Chapter 1: Introduction to Digital Signal Processing

Course Objectives

1. Introduce discrete-time signals and systems representation and analysis

2. Introduce design methods and realization structures of discrete-time systems

3. Introduce the application of digital signal processing

1. Introduction

- **Digital signal processing (DSP)** is concerned with the representation of signals in **digital form** and with the processing of these signals and the information that they carry.
- **DSP**, began to flourish in the 1960's, some of the important and **powerful processing techniques** that are in use today may be traced back to numerical algorithms that were proposed and studied centuries ago.
- Since the early **1970's**, when the first DSP chips were introduced, the **field of digital signal processing has evolved dramatically.**
- **DSP** has become an integral part of many **commercial products** and **applications**, and is becoming a common place term.

1.1 Sampling

- Most discrete-time signals come from **sampling** a continuous-time signal.
- Such as speech and audio signals, radar and sonar data, and seismic and biological signals.
- The process of converting these signals into digital form is called analog-to-digital (A/D) conversion.
- The reverse process of reconstructing an analog signal from its samples is known as digital-to-analog (D/A) conversion.

1.2 Analog to Digital Conversion (A/D)

- An A/D converter transforms an analog signal into a digital sequence. The input to the **A/D** converter, x_a(t), is a **real-valued** function of a **continuous variable**, **t**.
- For each value of **t**, the function x_a(t) may be any real number.
- The output of the A/D is a bit stream that corresponds to a **discrete time sequence, x(n),** with an amplitude that is quantized, for each value of **n**, to one of a **finite number** of possible values.
- The components of an A/D converter are shown in Fig. 1.1. The first is the **sampler**, which is sometimes referred to as a **continuous- to-discrete (C/D) converter, or ideal A/D converter**.

- The **sampler** converts the continuous-time signal x_a(t) into a discrete-time sequence x(n) by extracting the values of x_a(t) at integer multiples of the sampling period, T_s.

$$x(n) = x_a(nT_s)$$

- The samples $x_a(nT_s)$ have a continuous range of possible amplitudes.
- The **second** component of the A/D converter is the **quantizer,** which **maps** the continuous amplitude into a discrete set of amplitudes.
- For a uniform quantizer, the quantization process is defined by the number of **bits** and the quantization interval **Δ**.
- The last component is the **encoder**, which takes the **digital signal** **x'(n)** and produces a sequence of **binary code words**.

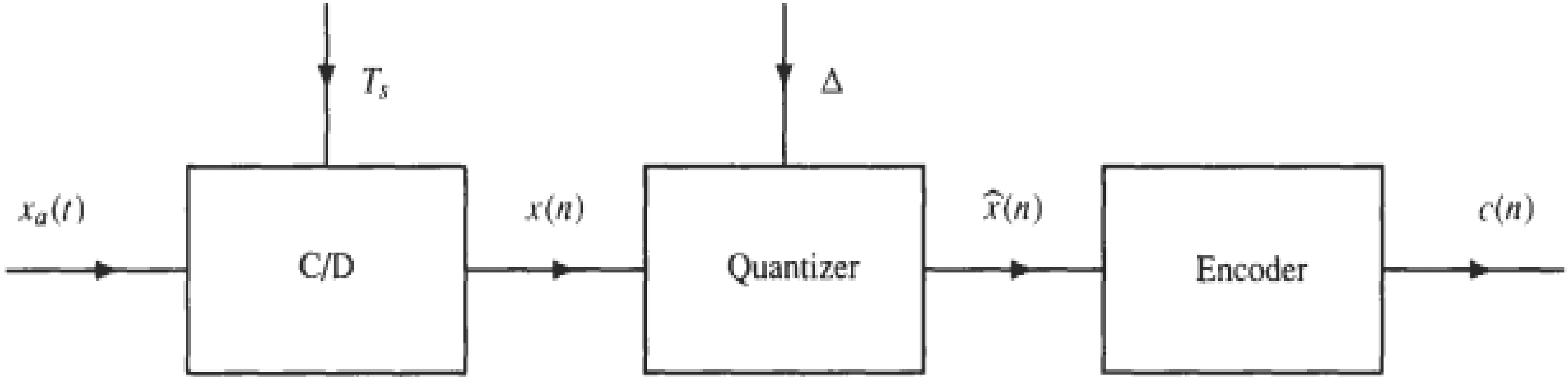


Fig. 1.1 The components of an analog-to-digital converter

1.2.1 Periodic Sampling

- Typically, discrete-time signals are formed by **periodically** sampling a continuous-time signal

$$x(n) = x_a(nT_s)$$

- The **sample spacing** T_s, is the sampling period, and $f_s=1/T_s$, is the **sampling frequency** in samples per second. This sampling process is illustrated in Fig. 1.2(a). First, the continuous-time signal is **multiplied** by a periodic sequence of **impulses.**

$$s_a(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT_s)$$

To form the sampled signal

$$x_s(t) = x_a(t)s_a(t) = \sum_{n=-\infty}^{\infty} x_a(nT_s)\delta(t - nT_s)$$

Sampled signal is converted into a discrete-time signal by **mapping** the **impulses** that are spaced in time by T_s into a sequence **x(n)** where the sample values are indexed by the integer variable **n**:

$$x(n) = x_a(nT_s)$$

This process is illustrated in Fig. 1.2(b).

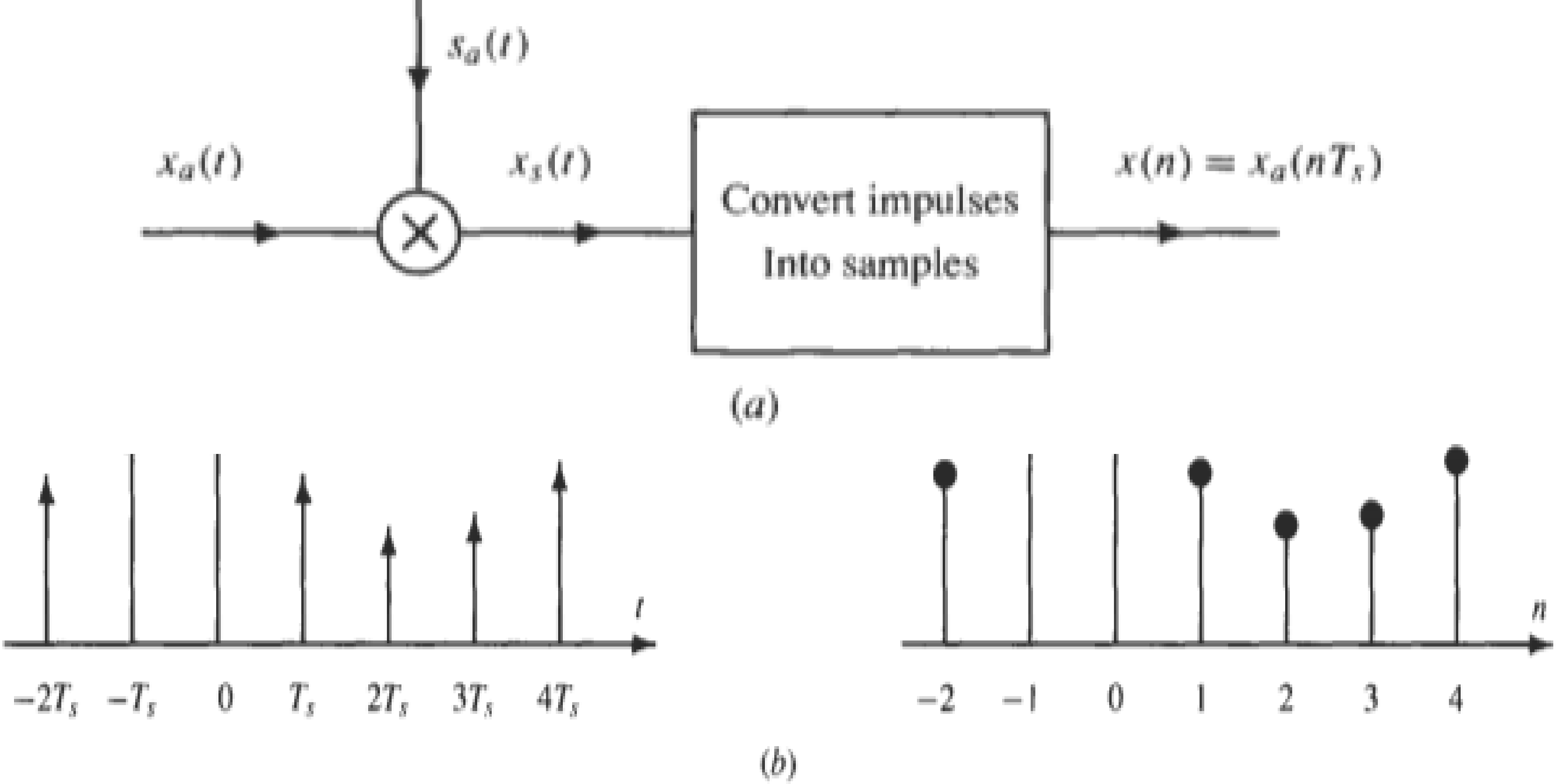


Fig. 1.2. Continuous-to discrete conversion. (a) A model that consists of multiplying x_a(t) by a sequence of impulses. followed by a system that converts impulses into samples. (b) An example that illustrates the conversion process.

