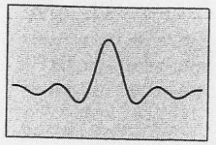


Sampling of Continuous-Time Signals

3

3.0 INTRODUCTION

Discrete-time signals can arise in many ways, but they most commonly occur as representations of continuous-time signals. This is partly due to the fact that processing of continuous-time signals is often carried out by discrete-time processing of sequences obtained by sampling. It is somewhat remarkable that under reasonable constraints, a continuous-time signal can be adequately represented by samples. In this chapter we discuss the process of periodic sampling in some detail, including the issue of aliasing when the signal is not bandlimited or the sampling rate is not high enough. Of particular importance is the fact that continuous-time signal processing can be implemented through a process of sampling, discrete-time processing, and subsequent reconstruction of a continuous-time signal.

3.1 PERIODIC SAMPLING

Although other possibilities exist (see Steiglitz, 1965; Oppenheim and Johnson, 1972), the typical method of obtaining a discrete-time representation of a continuous-time signal is through periodic sampling, wherein a sequence of samples $x[n]$ is obtained from a continuous-time signal $x_c(t)$ according to the relation

$$x[n] = x_c(nT), \qquad -\infty < n < \infty. \tag{3.1}$$

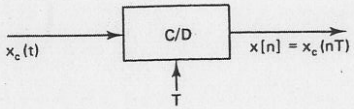


Figure 3.1 Block diagram representation of an ideal continuous-to-discrete (C/D) converter.

In Eq. (3.1), T is the *sampling period*, and its reciprocal, $f_s = 1/T$, is the *sampling frequency*, in samples per second.

We refer to a system that implements the operation of Eq. (3.1) as an *ideal continuous-to-discrete-time (C/D) converter*, and we depict it in block diagram form as indicated in Fig. 3.1. As an example of the relationship between $x_c(t)$ and $x[n]$, in Fig. 2.2 we illustrated a continuous-time speech waveform and the corresponding sequence of samples.

In a practical setting, the operation of sampling is often implemented by an analog-to-digital (A/D) converter. Such systems can be viewed as approximations to the ideal C/D converter. Important considerations in the implementation or choice of an A/D converter include quantization of the output samples, linearity, the need for sample-and-hold circuits, and limitations on the sampling rate. The effects of quantization are briefly discussed in Sections 3.7.2 and 3.7.3. Other practical issues of A/D conversion are outside the scope of this text.

The sampling operation is generally not invertible; i.e., given the output $x[n]$, it is not possible in general to reconstruct $x_c(t)$, the input to the sampler, since many continuous-time signals can produce the same output sequence of samples. The inherent ambiguity in sampling is of primary concern in signal processing. Fortunately it is possible to remove the ambiguity by restricting the class of input signals to the sampler.

It is convenient to mathematically represent the sampling process in the two stages depicted in Figure 3.2(a). This consists of an impulse train modulator followed by conversion of the impulse train to a sequence. Figure 3.2(b) illustrates two continuous-time signals and the results of impulse train sampling. Figure 3.2(c) depicts the corresponding output sequences. The essential difference between $x_s(t)$ and $x[n]$ is that $x_s(t)$ is, in a sense, a continuous-time signal (specifically an impulse train) that is zero except at integer multiples of T. The sequence $x[n]$ on the other hand is indexed on the integer variable n, which in effect introduces a time normalization; i.e, $x[n]$ contains no explicit information about the sampling rate. Furthermore, the samples of $x_c(t)$ are represented by finite numbers in $x[n]$ rather than as the areas of impulses as in $x_s(t)$.

It is important to emphasize that Fig. 3.2(a) is a mathematical representation of sampling, not a representation of any physical circuits or systems designed to implement the sampling operation. Whether a piece of hardware can be construed to be an approximation to the block diagram of Fig. 3.2(a) is a secondary issue at this point. We have introduced this representation of the sampling operation because it leads to a simple derivation of a key result and because this approach leads to a number of important insights that are difficult to obtain from a more formal derivation based on manipulation of Fourier transform formulas.

