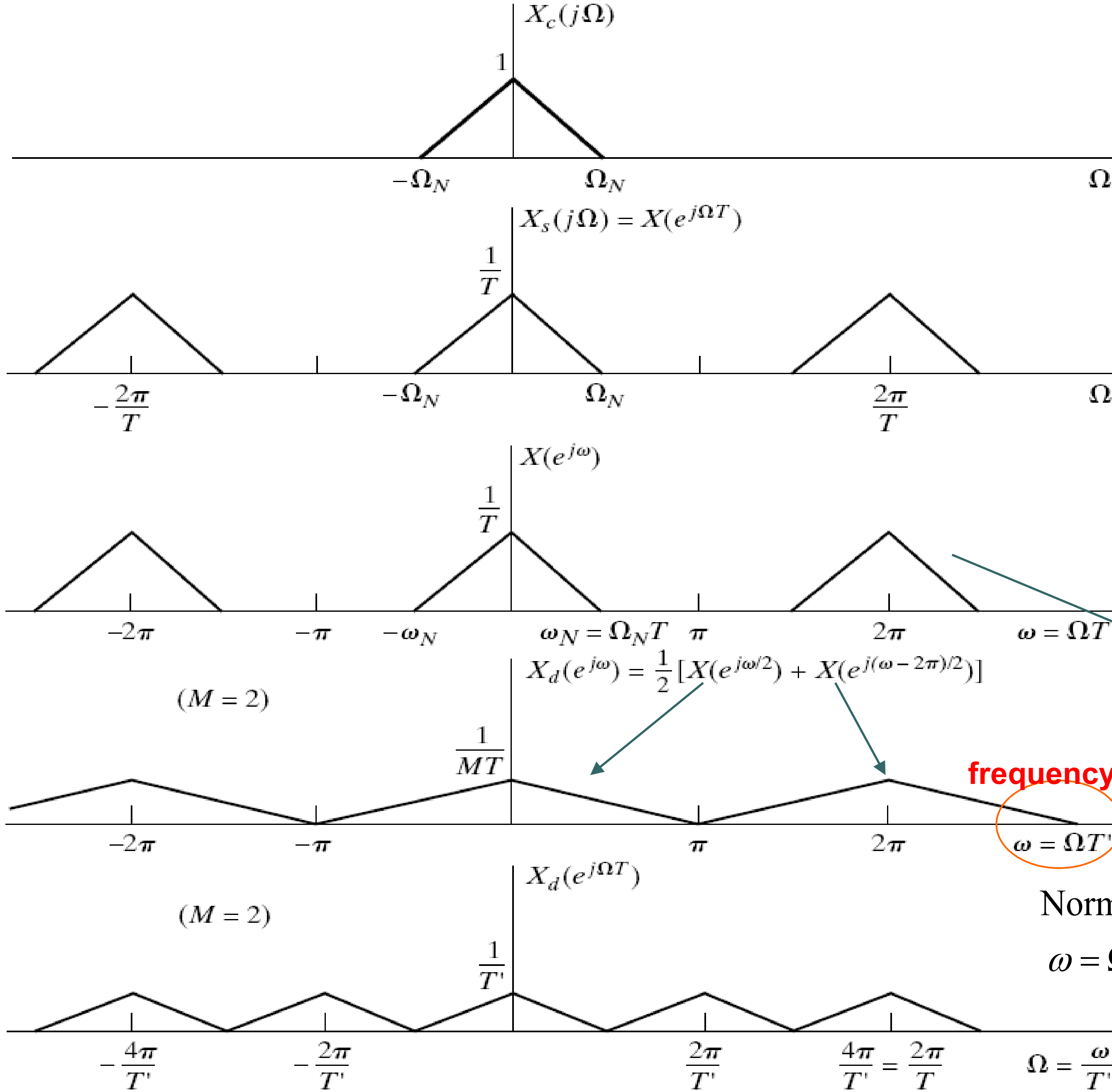
$$X_d\left(e^{j\omega}\right)=\frac{1}{M}\sum_{i=0}^{M-1}\left[\frac{1}{T}\sum_{r=-\infty}^{\infty}X_c\left(j\left(\frac{\omega}{MT}-\frac{2\pi k}{T}-\frac{2\pi i}{MT}\right)\right)\right]$$

- And finally

$$X_d\left(e^{j\omega}\right)=\frac{1}{M}\sum_{i=0}^{M-1}X\left(e^{j\left(\frac{\omega}{M}-\frac{2\pi i}{M}\right)}\right)$$

← M copies of $X\left(e^{j\omega}\right)$, frequency-scaled by M and shifted by multiple of 2π

Frequency domain analysis of Down sampling: No aliasing



Downsampling stretches the DTFT by a factor of M along with the ω axis

frequency-scaled by M

Normalized freq :

$\omega = \Omega T' = \Omega T M$

Down-sampling

$$X_d\left(e^{j\omega}\right)=\frac{1}{M}\sum_{i=0}^{M-1}X\left(e^{j\left(\frac{\omega}{M}-\frac{2\pi i}{M}\right)}\right)$$

$$M=2\therefore\ X_d\left(e^{j\omega}\right)=\frac{1}{2}(X\left(e^{j\left(\frac{\omega}{2}\right)}\right)+X\left(e^{j\left(\frac{\omega}{2}-\pi\right)}\right))$$

- Down-sampling

  1) expands each 2π-periodic repetition of $X\left(e^{j\omega}\right)$ by a factor of M along the ω axis ➔ new period is then M2π