The effect of the C/D converter may be analyzed in the frequency domain as follows. Because the Fourier transform of $\delta(t - nT_s)$ is $e^{-jn\Omega T_s}$, the Fourier transform of the sampled signal $x_s(t)$ is

$$X_s(j\Omega) = \sum_{n=-\infty}^{\infty} x_a(nT_s)e^{-jn\Omega T_s}$$

Another expression for $X_s(j\Omega)$ follows by noting that the Fourier transform of $s_a(t)$ is

$$S_a(j\Omega) = \frac{2\pi}{T_s} \sum_{k=-\infty}^{\infty} \delta(\Omega - k\Omega_s)$$

where $\Omega_s = 2\pi/T_s$ is the sampling frequency in radians per second. Therefore,

$$X_s(j\Omega) = \frac{1}{2\pi} X_a(j\Omega) * S_a(j\Omega) = \frac{1}{T_s} \sum_{k=-\infty}^{\infty} X_a(j\Omega - jk\Omega_s)$$

Finally, the discrete-time Fourier transform of $x(n)$ is

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x(n)e^{-jn\omega} = \sum_{n=-\infty}^{\infty} x_a(nT_s)e^{-jn\omega}$$

$$X(e^{j\omega}) = X_s(j\Omega)|_{\Omega=\omega/T_s} = \frac{1}{T_s}\sum_{k=-\infty}^{\infty} X_a\left(j\frac{\omega}{T_s} - j\frac{2\pi k}{T_s}\right)$$

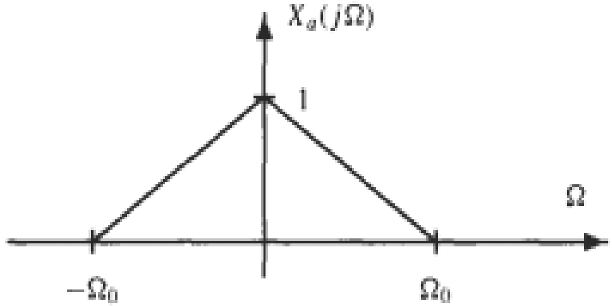
Thus, $X(e^{j\omega})$ is a frequency-scaled version of $X_s(j\Omega)$, with the scaling defined by

$$\omega = \Omega T_s$$

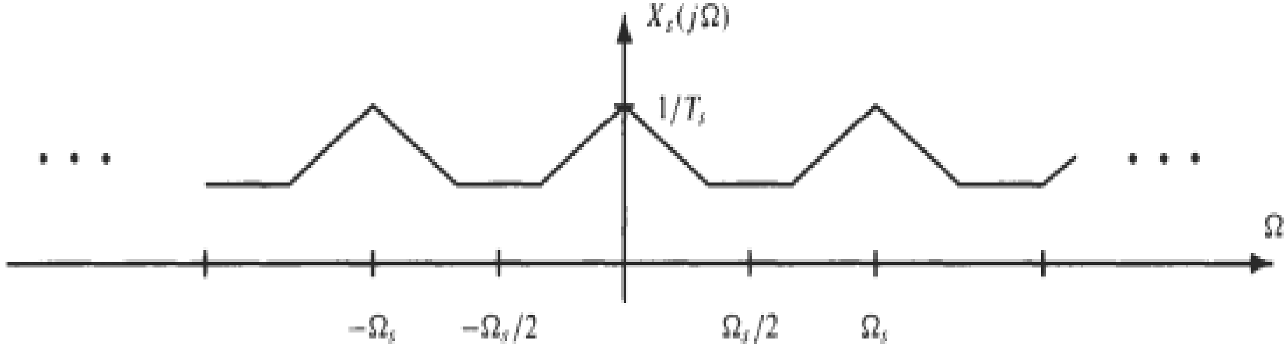
This scaling, which makes $X(e^{j\omega})$ periodic with a period of 2π, is a consequence of the time-scaling that occurs when $x_s(t)$ is converted to $x(n)$.

Example 1.2.1

Suppose that $x_a(t)$ is strictly bandlimited so that $X_a(j\Omega) = 0$ for $|\Omega| > \Omega_0$ as shown in the figure below.



If $x_a(t)$ is sampled with a sampling frequency $\Omega_s \geq 2\Omega_0$, the Fourier transform of $x_s(t)$ is formed by periodically replicating $X_a(j\Omega)$ as illustrated in the figure below.



This overlapping of spectral components is called *aliasing*. When aliasing occurs, the frequency content of $x_a(t)$ is corrupted, and $X_a(j\Omega)$ cannot be recovered from $X_s(j\Omega)$.

- **As illustrated in Example 1.2.1, if x_a(t) is strictly band limited so that the highest frequency in x_a(t) is Ωo, and if the sampling frequency is greater than 2Ωo**

$$\Omega_s \geq 2\Omega_0$$

no aliasing occurs, and $x_a(t)$ may be uniquely recovered from its samples $x_a(nT_s)$ with a low-pass filter. The following is a statement of the famous Nyquist sampling theorem:

Sampling Theorem: If $x_a(t)$ is strictly bandlimited,

$$X_a(j\Omega) = 0 \qquad |\Omega| > \Omega_0$$

then $x_a(t)$ may be uniquely recovered from its samples $x_a(nT_s)$ if

$$\Omega_s = \frac{2\pi}{T_s} \geq 2\Omega_0$$

The frequency Ω_0 is called the *Nyquist frequency*, and the minimum sampling frequency, $\Omega_s = 2\Omega_0$, is called the *Nyquist rate*.

- Signals that are found in physical systems will never be strictly **band limited**.
- An analog anti- aliasing filter is typically used to filter the signal prior to sampling in order to **minimize** the amount of **energy** above the Nyquist frequency.
- To **reduce** the amount of aliasing that occurs in the A/D converter.

1.2.2 Quantization and Encoding

- **A quantizer** is a **nonlinear** and **noninvertible** system that transforms an input sequence x(n).
- That has a continuous range of amplitudes into a sequence for which each value of **x(n)** assumes one of a finite number of possible values. This operation is denoted by

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

The quantizer has $L+1$ *decision levels* $x_1, x_2, \ldots, x_{L+1}$ that divide the amplitude range for $x(n)$ into L *intervals*

$$I_k = [x_k, x_{k+1}] \qquad k = 1, 2, \ldots, L$$

For an input $x(n)$ that falls within interval I_k, the quantizer assigns a value within this interval, $\hat{x}_k$, to $x(n)$.

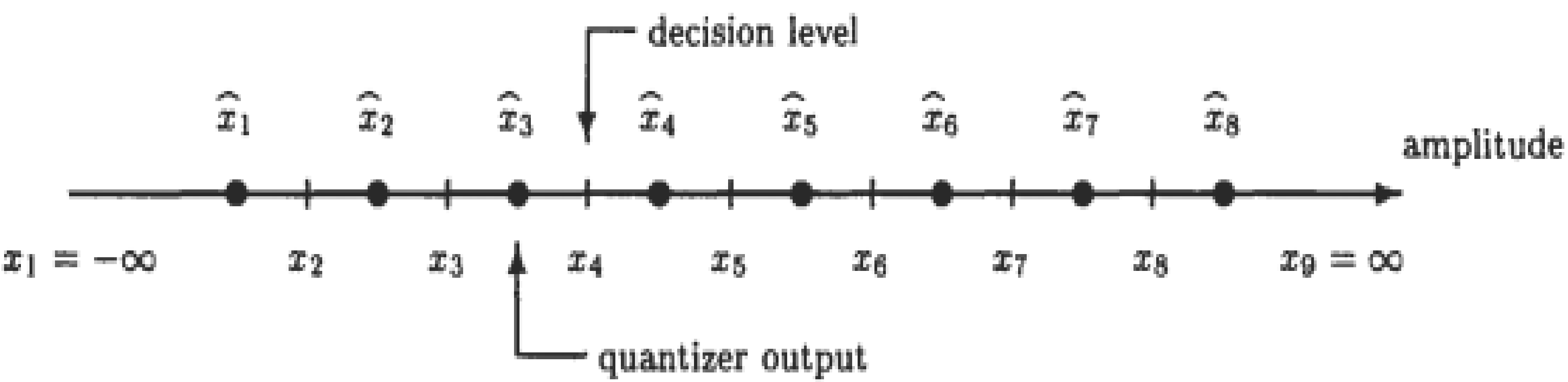


Fig. 1.3. A quantizer with nine decision levels that divide the input amplitudes into eight quantization intervals and eight possible quantizer outputs. X'k.