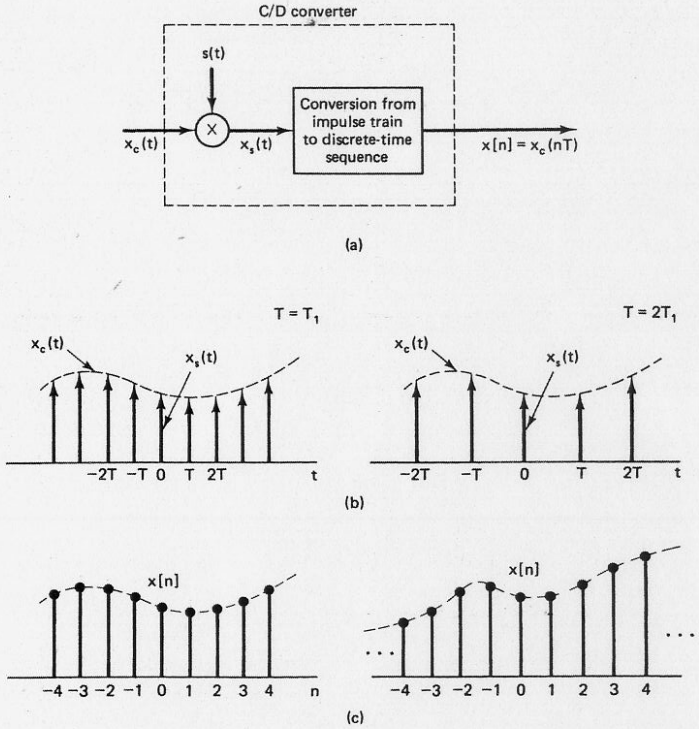


Figure 3.2 Sampling with a periodic impulse train followed by conversion to a discrete-time sequence. (a) Overall system. (b) $x_s(t)$ for two sampling rates. The dashed envelope represents $x_c(t)$. (c) The output sequence for the two different sampling rates.

3.2 FREQUENCY-DOMAIN REPRESENTATION OF SAMPLING

To derive the frequency-domain relation between the input and output of an ideal C/D converter, let us first consider the conversion of $x_c(t)$ to $x_s(t)$ through impulse train modulation. The modulating signal $s(t)$ is a periodic impulse train

$$s(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT), \tag{3.2}$$

where $\delta(t)$ is the unit impulse function or Dirac delta function. Consequently,

$$\begin{aligned} x_s(t) &= x_c(t)s(t) \\ &= x_c(t) \sum_{n=-\infty}^{\infty} \delta(t - nT). \end{aligned} \tag{3.3}$$

Through the "sifting property" of the impulse function, $x_s(t)$ can be expressed as

$$x_s(t) = \sum_{n=-\infty}^{\infty} x_c(nT)\delta(t - nT). \tag{3.4}$$

Let us now consider the Fourier transform of $x_s(t)$. Since from Eq. (3.3), $x_s(t)$ is the product of $x_c(t)$ and $s(t)$, the Fourier transform of $x_s(t)$ is the convolution of the Fourier transforms $X_c(j\Omega)$ and $S(j\Omega)$. The Fourier transform of a periodic impulse train is a periodic impulse train (Oppenheim and Willsky, 1983). Specifically, $S(j\Omega)$ is

$$S(j\Omega) = \frac{2\pi}{T} \sum_{k=-\infty}^{\infty} \delta(\Omega - k\Omega_s), \tag{3.5}$$

where $\Omega_s = 2\pi/T$ is the sampling frequency in radians/s. Since

$$X_s(j\Omega) = \frac{1}{2\pi} X_c(j\Omega) * S(j\Omega),$$

where $*$ denotes the operation of convolution, it follows that

$$X_s(j\Omega) = \frac{1}{T} \sum_{k=-\infty}^{\infty} X_c(j\Omega - kj\Omega_s). \tag{3.6}$$

Equation (3.6) provides the relationship between the Fourier transforms of the input and the output of the impulse train modulator in Fig. 3.2(a). We see from Eq. (3.6) that the Fourier transform of $x_s(t)$ consists of periodically repeated copies of the Fourier transform of $x_c(t)$. The copies of $X_c(j\Omega)$ are shifted by integer multiples of the sampling frequency and then superimposed to produce the periodic Fourier transform of the impulse train of samples. Figure 3.3 depicts the frequency-domain representation of impulse train sampling. Figure 3.3(a) represents a bandlimited Fourier transform where the highest nonzero frequency component in $X_c(j\Omega)$ is at Ω_N. Figure 3.3(b) shows the periodic impulse train $S(j\Omega)$, and Fig. 3.3(c) shows $X_s(j\Omega)$, the result of convolving $X_c(j\Omega)$ with $S(j\Omega)$. From Fig. 3.3(c) it is evident that when