  2) reduces the gain by a factor of M

- If *x[n]* is not band-limited to π/M, aliasing may result from spectral overlap

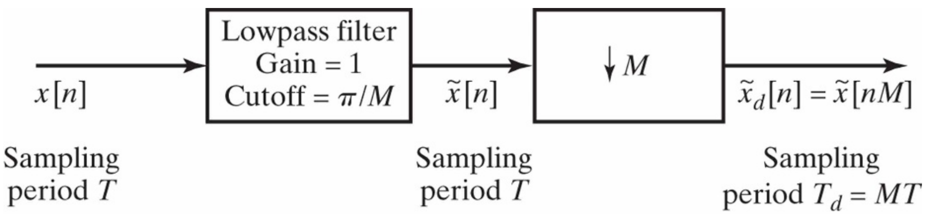
Down-sampling

$$M = 2 \therefore \; X_d\left(e^{j\omega}\right) = \frac{1}{2}\left(X\left(e^{j\left(\frac{\omega}{2}\right)}\right) + X\left(e^{j\left(\frac{\omega}{2}-\pi\right)}\right)\right)$$

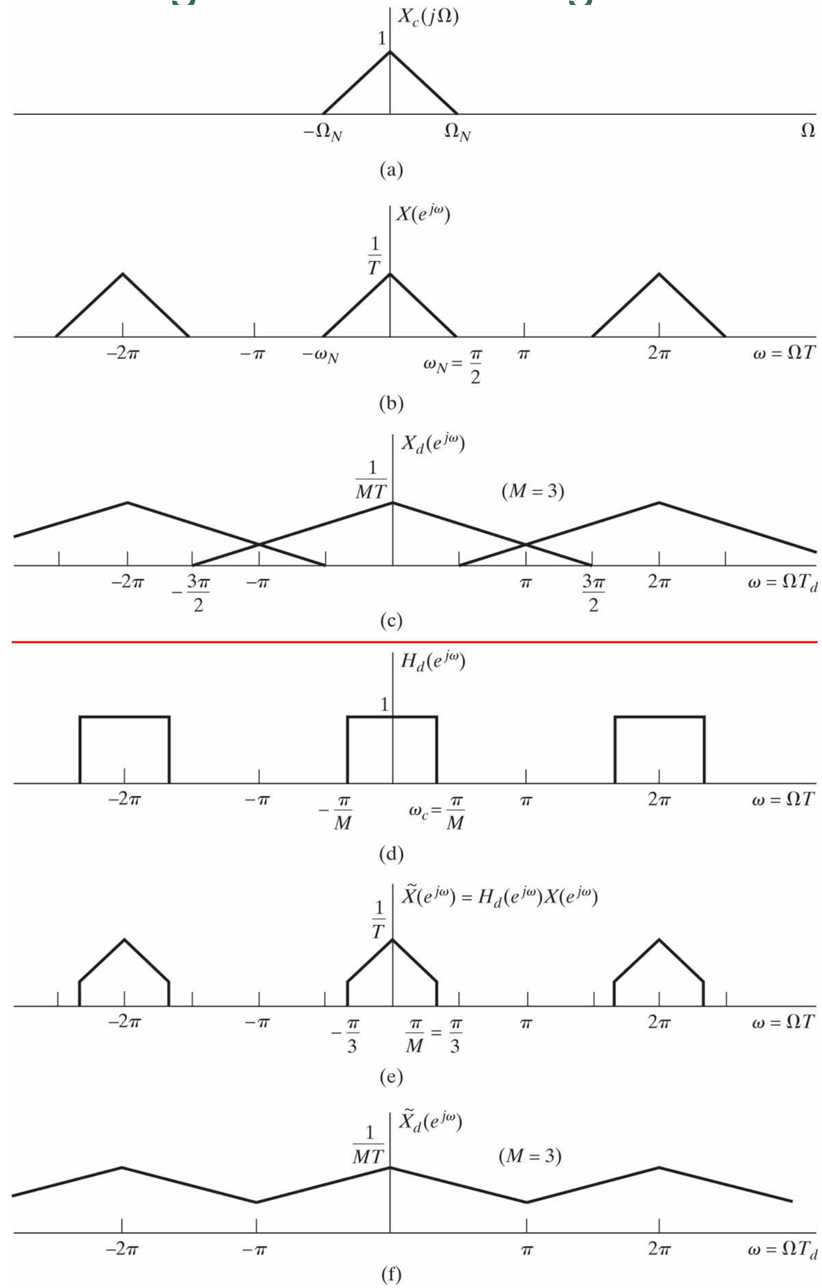
- Interpretation 1: two stages
  1. Create copies and shift by pi
  2. Scale by 2

- Interpretation 2: two stages
  1. Create 2 copies
     (in time domain, sample with 2)
  2. Scale frequency-axis by *2*
     (multiply frequency by 2)

(a)–(c) Downsampling with aliasing
(d)–(f) Downsampling with prefiltering to avoid aliasing