Quantizers may have quantization levels that are either **uniformly** or **non uniformly** spaced. When the quantization intervals are uniformly spaced.

$$\Delta = x_{k+1} - x_k$$

- Δ is called the **quantization step size** or the **resolution** of the quantizer.
- **Quantizer** is said to be a uniform or linear quantizer, the number of levels in a quantizer is generally of the form

$$L = 2^{B+1}$$

- In order to make the most efficient use of a (B + 1) bit binary code word.

- A 3-bit uniform quantizer in which the quantizer output is rounded to the nearest quantization level is illustrated in Fig. 1-4.
- With L $=2^{B+1}$ quantization levels and a step size Δ, the range of the quantizer is

$$R = 2^{B+1} \cdot \Delta$$

Therefore, if the quantizer input is bounded,

$$|x(n)| \leq X_{\max}$$

the range of possible input values may be covered with a step size

$$\Delta = \frac{X_{\max}}{2^{B}}$$

With rounding, the quantization error

$$e(n) = Q[x(n)] - x(n)$$

will be bounded by

$$-\frac{\Delta}{2} < e(n) < \frac{\Delta}{2}$$

However, if $|x(n)|$ exceeds $X_{\max}$, then $x(n)$ will be *clipped*, and the quantization error could be very large.

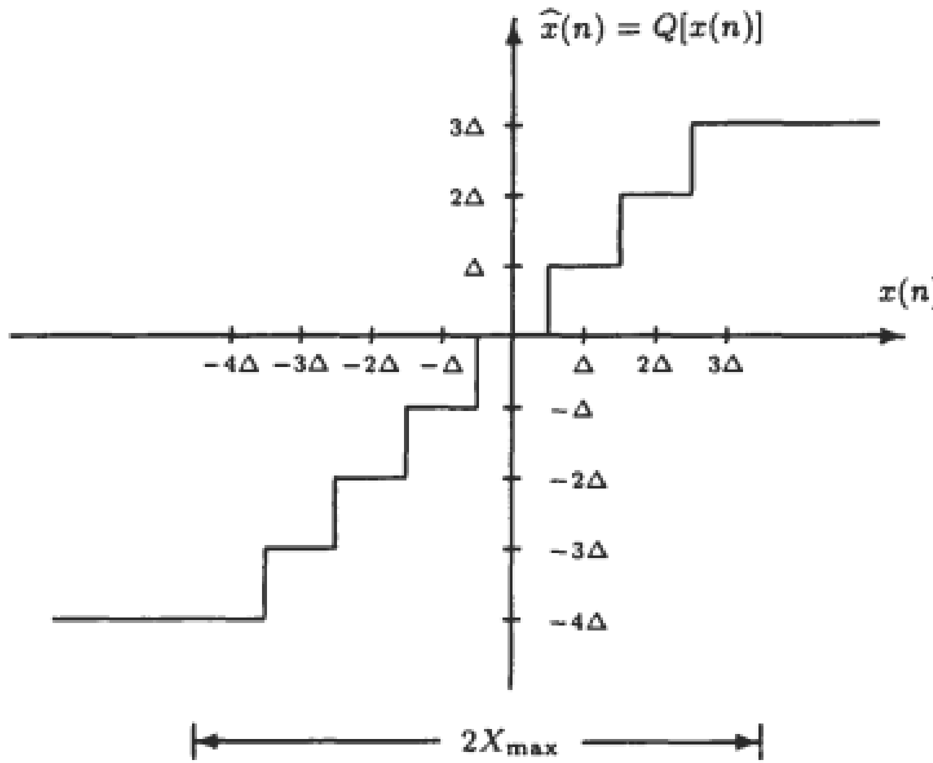


Fig. 1-4. A 3-bit uniform quantizer

- A useful model for the quantization process is given in Fig. 1-5. Here, the quantization error is assumed to be an additive noise source.
- Because the quantization error is typically not known, the quantization error is described statistically.
- It is generally assumed that e(n) is a sequence of random variables where

1. The statistics of e(n) do not change with time (the quantization noise is a stationary random process).
2. The quantization noise e(n) is a sequence of uncorrelated random variables.
3. The quantization noise e(n) is uncorrelated with the quantizer input x(n).
4. The probability density function of e(n) is uniformly distributed over the range of values of the quantization error.

- It is easy to find cases in which these assumptions do not hold (e.g., if x(n) is a constant), they are generally valid for rapidly varying signals with fine quantization (Δ small).

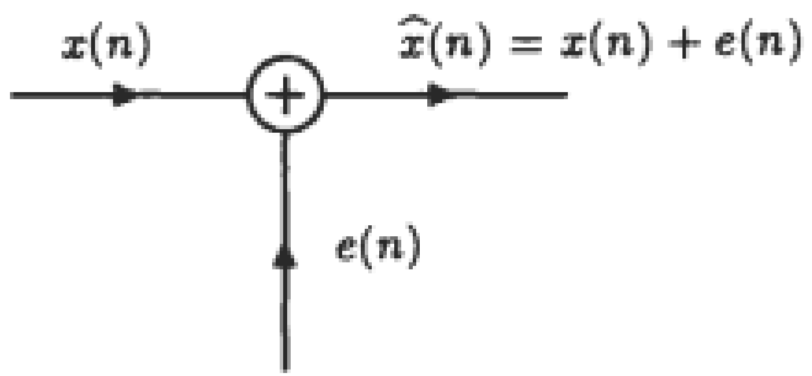


Fig. 1-5. A quantization noise model

- **With rounding, the quantization noise is uniformly distributed over the interval [-Δ/2, Δ/2], and the quantization noise power (the variance) is**

$$\sigma_e^2 = \frac{\Delta^2}{12}$$

With a step size

$$\Delta = \frac{X_{\max}}{2^B}$$

and a signal power σ_x^2, the signal-to-quantization noise ratio, in decibels (dB), is

$$\text{SQNR} = 10 \log \frac{\sigma_x^2}{\sigma_e^2} = 6.02B + 10.81 - 20 \log \frac{X_{\max}}{\sigma_x}$$

- **Thus, the signal-to-quantization noise ratio increases approximately 6 dB for each bit. The output of the quantizer is sent to an encoder, which assigns a unique binary number (codeword) to each quantization level.**
- **Any assignment of codewords to levels may be used, and many coding schemes exist.**
- **Most digital signal processing systems use the two's-complement representation. In this system, with a (B + 1) bit codeword,**

1.3 Digital to Analog Conversion

As stated in the sampling theorem, if $x_a(t)$ is strictly bandlimited so that $X_a(j\Omega) = 0$ for $|\Omega| > \Omega_0$, and if $T_s < \pi/\Omega_0$, then $x_a(t)$ may be uniquely reconstructed from its samples $x(n) = x_a(nT_s)$. The reconstruction process involves two steps, as illustrated in Fig. 1-6. First, the samples $x(n)$ are converted into a sequence of impulses,

$$x_s(t) = \sum_{n=-\infty}^{\infty} x(n)\delta(t - nT_s)$$

and then $x_a(t)$ is filtered with a *reconstruction filter*, which is an ideal low-pass filter that has a frequency response given by

$$H_r(j\Omega) = \begin{cases} T_s & |\Omega| \le \dfrac{\pi}{T_s} \\ 0 & |\Omega| > \dfrac{\pi}{T_s} \end{cases}$$

This system is called an *ideal discrete-to-continuous* (D/C) converter. Because the impulse response of the reconstruction filter is