$$\Omega_s - \Omega_N > \Omega_N, \quad \text{or} \quad \Omega_s > 2\Omega_N, \tag{3.7}$$

the replicas of $X_c(j\Omega)$ do not overlap and therefore, when they are added together in Eq. (3.6), there remains (to within a scale factor of $1/T$) a replica of $X_c(j\Omega)$ at each integer multiple of Ω_s. Consequently $x_c(t)$ can be recovered from $x_s(t)$ with an ideal lowpass filter. This is depicted in Fig. 3.4(a), which shows the impulse train modulator followed by a linear time-invariant system with frequency response $H_r(j\Omega)$. For $X_c(j\Omega)$ as in Fig. 3.4(b), $X_s(j\Omega)$ would be as shown in Fig. 3.4(c), where it is assumed that $\Omega_s > 2\Omega_N$. Since

$$X_r(j\Omega) = H_r(j\Omega)X_s(j\Omega), \tag{3.8}$$

it follows that if $H_r(j\Omega)$ is an ideal lowpass filter with gain T and cutoff frequency Ω_c such that

$$\Omega_N < \Omega_c < (\Omega_s - \Omega_N), \tag{3.9}$$

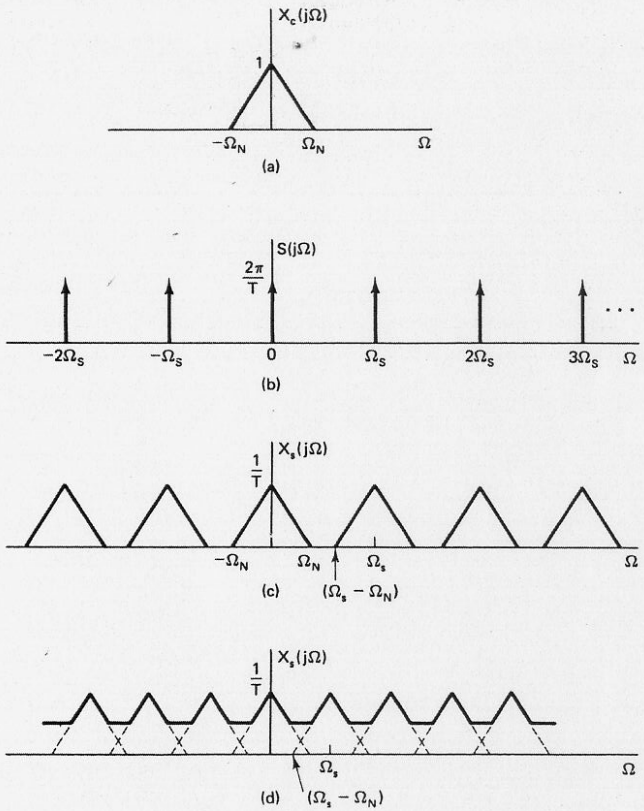


Figure 3.3 Effect in the frequency domain of sampling in the time domain. (a) Spectrum of the original signal. (b) Spectrum of sampling function. (c) Spectrum of sampled signal with $\Omega_S > 2\Omega_N$. (d) Spectrum of sampled signal with $\Omega_S < 2\Omega_N$.

then

$$X_r(j\Omega) = X_c(j\Omega), \tag{3.10}$$

as depicted in Fig. 3.4(e).

If the inequality (3.7) does not hold, i.e., if $\Omega_s \leq 2\Omega_N$, the copies of $X_c(j\Omega)$ overlap so that when they are added together $X_c(j\Omega)$ is no longer recoverable by lowpass filtering. This is illustrated in Fig. 3.3(d). In this case, the reconstructed output $x_r(t)$ in Fig. 3.4(a) is related to the original continuous-time input through a distortion referred to as *aliasing*. Figure 3.5 illustrates aliasing in the frequency domain for the simple case of a cosine signal. Figure 3.5(a) shows the Fourier transform of the signal