General system for sampling rate reduction by *M*



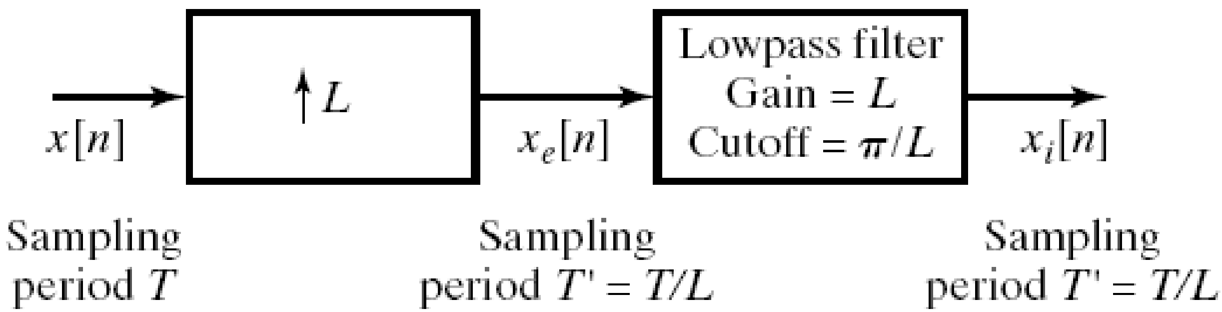
Sampling of DT Signals

Increasing the Sampling Rate by Integer Factor: Up-sampling

- Increase the sampling rate of a sequence by interpolating it

$$x_i[n] = x[n/L] = x_c(nT/L)$$

➔ Sampling rate expander

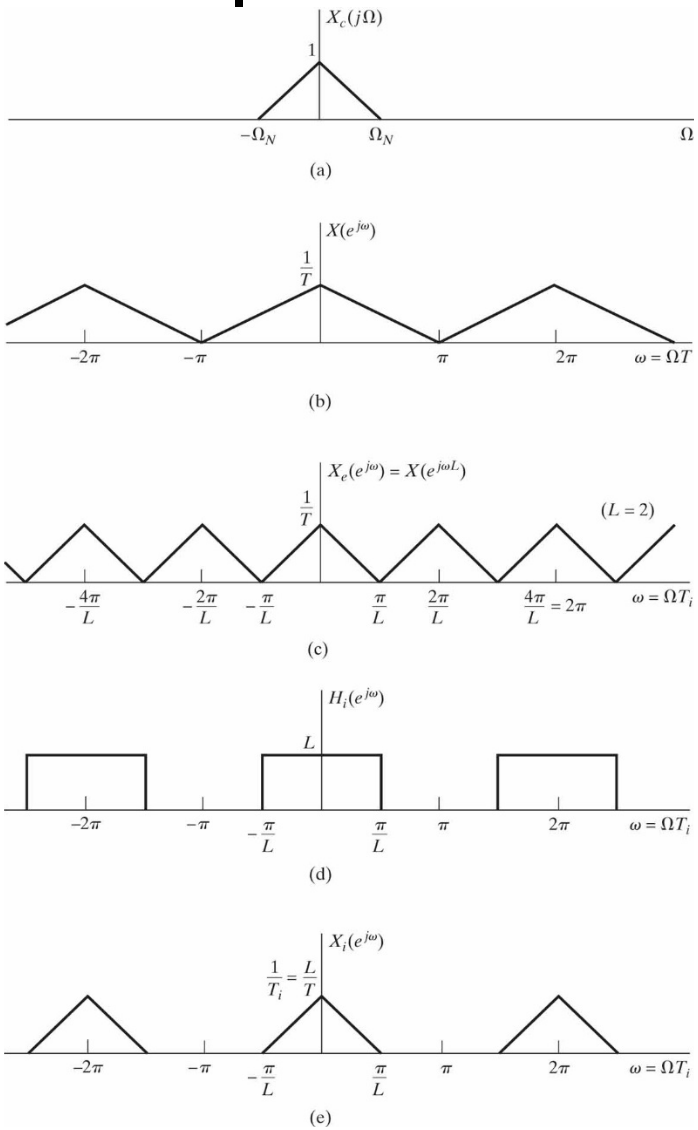


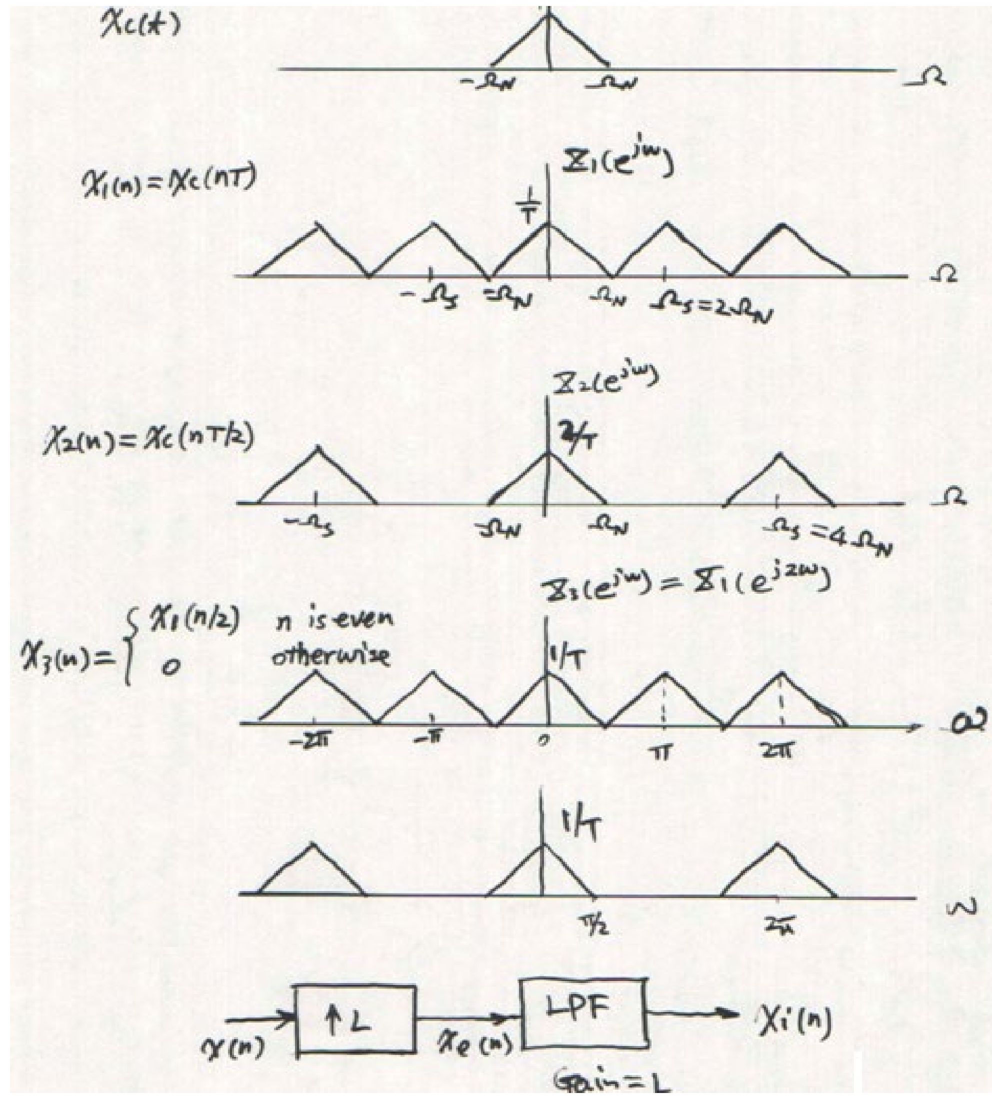
- We obtain $x_i[n]$ that is identical to what we would get by reconstructing the signal and re-sampling it with T'=T/L