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

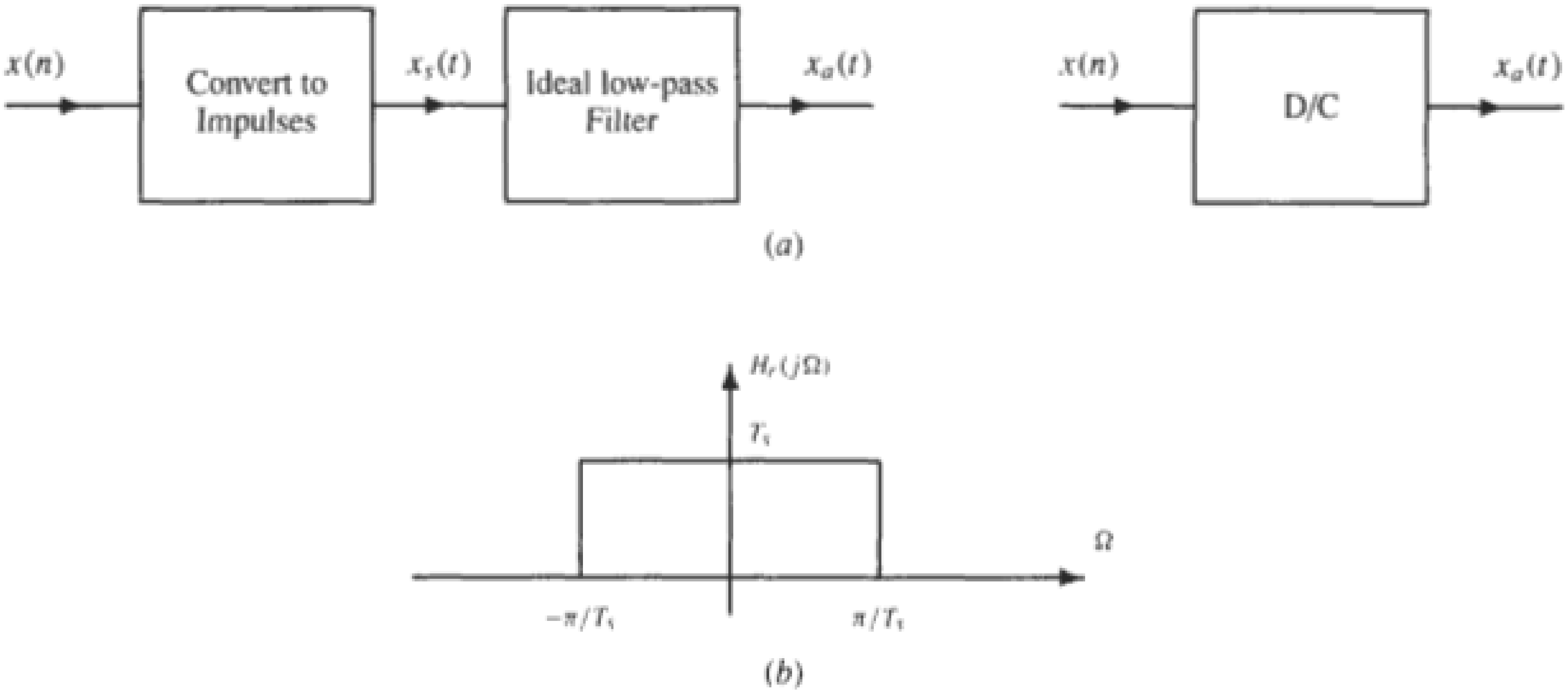


Fig. 1-6. (a) A discrete-to-continuous converter with an ideal low-pass reconstruction filter. (h) The frequency response of the ideal reconstruction filter.

the output of the filter is

$$x_a(t) = \sum_{n=-\infty}^{\infty} x(n)h_r(t - nT_s) = \sum_{n=-\infty}^{\infty} x(n)\frac{\sin \pi(t - nT_s)/T_s}{\pi(t - nT_s)/T_s}$$

This *interpolation formula* shows how $x_a(t)$ is reconstructed from its samples $x(n) = x_a(nT_s)$. In the frequency domain, the interpolation formula becomes

$$X_a(j\Omega) = \sum_{n=-\infty}^{\infty} x(n)H_r(j\Omega)e^{-jn\Omega T_s}$$

$$= H_r(j\Omega) \sum_{n=-\infty}^{\infty} x(n)e^{-jn\Omega T_s} = H_r(j\Omega)X(e^{j\Omega T_s})$$

which is equivalent to

$$X_a(j\Omega) = \begin{cases} T_s X(e^{j\Omega T_s}) & |\Omega| < \dfrac{\pi}{T_s} \\ 0 & \text{otherwise} \end{cases}$$

Thus, $X(e^{j\omega})$ is frequency scaled ($\omega = \Omega T_s$), and then the low-pass filter removes all frequencies in the periodic spectrum $X(e^{j\Omega T_s})$ above the cutoff frequency $\Omega_c = \pi/T_s$.

Because it is not possible to implement an ideal low-pass filter, many D/A converters use a *zero-order hold* for the reconstruction filter. The impulse response of a zero-order hold is

$$h_0(t) = \begin{cases} 1 & 0 \le t \le T_s \\ 0 & \text{otherwise} \end{cases}$$

and the frequency response is

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

After a sequence of samples $x_a(nT_s)$ has been converted to impulses, the zero-order hold produces the staircase approximation to $x_a(t)$ shown in Fig. 1-7. With a zero-order hold, it is common to postprocess the output with a *reconstruction compensation filter* that approximates the frequency response

$$H_c(j\Omega) = \begin{cases} \dfrac{\Omega T_s/2}{\sin(\Omega T_s/2)} e^{j\Omega T_s/2} & |\Omega| \le \dfrac{\pi}{T_s} \\ 0 & |\Omega| > \dfrac{\pi}{T_s} \end{cases}$$

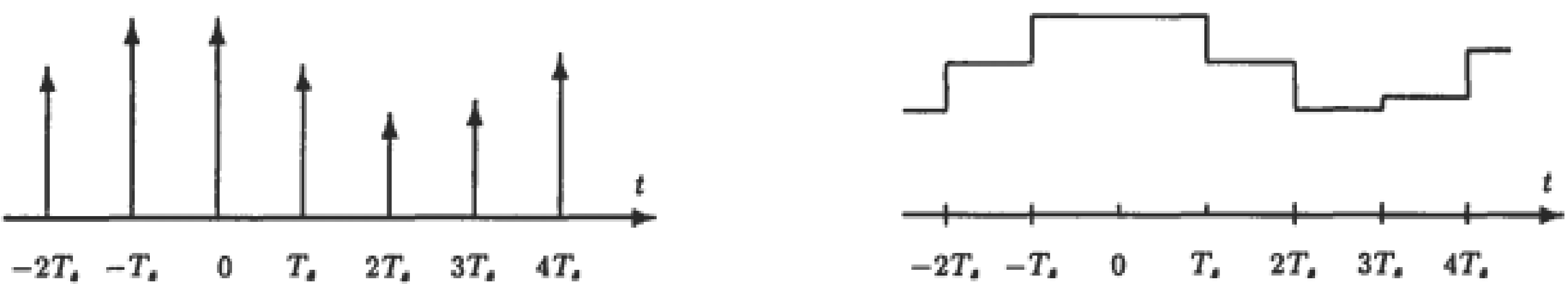


Fig. 1-7. The use of a zero-order hold to interpolate between the samples in $x_s(t)$