$$x_c(t) = \cos \Omega_0 t. \tag{3.11}$$

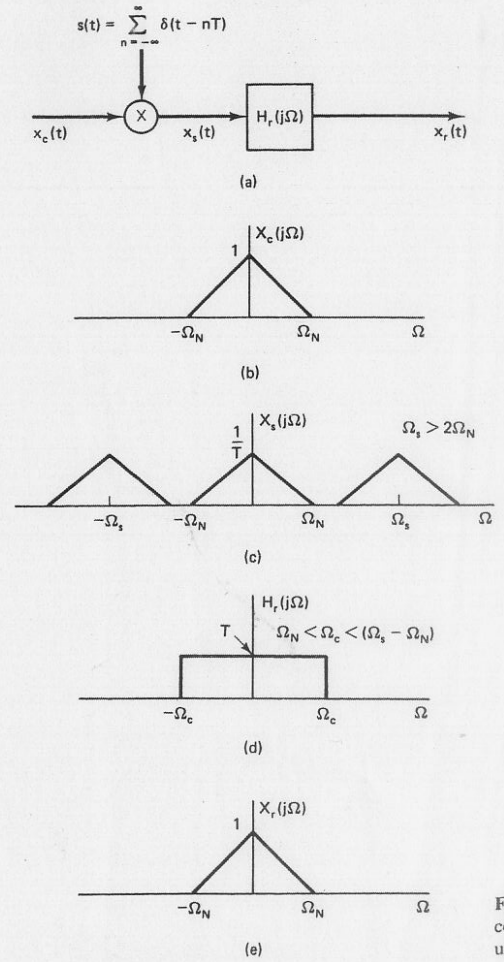


Figure 3.4 Exact recovery of a continuous-time signal from its samples using an ideal lowpass filter.

Part (b) shows the Fourier transform of $x_s(t)$ with $\Omega_0 < \Omega_s/2$, and part (c) shows the Fourier transform of $x_s(t)$ with $\Omega_0 > \Omega_s/2$. Parts (d) and (e) correspond to the Fourier transform of the lowpass filter output for $\Omega_0 < \Omega_s/2 = \pi/T$ and $\Omega_0 > \pi/T$, respectively, with $\Omega_c = \Omega_s/2$. Figures 3.5(c) and (e) correspond to the case of aliasing. With no aliasing (b and d), the reconstructed output $x_r(t)$ is

$$x_r(t) = \cos \Omega_0 t. \tag{3.12}$$

With aliasing, the reconstructed output is

$$x_r(t) = \cos(\Omega_s - \Omega_0)t, \tag{3.13}$$

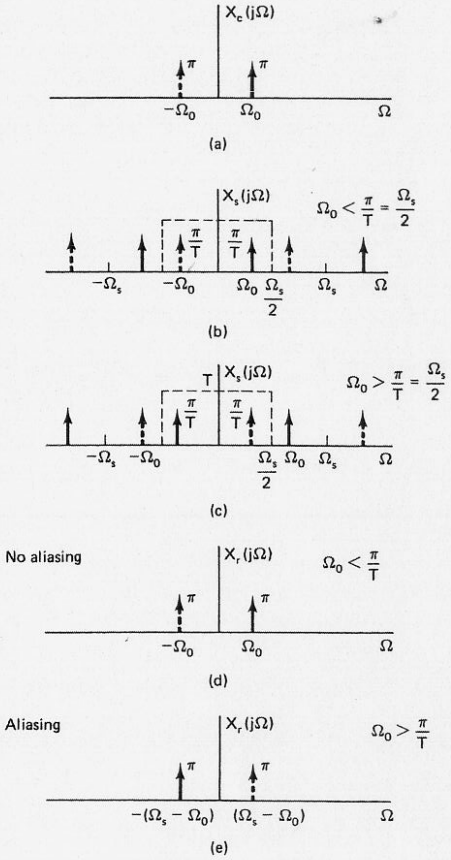


Figure 3.5 The effect of aliasing in the sampling of a cosine signal.

i.e., the higher-frequency signal $\cos \Omega_0 t$ has taken on the identity (alias) of the lower-frequency signal $\cos(\Omega_s - \Omega_0)t$ as a consequence of the sampling and reconstruction. This discussion is the basis for the Nyquist sampling theorem (Nyquist, 1928; Shannon, 1949), stated as follows.

Nyquist Sampling Theorem. Let $x_c(t)$ be a bandlimited signal with

$$X_c(j\Omega) = 0 \qquad \text{for } |\Omega| > \Omega_N. \tag{3.14a}$$

Then $x_c(t)$ is uniquely determined by its samples $x[n] = x_c(nT)$, $n = 0, \pm 1, \pm 2, \ldots$, if

$$\Omega_s = \frac{2\pi}{T} > 2\Omega_N. \tag{3.14b}$$

The frequency Ω_N is commonly referred as the *Nyquist frequency*, and the frequency $2\Omega_N$ that must be exceeded by the sampling frequency is called the *Nyquist rate*.