- Up sampling consists of two steps: Expanding & Interpolating

$$x_e[n] = \begin{cases} x[n/L] & n = 0, \pm L, \pm 2L, \ldots \\ 0 & else \end{cases} = \sum_{k=-\infty}^{\infty} x[k]\delta[n-kL]$$

Illustration of Upsampling



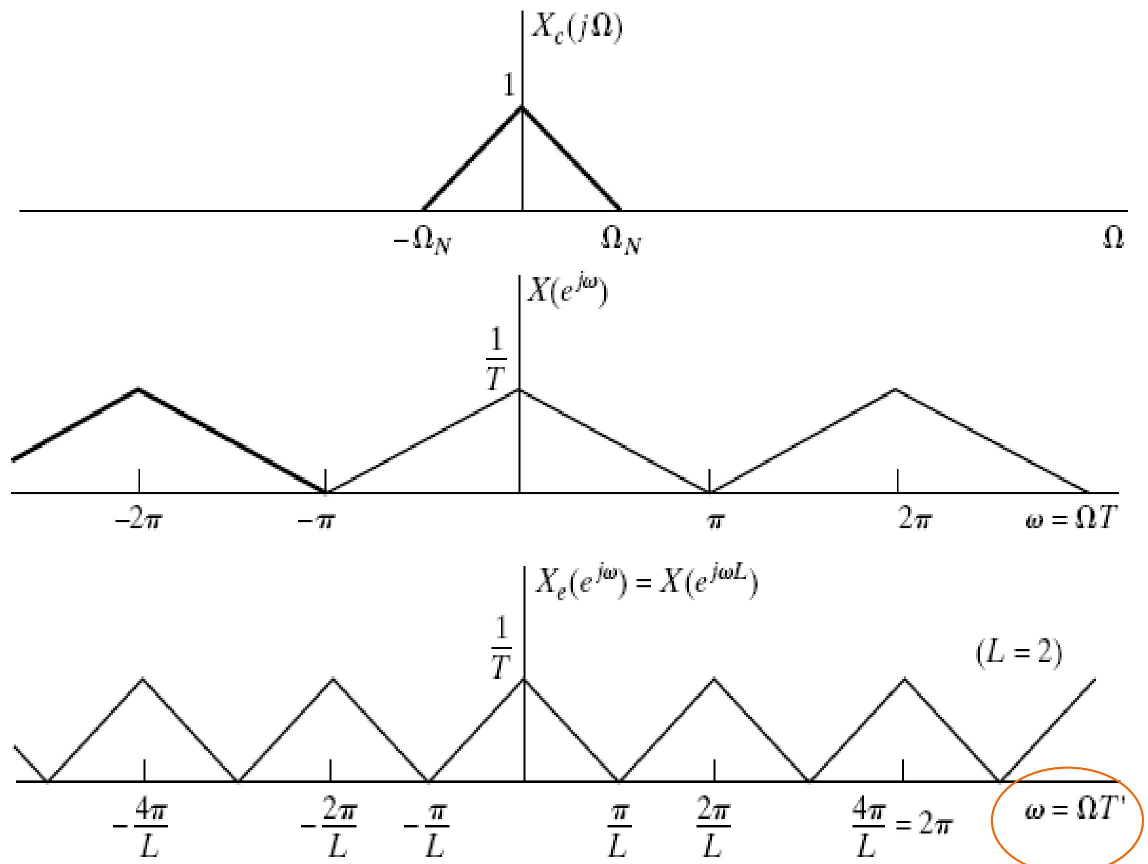


Sampling of DT Signals: Upsampling/Expanding

- The DTFT of $x_e[n]$ can be written as

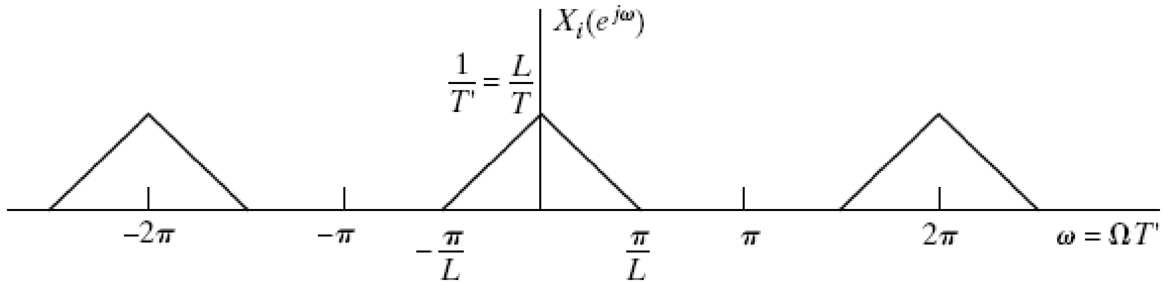
$$X_e\left(e^{j\omega}\right) = \sum_{n=-\infty}^{\infty}\left(\sum_{k=-\infty}^{\infty} x[k]\delta[n-kL]\right)e^{-j\omega n} = \sum_{k=-\infty}^{\infty} x[k]e^{-j\omega Lk} = X\left(e^{j\omega L}\right)$$

- The output of the expander is frequency-scaled

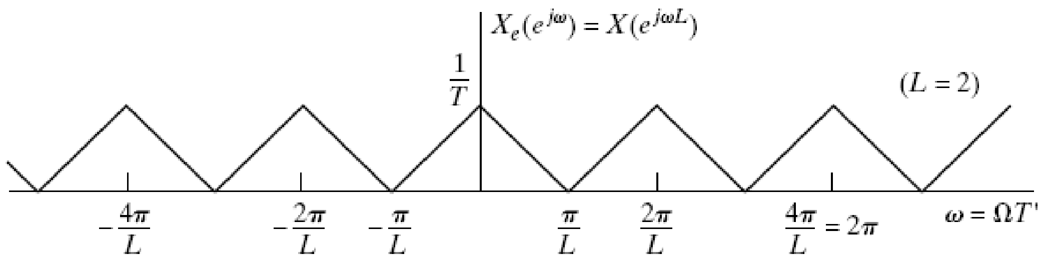