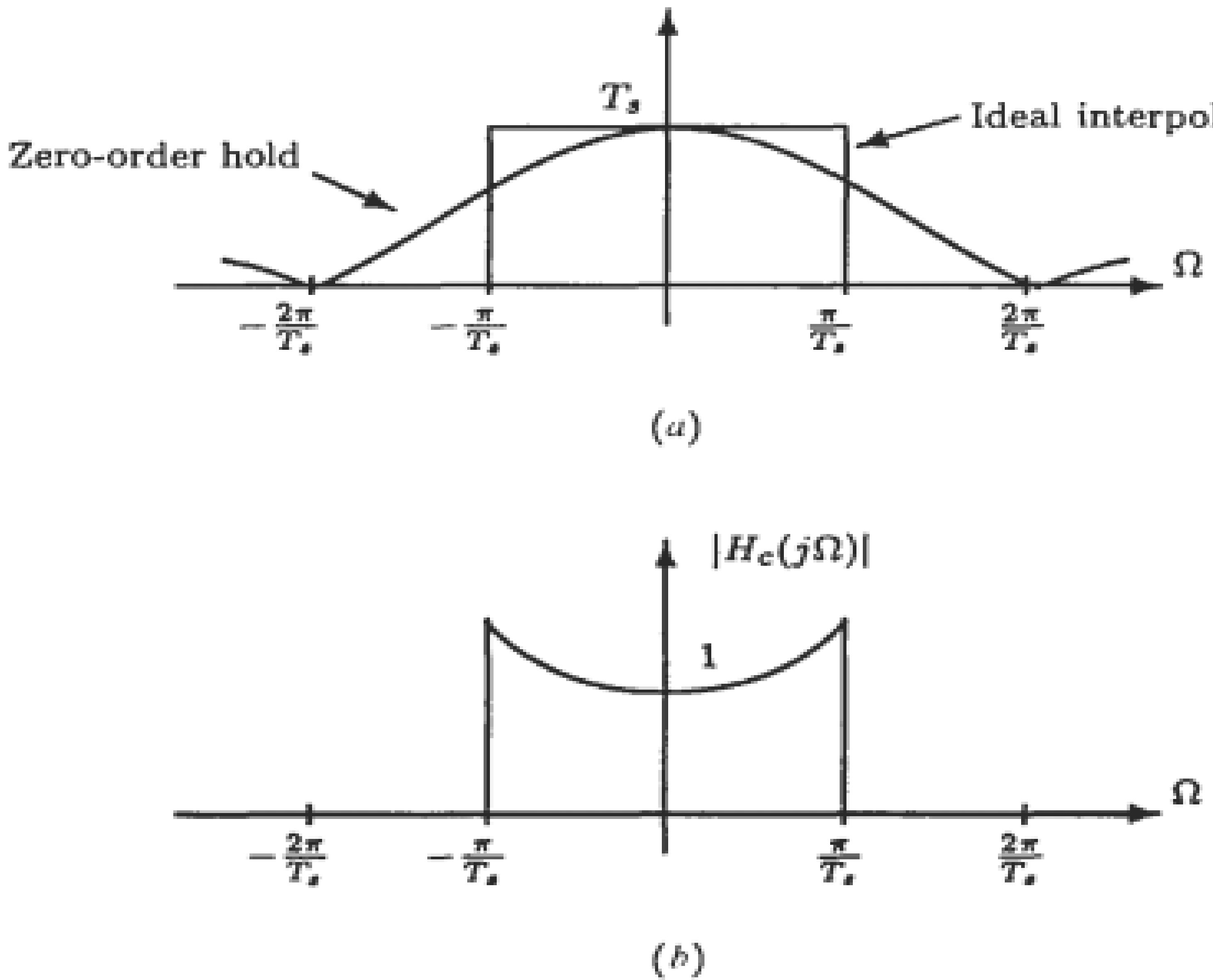
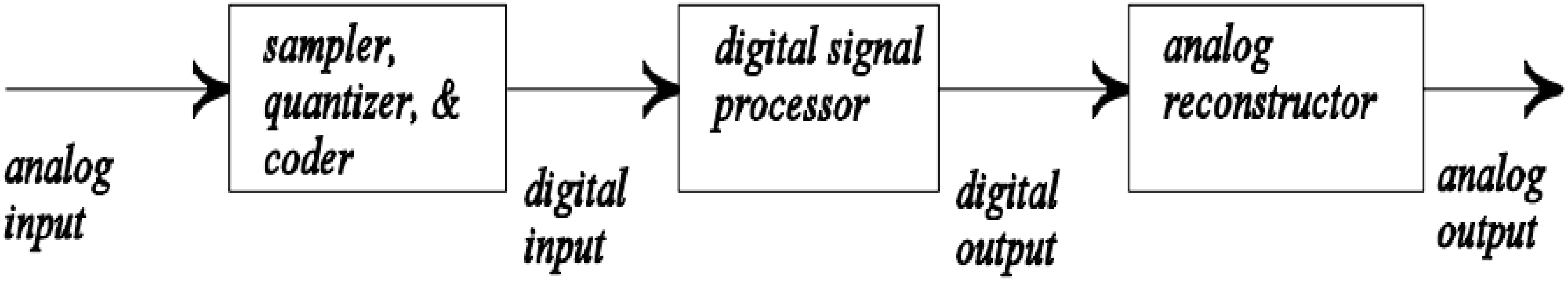


Fig. 1-8. (a) The magnitude of the frequency response of a zero-order hold compared to the ideal reconstruction filter. (b) The ideal reconstruction compensation filter.

1.4 Sampling and Reconstruction

- In fact that continuous-time signal processing can be implemented through the process of sampling.
- Thus, basic digital processing of continuous-time signals proceeds in three stages:

1. The continuous-time signal is digitized, i.e., it is sampled, quantized, and coded to a finite number of bits (A/D conversion).
2. The digitized samples are processed by a digital signal processor.
3. The resulting output samples of the processor is converted back into continuous-time form by an analog reconstructor (D/A conversion).



- ✓ The digital signal processor can be programmed to perform a variety of signal processing operations.
- ✓ Such as filtering, spectrum estimation, and other DSP algorithms.
- ✓ The digital signal processor may be realized by general purpose computer, minicomputer, special purpose DSP chip.

- **Most signals encountered are analog in nature, such signals are continuous in range and domain.**
- **Analog systems to extract some desired information or to change their characteristics. But there are many reasons for processing analog signals digitally:**

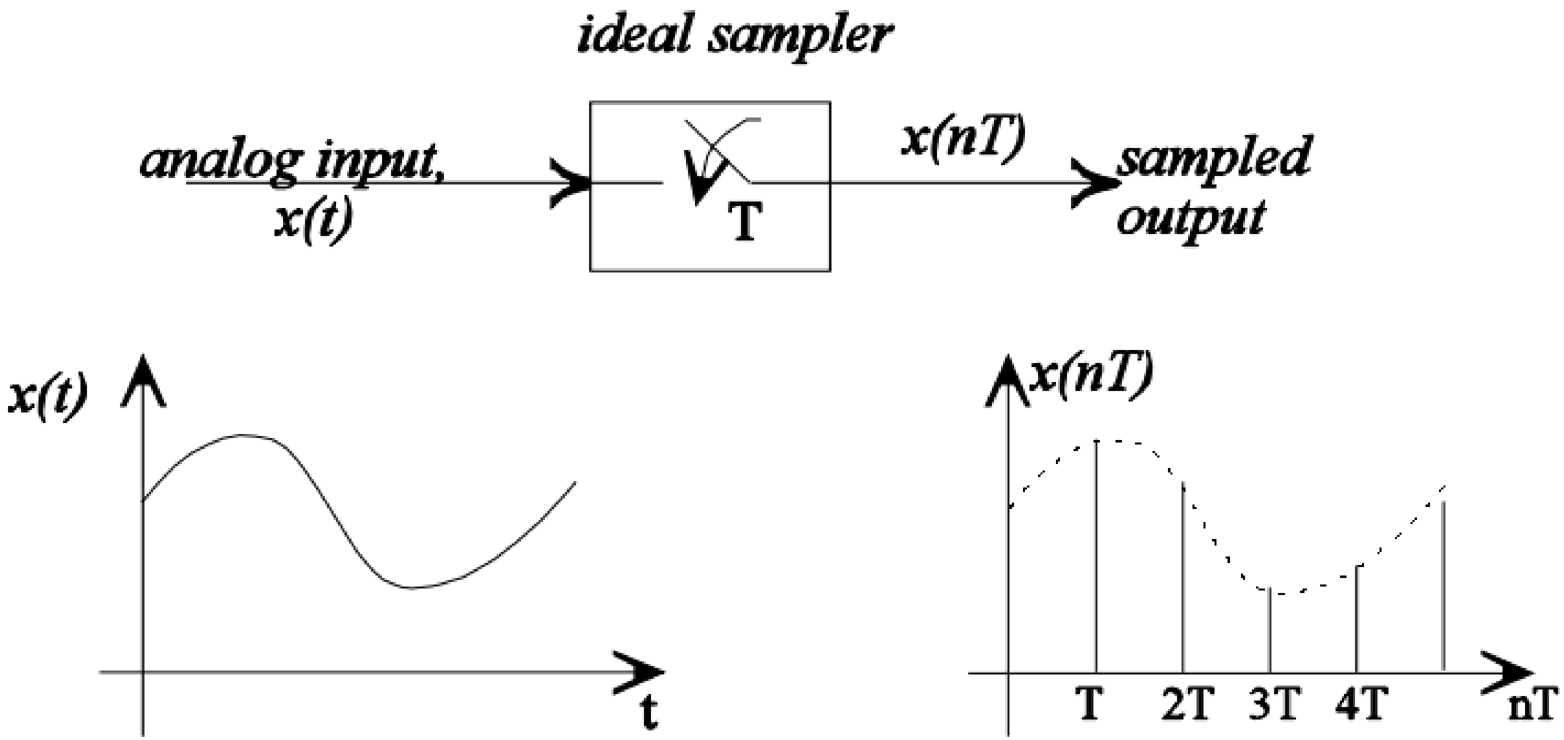
1. **A digital programmable system allows flexibility in reconfiguring the digital signal processing operation simply by changing the program.**
2. **Digital signal processing provides better control of accuracy requirements. This is normally associated with the accuracy requirements of the A/D converter and the digital signal processor.**
3. **Digital signals are easily stored, without deterioration or loss of signal fidelity.**
4. **Digital implementation of the signal processing system is cheaper than its analog counterpart.**

- Some examples are digital signal processing techniques in speech processing and signal transmission on telephone channels, image processing and transmission, processing of signals received from outer space, and in a vast variety of other applications.

- Digital implementation has its limitations, however, speed of operation of A/D converters and digital signal processors is one.
- Wide bandwidth require fast sampling rate A/D converters and digital signal processors.

1.5 Sampling Theorem

A typical method of obtaining a **discrete-time representation** of a continuous-time signal is through periodic sampling, where in a sequence of samples x[nT] is obtained from a continuous-time signal x(t).



- Time is discretized in units of the sampling interval T, i.e., t = nT, n= 0,1,2,...