Thus far we have considered only the impulse train modulator in Fig. 3.2(a). Our eventual objective is to express $X(e^{j\omega})$, the discrete-time Fourier transform of the sequence $x[n]$, in terms of $X_s(j\Omega)$ and $X_c(j\Omega)$. To this end, let us consider an alternative expression for $X_s(j\Omega)$. Applying the Fourier transform to Eq. (3.4), we obtain

$$X_s(j\Omega) = \sum_{n=-\infty}^{\infty} x_c(nT)e^{-j\Omega Tn}. \tag{3.15}$$

Since

$$x[n] = x_c(nT) \tag{3.16}$$

and

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}, \tag{3.17}$$

it follows that

$$X_s(j\Omega) = X(e^{j\omega})|_{\omega=\Omega T} = X(e^{j\Omega T}). \tag{3.18}$$

Consequently from Eqs. (3.6) and (3.18),

$$X(e^{j\Omega T}) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c(j\Omega - jk\Omega_s), \tag{3.19}$$

or, equivalently,

$$X(e^{j\omega}) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c\left(j\frac{\omega}{T} - j\frac{2\pi k}{T}\right). \tag{3.20}$$

From Eqs. (3.18)–(3.20) we see that $X(e^{j\omega})$ is simply a frequency-scaled version of $X_s(j\Omega)$ with the frequency scaling specified by $\omega = \Omega T$. This scaling can alternatively be thought of as a normalization of the frequency axis so that the frequency $\Omega = \Omega_s$ in $X_s(j\Omega)$ is normalized to $\omega = 2\pi$ for $X(e^{j\omega})$. The fact that there is a frequency scaling or normalization in the transformation from $X_s(j\Omega)$ to $X(e^{j\omega})$ is directly associated with the fact that there is a time normalization in the transformation from $x_s(t)$ to $x[n]$. Specifically, as we see in Fig. 3.2, $x_s(t)$ retains a spacing between samples equal to the sampling period T. In contrast, the "spacing" of sequence values $x[n]$ is always unity, i.e., the time axis is normalized by a factor of T. Correspondingly in the frequency domain, the frequency axis is normalized by a factor of $f_s = 1/T$.

3.3 RECONSTRUCTION OF A BANDLIMITED SIGNAL FROM ITS SAMPLES

According to the sampling theorem, samples of a continuous-time bandlimited signal taken frequently enough are sufficient to represent the signal exactly in the sense that the signal can be recovered from the samples and from knowledge of the sampling

period. Impulse train modulation provides a convenient means for understanding the process of reconstructing the continuous-time bandlimited signal from its samples.

In Section 3.2 we saw that if the conditions of the sampling theorem are met and if the modulated impulse train is filtered by an appropriate lowpass filter, then the Fourier transform of the filter output will be identical to the Fourier transform of the original continuous-time signal $x_c(t)$, and thus the output of the filter will be $x_c(t)$. If we are given a sequence of samples $x[n]$, we can form an impulse train $x_s(t)$ in which successive impulses are assigned an area equal to successive sequence values, i.e.,

$$x_s(t) = \sum_{n=-\infty}^{\infty} x[n]\delta(t - nT). \tag{3.21}$$

The nth sample is associated with the impulse at $t = nT$, where T is the sampling period associated with the sequence $x[n]$. If this impulse train is the input to an ideal lowpass continuous-time filter with frequency response $H_r(j\Omega)$ and impulse response $h_r(t)$, then the output of the filter will be

$$x_r(t) = \sum_{n=-\infty}^{\infty} x[n]h_r(t - nT). \tag{3.22}$$

A block diagram representation of this signal reconstruction process is shown in Fig. 3.6(a). Recall that the ideal reconstruction filter has a gain of T (to compensate for the factor of $1/T$ in Eq. 3.19 or 3.20) and a cutoff frequency Ω_c between Ω_N and $\Omega_s - \Omega_N$. A convenient and commonly used choice of the cutoff frequency is $\Omega_c = \Omega_s/2 = \pi/T$. This choice is appropriate for any relationship between Ω_s and Ω_N that avoids aliasing (i.e., so long as $\Omega_s > 2\Omega_N$). Figure 3.6(b) shows the frequency response of the ideal reconstruction filter. The corresponding impulse response, $h_r(t)$, is the inverse Fourier transform of $H_r(j\Omega)$, and for cutoff frequency π/T is given by

$$h_r(t) = \frac{\sin \pi t/T}{\pi t/T}. \tag{3.23}$$

This impulse response is shown in Figure 3.6(c). From substituting Eq. (3.23) into Eq. (3.22) it follows that

$$x_r(t) = \sum_{n=-\infty}^{\infty} x[n] \frac{\sin[\pi(t - nT)/T]}{\pi(t - nT)/T}. \tag{3.24}$$