Interpolating sampled DT signals

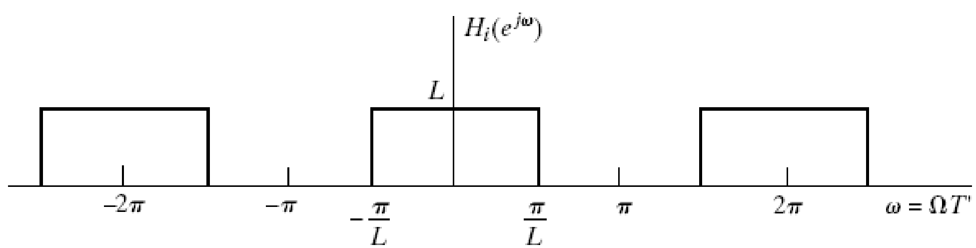
- The DTFT of the desired interpolated signals is



- The extrapolator output is given as



- To get interpolated signal we apply the following ideal LPF



Interpolator in Time Domain

- $x_i[n]$ in a low-pass filtered version of $x[n]$
- The low-pass filter impulse response is $$h_i[n] = \frac{\sin(\pi n / L)}{\pi n / L}$$
- Hence the interpolated signal is written as

$$x_i[n] = \sum_{k=-\infty}^{\infty} x[k] \frac{\sin(\pi(n-kL)/L)}{\pi(n-kL)/L}$$

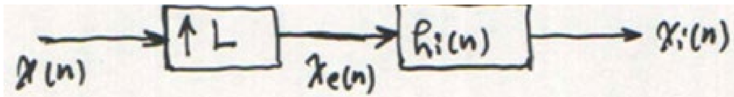
- Note that $h_i[0] = 1$

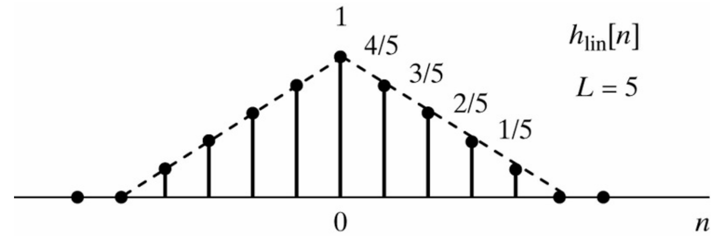
  $h_i[n] = 0 \quad \mathrm{n} = \pm\mathrm{L}, \pm 2\mathrm{L}, \ldots$