- **Sampling** introduces a very drastic chopping operation of and therefore will introduce high x(t) frequency components into the frequency spectrum.
- Every frequency component of the original signal is periodically replicated over the entire frequency axis, with period given by the sampling rate

$$f_s = \frac{1}{T_s}, \quad \text{samples/sec. (Hertz)}$$

❖ In general, we will consider what the effect of sampling on the characteristics of the signal will be. Consider a continuous-time complex sinusoid at frequency f_o,

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

and its sampled version

$$x(nT) = e^{j2\pi f_o nT}.$$

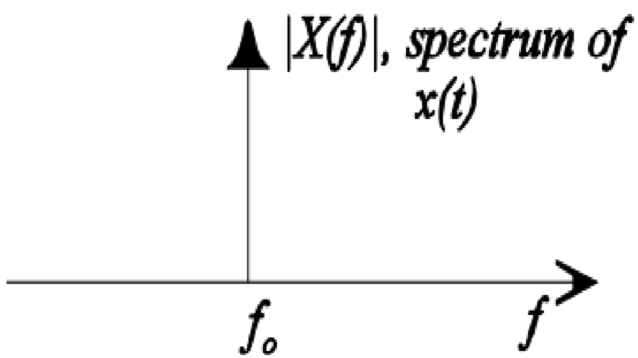
❖ The Fourier transform of the continuous-time complex sinusoid x(t) is X(f) and that of the sampled complex sinusoid x(nT) is $X_d(f)$, obtained by using appropriate Fourier transforms, i.e.,

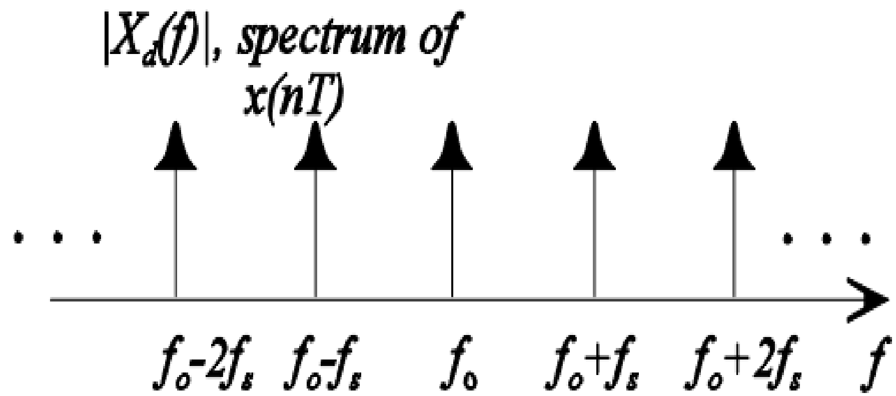
$$X(\Omega) = \int_{-\infty}^{\infty} x(t) e^{-j\Omega t} dt, \qquad \Omega = 2\pi f$$

and

$$X_d(\omega) = \sum_{n=-\infty}^{\infty} x(nT) e^{-j\omega n}, \qquad \omega = 2\pi f T = 2\pi f / f_s$$

- The magnitude response of these transforms are shown below. Note that the sampling process generates high frequency components that appear regularly.
- These components are replicas of the original continuous signal spectrum periodically replicated over the entire frequency axis with period the sampling rate $f_s = 1/T$.





- The sampling Theorem provides a quantitative measure on how to choose the sampling interval T (i.e., the lower bound for the sampling rate f_s).
- The condition on T is very important in that T must be small enough to capture signal variations but not too small to avoid processing too many samples.
- A quantitative criterion is provided by the sampling theorem. The criterion of the sampling theorem may be stated as follows: for accurate representation of x(t) by its samples x(nT), two conditions must be met:

1. The signal must be band limited, taht is, its frequency spectrum must be limited x(t) to contain frequencies up to some maximum frequency, i.e., $f \leq f_{max}$
2. The sampling rate must satisfy the condition that it must, at least, be twice the f_s highest frequency contained in the spectrum of the continuous-time signal x(t), i.e.,

$$f_s \geq 2f_{\max}$$

$$\text{i.e.,} \quad T_s \leq \frac{1}{2f_{\max}}$$

- A typical band limited spectrum of a continuous-time signal is shown below. The highest frequency

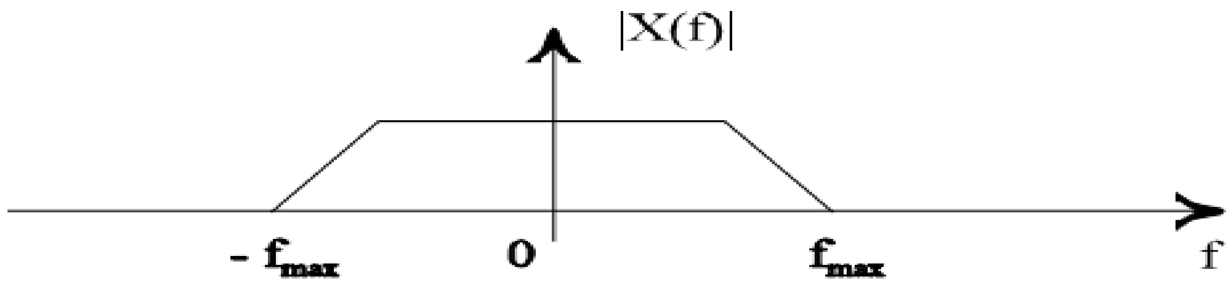


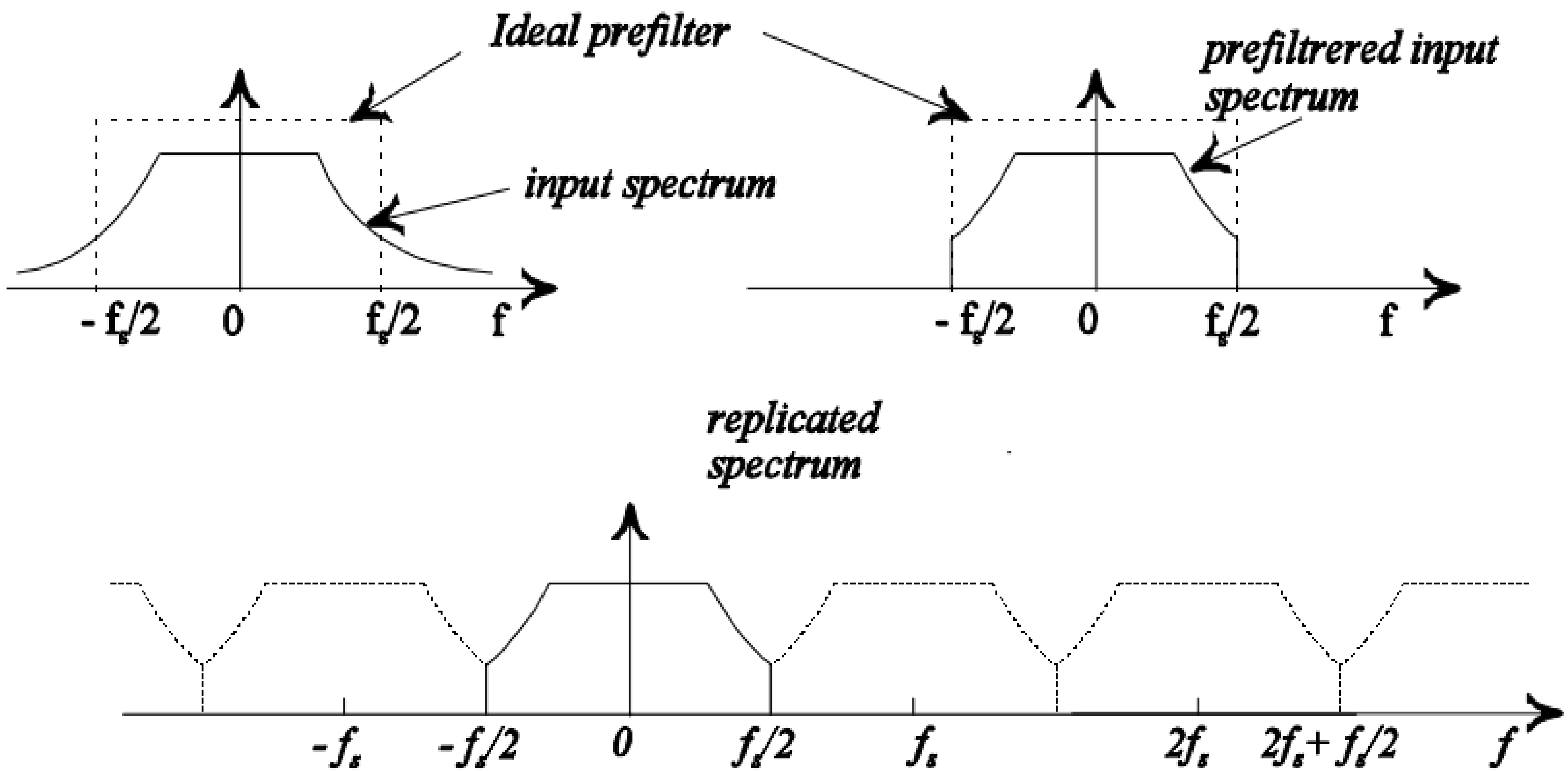
Fig. A typical band limited spectrum.