From the frequency-domain argument of Section 3.2 we saw that if $x[n] = x_c(nT)$, where $X_c(j\Omega) = 0$ for $|\Omega| \geq \pi/T$, then $x_r(t)$ is equal to $x_c(t)$. It is not immediately obvious that this is true by considering Eq. (3.24) alone. However, useful insight is gained by looking at Eq. (3.24) more closely. First let us consider the function $h_r(t)$ given by Eq. (3.23). We note that

$$h_r(0) = 1. \tag{3.25a}$$

This follows from l'Hôpital's rule. In addition,

$$h_r(nT) = 0 \qquad \text{for } n = \pm 1, \pm 2, \ldots. \tag{3.25b}$$

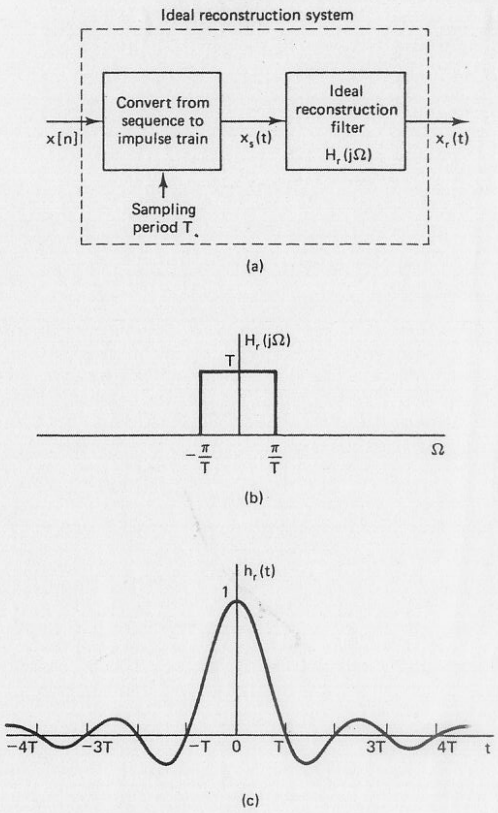


Figure 3.6 (a) Block diagram of an ideal bandlimited signal reconstruction system. (b) Frequency response of an ideal reconstruction filter. (c) Impulse response of an ideal reconstruction filter.

It follows from Eqs. (3.25) and Eq. (3.22) that if $x[n] = x_c(nT)$, then

$$x_r(mT) = x_c(mT) \tag{3.26}$$

for all integer values of m. That is, the signal that is reconstructed by Eq. (3.24) has the same values at the sampling times as the original continuous-time signal, independent of the sampling period T.

In Fig. 3.7 we show a continuous-time signal $x_c(t)$ and the corresponding modulated impulse train. Figure 3.7(c) shows several of the terms

$$x[n] \frac{\sin[\pi(t - nT)/T]}{\pi(t - nT)/T}$$

and the resulting reconstructed signal $x_r(t)$. As suggested by this figure, the ideal lowpass filter *interpolates* between the impulses of $x_s(t)$ to construct a continuous-time signal $x_r(t)$. From Eq. (3.26), the resulting signal is an exact reconstruction of $x_c(t)$ at

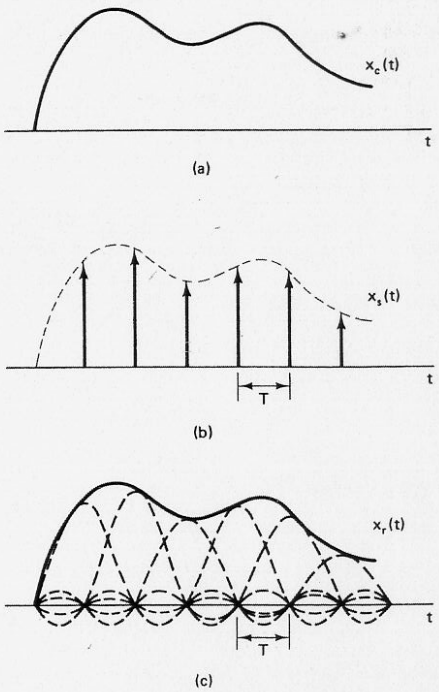


Figure 3.7 Ideal bandlimited interpolation.

the sampling times. The fact that the lowpass filter interpolates the correct reconstruction between the samples, if there is no aliasing, follows from our frequency-domain analysis of the sampling and reconstruction process.