➔ the filter output can be written as

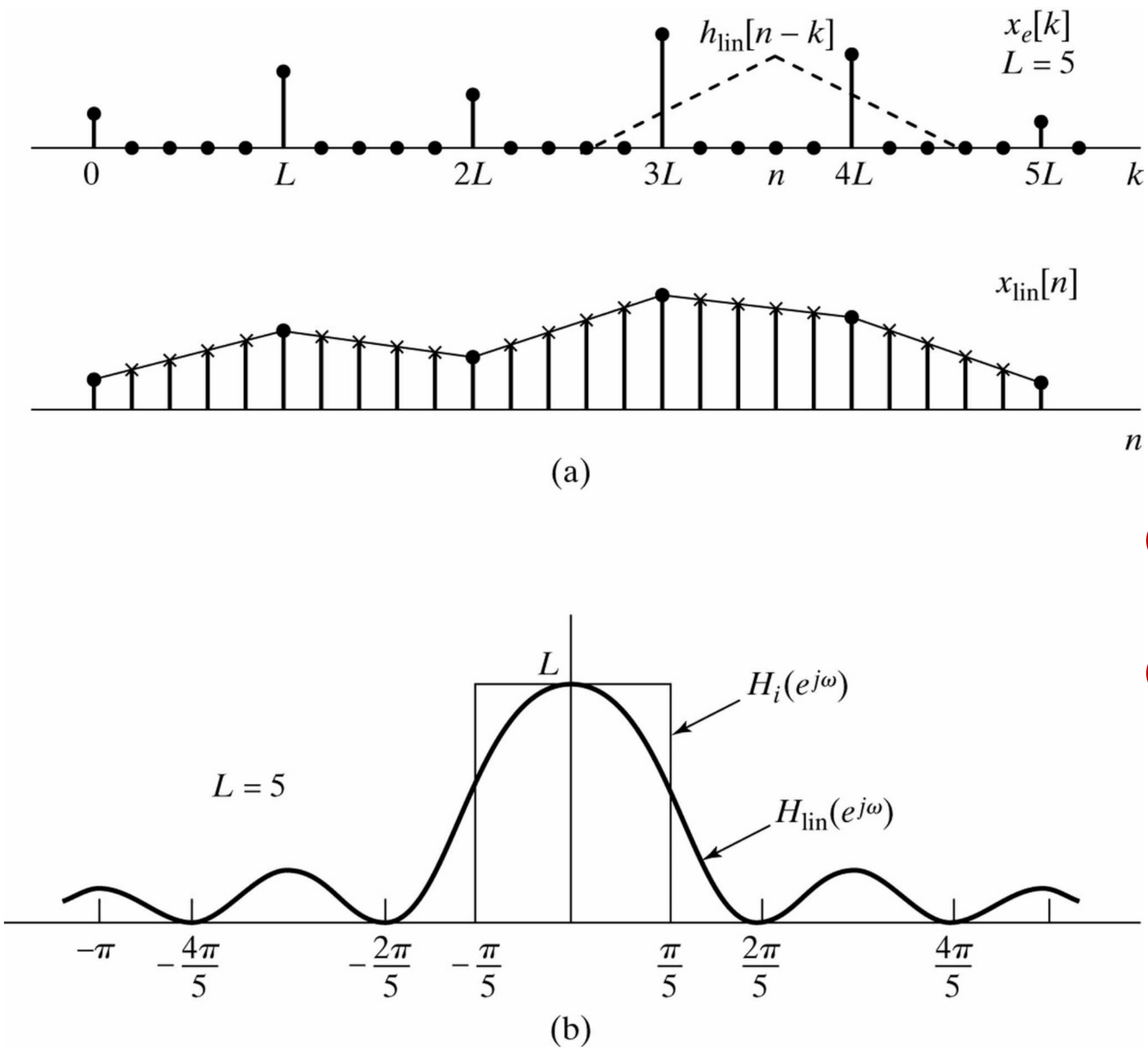
$$x_i[n] = x[n/L] = x_c(nT/L) = x_c(nT') \qquad \text{for n} = 0, \pm\mathrm{L}, \pm 2\mathrm{L}, \ldots$$