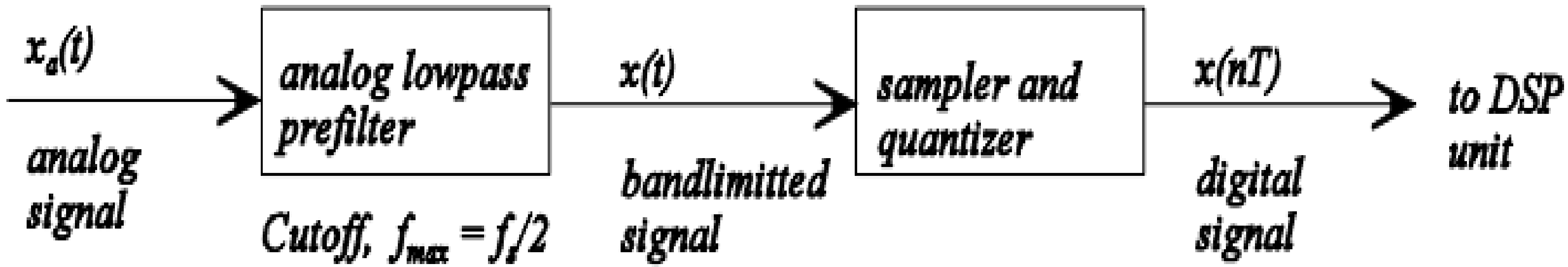
contained in the spectrum is f_{max}. Thus, the minimum sampling rate allowed by the sampling theorem, $f_s = 2f_{\max}$, is called the *Nyquist rate*. The quantity $f_s/2$ is called the *Nyquist frequency* or *folding frequency* and interval $[-f_s/2, f_s/2]$ is called the *Nyquist interval*. The Nyquist frequency, $f_s/2$, defines the cutoff frequency of the lowpass antialiasing prefilters and post filters.

Sampling rates for some common DSP applications:

application	$f_{\max}$	f_s
geophysical	*500 Hz*	*1 kHz*
biomedical	*1 kHz*	*2 kHz*
mechanical	*2 kHz*	*4 kHz*
speech	*4 kHz*	*8 kHz*
audio	*20 kHz*	*40 kHz*
video	*4 MHz*	*8 MHz*

Antialiasing prefilters:- Most signals are **not band-limited** and hence hey must be made so by low pass filtering before sampling. This is done by an analog low pass prefilter known as an antialiasing prefilter. The cutoff frequency of the prefilter, f_{max} , must be at most equal to the Nyquist frequency **$f_s/2$**.





Example:

1. Digital audio: we wish to digitize a music piece at a sampling rate of 40 kHz. The piece must be pre filtered to 20 kHz.
2. The spectrum of speech pre filtered to about 4 kHz results in very intelligible speech. In digital speech applications it is adequate to use a sampling rate of about 8 kHz and pre filter the speech waveform to about 4 kHz.

Solved Problems

1 Consider the discrete-time sequence

$$x(n) = \cos\left(\frac{n\pi}{8}\right)$$

Find two different continuous-time signals that would produce this sequence when sampled at a frequency of $f_s = 10$ Hz.

A continuous-time sinusoid

$$x_a(t) = \cos(\Omega_0 t) = \cos(2\pi f_0 t)$$

that is sampled with a sampling frequency of f_s results in the discrete-time sequence

$$x(n) = x_a(nT_s) = \cos\left(2\pi \frac{f_0}{f_s} n\right)$$

However, note that for any integer k,

$$\cos\left(2\pi \frac{f_0}{f_s} n\right) = \cos\left(2\pi \frac{f_0 + kf_s}{f_s} n\right)$$

Therefore, any sinusoid with a frequency

$$f = f_0 + kf_s$$

will produce the same sequence when sampled with a sampling frequency f_s. With $x(n) = \cos(n\pi/8)$, we want

$$2\pi \frac{f_0}{f_s} = \frac{\pi}{8}$$

or

$$f_0 = \tfrac{1}{16} f_s = 625 \text{ Hz}$$

Therefore, two signals that produce the given sequence are

$$x_1(t) = \cos(1250\pi t)$$

.2 If the Nyquist rate for $x_a(t)$ is Ω_s, what is the Nyquist rate for each of the following signals that are derived from $x_a(t)$?

(a) $\dfrac{dx_a(t)}{dt}$

(b) $x_a(2t)$

(c) $x_a^2(t)$

(d) $x_a(t)\cos(\Omega_0 t)$

(a) The Nyquist rate is equal to twice the highest frequency in $x_a(t)$. If

$$y_a(t) = \frac{dx_a(t)}{dt}$$

then

$$Y_a(j\Omega) = j\Omega X_a(j\Omega)$$

Thus, if $X_a(j\Omega) = 0$ for $|\Omega| > \Omega_0$, the same will be true for $Y_a(j\Omega)$. Therefore, the Nyquist frequency is not changed by differentiation.

(b) The signal $y_a(t) = x_a(2t)$ is formed from $x_a(t)$ by *compressing* the time axis by a factor of 2. This results in an *expansion* of the frequency axis by a factor of 2. Specifically, note that

$$Y_a(j\Omega) = \int_{-\infty}^{\infty} y_a(t)e^{-j\Omega t}\,dt = \int_{-\infty}^{\infty} x_a(2t)e^{-j\Omega t}\,dt$$

$$= \int_{-\infty}^{\infty} \tfrac{1}{2}x_a(\tau)e^{-j\Omega\tau/2}\,d\tau = \tfrac{1}{2}X_a\left(\frac{j\Omega}{2}\right)$$

Consequently, if the Nyquist frequency for $x_a(t)$ is Ω_s, the Nyquist frequency for $y_a(t)$ will be $2\Omega_s$.

(*c*) When two signals are multiplied, their Fourier transforms are convolved. Therefore, if

$$y_a(t) = x_a^2(t)$$

then

$$Y_a(j\Omega) = \frac{1}{2\pi} X_a(j\Omega) * X_a(j\Omega)$$

Thus, the highest frequency in $y_a(t)$ will be twice that of $x_a(t)$, and the Nyquist frequency will be $2\Omega_s$.

(*d*) Modulating a signal by $\cos(\Omega_0 t)$ shifts the spectrum of $x_a(t)$ up and down by Ω_0. Therefore, the Nyquist frequency for $y_a(t) = \cos(\Omega_0 t)x_a(t)$ will be $\Omega_s + 2\Omega_0$.

3. How many bits are needed in an A/D converter if we want a signal-to-quantization noise ratio of at least 90 dB? Assume that $x_a(t)$ is gaussian with a variance σ_x^2, and that the range of the quantizer extends from $-3\sigma_x$ to $3\sigma_x$; that is, $X_{\max} = 3\sigma_x$ (with this value for $X_{\max}$, only about one out of every 1000 samples will exceed the quantizer range).

For a $(B+1)$-bit quantizer, the signal-to-quantization noise ratio is

$$\text{SQNR} = 6.02B + 10.81 - 20\log\frac{X_{\max}}{\sigma_x}$$

With $X_{\max} = 3\sigma_x$ this becomes

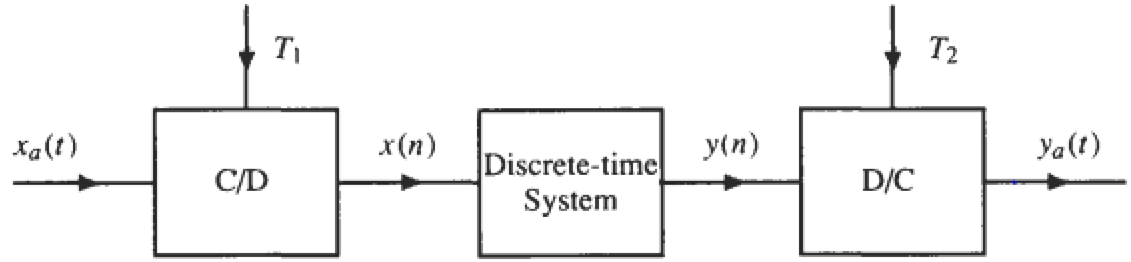
$$\text{SQNR} = 6.02B + 10.81 - 20\log 3 = 6.02B + 10.81 - 9.54 = 6.02B + 1.27$$

If we want a signal-to-quantization noise ratio of 90 dB, we require

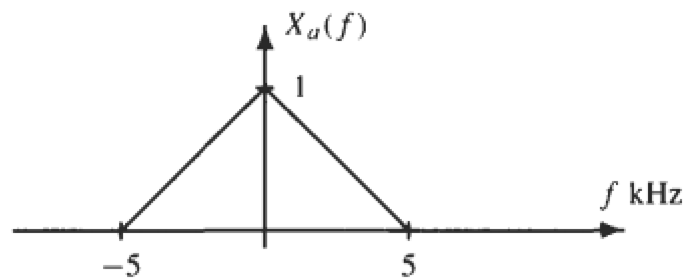
$$B = \frac{90 - 1.27}{6.02} = 14.74$$

or $B + 1 = 16$ bits.