It is useful to formalize the preceding discussion by defining an ideal system for reconstructing a bandlimited signal from a sequence of samples. We will call this system the *ideal discrete-to-continuous-time (D/C) converter*. The desired system is depicted in Fig. 3.8. As we have seen, the ideal reconstruction process can be represented as the conversion of the sequence to an impulse train as in Eq. (3.21) followed by filtering with an ideal lowpass filter, resulting in the output given by Eq. (3.24). The intermediate step of conversion to an impulse train is a mathematical convenience in deriving Eq. (3.24) and in understanding the signal reconstruction process. However, once we are familiar with this process, it is useful to define a more compact representation, as depicted in Fig. 3.8(b), where the input is the sequence $x[n]$ and the output is the continuous-time signal $x_r(t)$ given by Eq. (3.24).

The properties of the ideal D/C converter are most easily seen in the frequency domain. To derive an input/output relation in the frequency domain, consider the Fourier transform of Eq. (3.24), which is

$$X_r(j\Omega) = \sum_{n=-\infty}^{\infty} x[n]H_r(j\Omega)e^{-j\Omega Tn}.$$

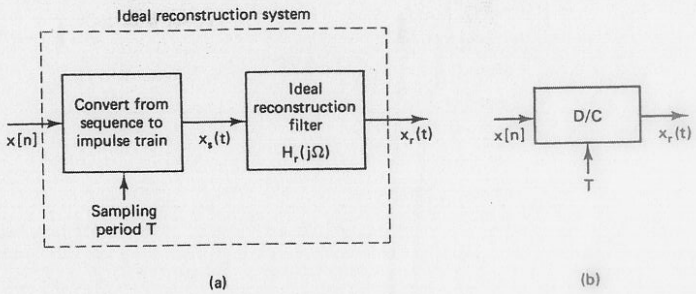


Figure 3.8 (a) Ideal bandlimited signal reconstruction. (b) Equivalent representation as an ideal D/C converter.

By factoring $H_r(j\Omega)$ out of the sum, we can write

$$X_r(j\Omega) = H_r(j\Omega)X(e^{j\Omega T}). \tag{3.27}$$

Equation (3.27) provides a frequency-domain description of the ideal D/C converter. According to Eq. (3.27), $X(e^{j\omega})$ is frequency scaled (i.e., ω is replaced by ΩT). The ideal lowpass filter $H_r(j\Omega)$ selects the base period of the resulting periodic Fourier transform $X(e^{j\Omega T})$ and compensates for the $1/T$ scaling inherent in sampling. Thus, if the sequence $x[n]$ has been obtained by sampling a bandlimited signal at the Nyquist rate or higher, then the reconstructed signal $x_r(t)$ will be equal to the original bandlimited signal. In any case, it is also clear from Eq. (3.27) that the output of the ideal D/C converter is always bandlimited to at most the cutoff frequency of the lowpass filter, which is typically taken to be one-half the sampling frequency.

3.4 DISCRETE-TIME PROCESSING OF CONTINUOUS-TIME SIGNALS

A major application of discrete-time systems is in the processing of continuous-time signals. This is accomplished by a system of the general form depicted in Fig. 3.9. The system is a cascade of a C/D converter followed by a discrete-time system followed by

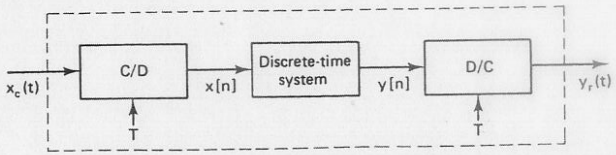


Figure 3.9 Discrete-time processing of continuous-time signals.

a D/C converter. The block diagram of Fig. 3.9 represents a large class of systems since the sampling rate and the discrete-time system can be chosen as we wish. Note that the overall system is equivalent to a continuous-time system since it transforms the continuous-time input signal $x_c(t)$ into the continuous-time output signal $y_r(t)$. However, the properties of the system are dependent on the choice of the discrete-time system and the sampling rate. We assume in Fig. 3.9 that the C/D and D/C converters have the same sampling rate. This is not essential, and later sections of this chapter and some of the problems at the end of the chapter consider systems where the input and output sampling rates are not the same.