Linear interpolation





Impulse response for linear interpolation

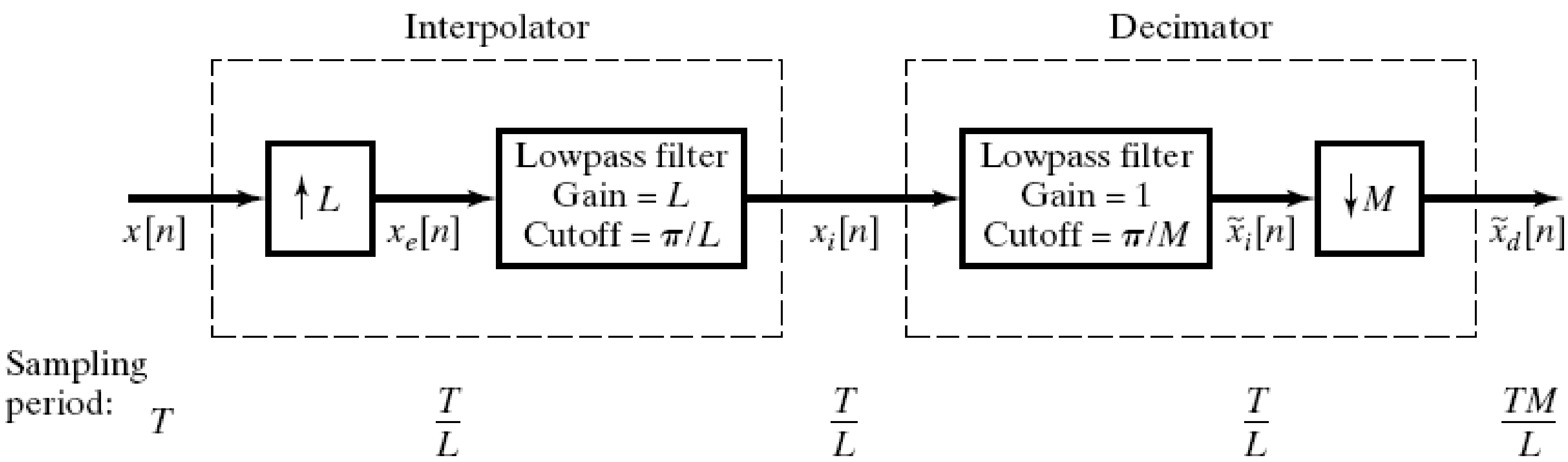


(a) Illustration of linear interpolation by filtering

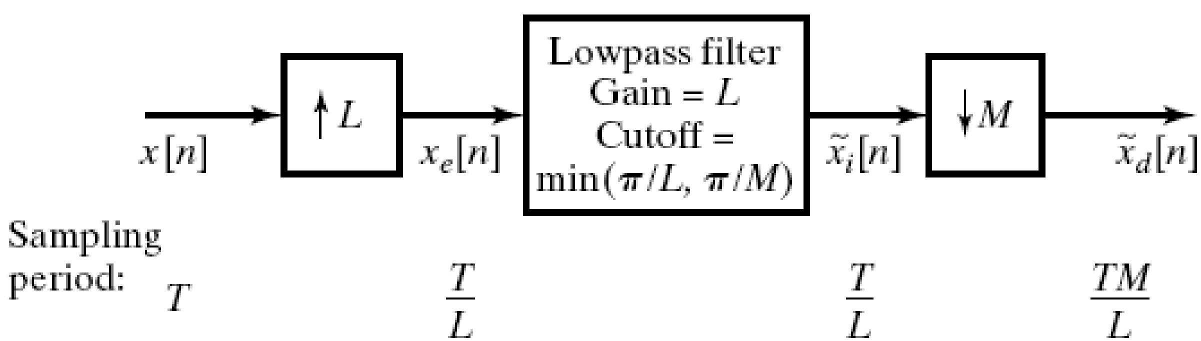
(b) Frequency response of linear interpolator compared with ideal lowpass interpolation filter

Multi-rate processing:
Combine decimation and interpolation

Changing the Sampling Rate by <u>Non-Integer Factor:</u>



- Since both interpolation and anti-aliasing filters are low-pass filters, the filter with the smallest bandwidth is more restrictive and can therefore be used in place of both filters



Sampling of DT Signals:

Changing the Sampling Rate by Non-Integer Factor

- If M>L: net increase of sampling period (or decrease in sampling frequency)

  ➔net operation is downsampling

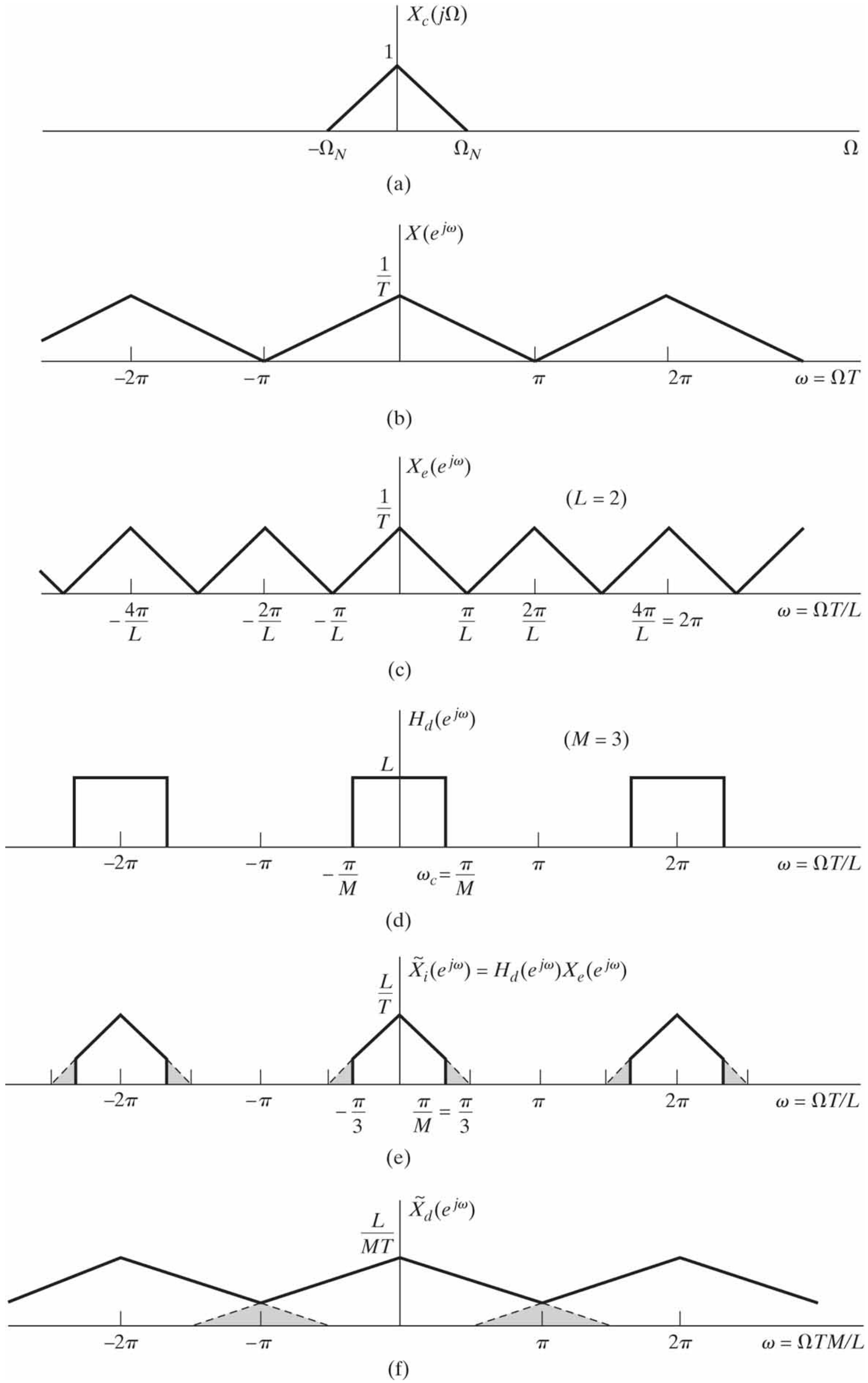
  ➔ π / M is the dominant cutoff frequency & the low-pass filter should have cutoff at π / M
- If $M<L$ **and** T respects Nyquist theorem