.4 The following system is used to process an analog signal with a discrete-time system.



Suppose that $x_a(t)$ is bandlimited with $X_a(f) = 0$ for $|f| > 5$ kHz as shown in the figure below,



and that the discrete-time system is an ideal low-pass filter with a cutoff frequency of $\pi/2$.

(*a*) Find the Fourier transform of $y_a(t)$ if the sampling frequencies are $f_1 = f_2 = 10$ kHz.

(*b*) Repeat for $f_1 = 20$ kHz and $f_2 = 10$ kHz.

(*c*) Repeat for $f_1 = 10$ kHz and $f_2 = 20$ kHz.

(*a*) When the sampling frequencies of the C/D and D/C converters are the same, and $x_a(t)$ is bandlimited with $X_a(j\Omega) = 0$ for $|\Omega| > \pi/T_1$, this system is equivalent to an analog filter with a frequency response

$$H_a(j\Omega) = \begin{cases} H(e^{j\Omega T_1}) & |\Omega| < \dfrac{\pi}{T_1} \\ 0 & \text{else} \end{cases}$$

Therefore, if $H(e^{j\omega})$ is a low-pass filter with a cutoff frequency $\pi/2$, the cutoff frequency of $H_a(j\Omega)$, denoted by Ω_0, is given by

$$\Omega_0 T_1 = \frac{\pi}{2}$$

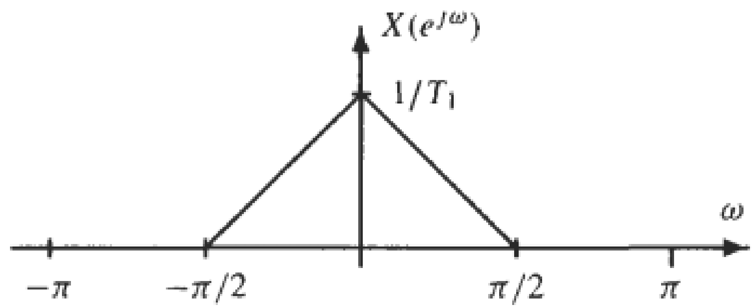
or

$$2\pi f_0 \cdot T_1 = \frac{\pi}{2}$$

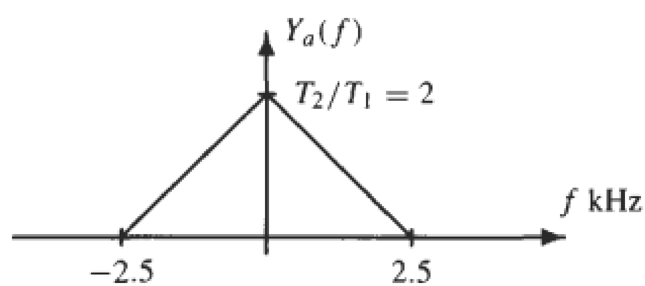
Thus,

$$f_0 = \tfrac{1}{4} f_1 = 2500 \text{ Hz}$$

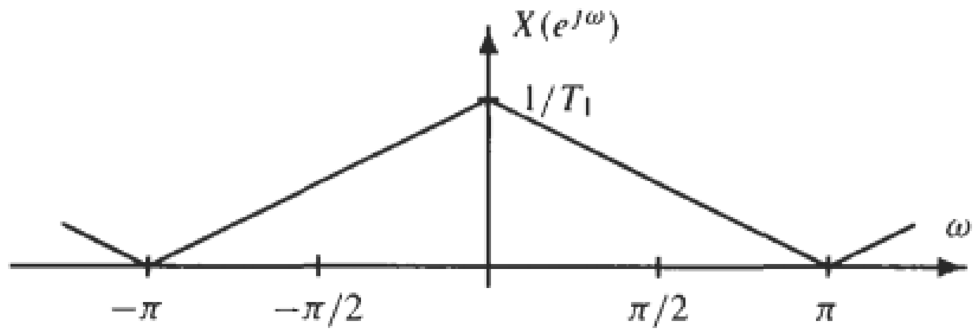
(*b*) When the sampling frequencies of the C/D and D/C are different, it is best to plot the spectrum of the signals as they progress through the system. With $X_a(f)$ as shown above, the discrete-time Fourier transform of $x(n)$ is



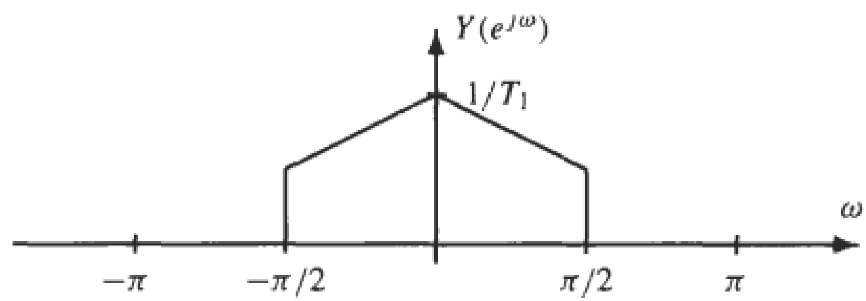
Because the cutoff frequency of the discrete-time low-pass filter is $\pi/2$, $y(n) = x(n)$, and the output of the D/C converter is as plotted below.



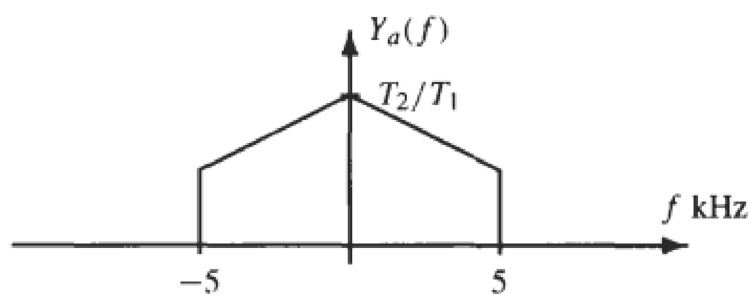
(c) With $f_1 = 10$ kHz, we are sampling $x_a(t)$ at the Nyquist rate, and the spectrum of $x(n)$ is



and the output of the low-pass filter is as shown below.



Therefore, the spectrum of $y_a(t)$ is as follows:



5 A continuous-time signal $x_a(t)$ is to be filtered to remove frequency components in the range 5 kHz $\leq f \leq$ 10 kHz. The maximum frequency present in $x_a(t)$ is 20 kHz. The filtering is to be done by sampling $x_a(t)$, filtering the sampled signal, and reconstructing an analog signal using an ideal D/C converter. Find the minimum sampling frequency that may be used to avoid aliasing, and for this minimum sampling rate, find the frequency response of the ideal digital filter $H(e^{j\omega})$ that will remove the desired frequencies from $x_a(t)$.

Because the highest frequency in $x_a(t)$ is 20 kHz, the minimum sampling frequency to avoid aliasing is $f_s = 40$ kHz. The relationship between the continuous frequency variable Ω and the discrete frequency variable ω is given by

$$\omega = \Omega T_s$$

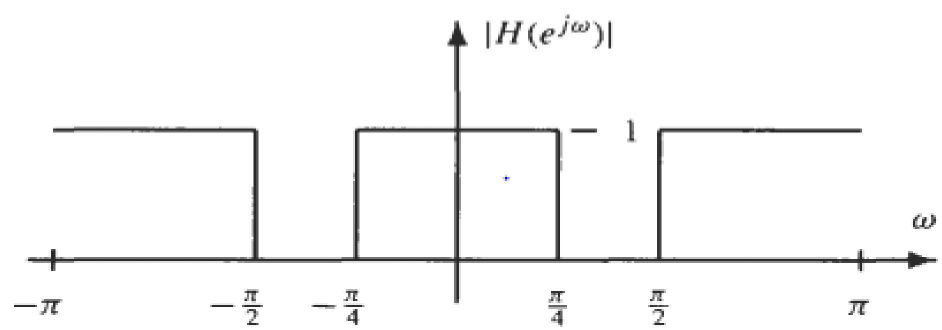
or

$$\omega = 2\pi \frac{f}{f_s}$$

Therefore, the frequency range 5 kHz $\leq f \leq$ 10 kHz corresponds to a digital frequency range

$$\frac{\pi}{4} \leq \omega \leq \frac{\pi}{2}$$

and the desired digital filter is a bandstop filter that has a frequency response as illustrated in the figure below.



Questions ?