  ➔ π / L is the dominant cutoff frequency

  ➔no need to further limit the bandwidth of the signal below Nyquist frequency

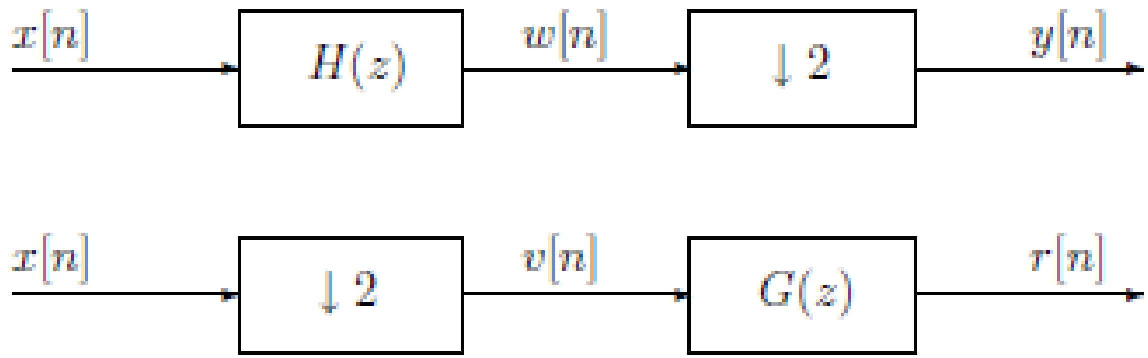
- Interpolation and downsampling are not reversible, due to loss of data

Example: changing the rate by 2/3



Example

Consider the following system



Find G(z) such that y[n] = r[n].

Outline

- Introduction
- Sampling
- Reconstruction
- The Effect of Under-sampling: Aliasing
- DT Processing of CT Signals
- A/D and D/A Conversion
- Sampling of DT Signals
- Summary

Summary: to know ..

- How the sampling is derived using the FT ..
- Lowpass filters for reconstruction ..
- The sampled signal spectrum contains the original spectrum and its replicas (aliases) at $k\Omega_s$, k=+/- 1,2,….
- We need a prefilter when sampling a signal
  - To avoid aliasing
  - The filter should be a lowpass filter with cutoff frequency at *fs* /2
- Sample-and-hold and linear interpolation
- Why the ideal interpolation filter is a lowpass filter with cutoff frequency at *fs/2*
- The ideal interpolation kernel is the *sinc* function.
- Why to apply a pre-filter before sampling
- ….

Summary

- The information carried by a signal can be defined either in terms of its *Time Domain* pattern or its *Frequency Domain* spectrum

- The amount of information in a CT (analog) signal x(t) can be specified by a finite number of values: samples x[n]

- The *Sampling Theorem* states that we can collect all the information in a signal by sampling at a rate $2\Omega_N$, where $2\Omega_N$ is the bandwidth of x(t)
  - We can, therefore, reconstruct the actual shape of the original CT signal at any instant 'in between' the sampled instants
  - This reconstruction is not a guess but a true reconstruction

Summary

- The FT of a DT signal is a function of the continuous variable ω, and it is periodic with period 2π
  - Given a value of ω, the FT gives back a complex number that can be interpreted as magnitude and phase (translation in time) of the sinusoidal component at that frequency ω

- Sampling is a multiplication with a periodic impulse train
- FT of sampled signal: original FT + shifted versions at multiples of Ω_s
- Sampling the CT signal x(t) with interval T, we get the DT signal $x[n]=x(nT)$ which is a function of the discrete variable n

- Sampling a CT signal with sampling rate fs produces a DT signal whose FT is the periodic replication of the original signal, and the replication period is Ts
  - The Fourier variable ω for functions of discrete variable is converted into the frequency variable f (in Hertz) by means of $f=\omega/(2\pi T)$

Summary

- A/D converters convert CT signals into sequences with discrete sample values
  - Operates with the use of sampling and quantization
- D/A converters convert sequences with discrete sample values into CT signals
  - Analyzed as conversion to impulse train followed by reconstruction filtering
- Zero-order hold is a simple but low performance filter

- Upsampling and downsampling allow for changes in the effective sample rate of sequences
  - Allows matching of sample rates of A/D, D/A, and digital processor
  - Analysis: downsampler/upsampler similar to A/D, D/A
- When performing a frequency-domain analysis of systems with up/downsamplers, it is strongly recommended to carry out the analysis in the z-domain until the last step
  - Working directly in the ω domain can easily lead to errors