The previous sections of this chapter have been devoted to understanding the C/D and D/C conversion operations in Fig. 3.9. For convenience and as a first step in understanding the overall system of Fig. 3.9, we summarize the mathematical representations of these operations.

The C/D converter produces a discrete-time signal

$$x[n] = x_c(nT), \tag{3.28}$$

i.e., a sequence of samples of the continuous-time input signal $x_c(t)$. The Fourier transform of this sequence is related to the Fourier transform of the continuous-time input signal as

$$X(e^{j\omega}) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c\left(j\frac{\omega}{T} - j\frac{2\pi k}{T}\right). \tag{3.29}$$

The D/C converter creates a continuous-time output signal of the form

$$y_r(t) = \sum_{n=-\infty}^{\infty} y[n]\frac{\sin[\pi(t-nT)/T]}{\pi(t-nT)/T}, \tag{3.30}$$

where the sequence $y[n]$ is the output of the discrete-time system when the input to the system is $x[n]$. From Eq. (3.27), $Y_r(j\Omega)$, the continuous-time Fourier transform of $y_r(t)$, and $Y(e^{j\omega})$, the discrete-time Fourier transform of $y[n]$, are related by

$$\begin{aligned} Y_r(j\Omega) &= H_r(j\Omega)Y(e^{j\Omega T}) \\ &= \begin{cases} TY(e^{j\Omega T}), & |\Omega| < \pi/T, \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \tag{3.31}$$

Next let us relate the output sequence $y[n]$ to the input sequence $x[n]$ or equivalently $Y(e^{j\omega})$ to $X(e^{j\omega})$. A simple example is the identity system, i.e., $y[n] = x[n]$. This is the case that we have studied in detail already. We know that if $x_c(t)$ has a bandlimited Fourier transform such that $X_c(j\Omega) = 0$ for $|\Omega| \geq \pi/T$ and if the discrete-time system in Fig. 3.9 is the identity system such that $y[n] = x[n] = x_c(nT)$, then the output will be $y_r(t) = x_c(t)$. Recall that in proving this result, we utilized the frequency-domain representations of the continuous-time and discrete-time signals, since the key concept of aliasing is most easily understood in the frequency domain. Likewise, when we deal with systems more complicated than the identity system, we generally carry out the analysis in the frequency domain. If the discrete-time system is nonlinear or time-varying, it is usually difficult to obtain a general relationship between the Fourier transforms of the input and the output of the

system. (In Problem 3.16 we consider an example of the system of Fig. 3.9 in which the discrete-time system is nonlinear.) However, the linear time-invariant case leads to a rather simple and very useful result.

3.4.1 Linear Time-Invariant Discrete-Time Systems

If the discrete-time system in Fig. 3.9 is linear and time-invariant, we then have

$$Y(e^{j\omega}) = H(e^{j\omega})X(e^{j\omega}), \tag{3.32}$$

where $H(e^{j\omega})$ is the frequency response of the system or, equivalently, the Fourier transform of the unit sample response, and $X(e^{j\omega})$ and $Y(e^{j\omega})$ are the Fourier transforms of the input and output, respectively. Combining Eqs. (3.31) and (3.32), we obtain

$$Y_r(j\Omega) = H_r(j\Omega)H(e^{j\Omega T})X(e^{j\Omega T}). \tag{3.33}$$

Next, using Eq. (3.29) with $\omega = \Omega T$, we have

$$Y_r(j\Omega) = H_r(j\Omega)H(e^{j\Omega T})\frac{1}{T}\sum_{k=-\infty}^{\infty} X_c\left(j\Omega - j\frac{2\pi k}{T}\right). \tag{3.34}$$

If $X_c(j\Omega) = 0$ for $|\Omega| \geq \pi/T$, then the ideal lowpass reconstruction filter $H_r(j\Omega)$ cancels the factor $1/T$ and selects only the term in Eq. (3.34) for $k = 0$, i.e.,

$$Y_r(j\Omega) = \begin{cases} H(e^{j\Omega T})X_c(j\Omega), & |\Omega| < \pi/T, \\ 0, & |\Omega| \geq \pi/T. \end{cases} \tag{3.35}$$

Thus if $X_c(j\Omega)$ is bandlimited and the sampling rate is above the Nyquist rate, the output is related to the input through an equation of the form

$$Y_r(j\Omega) = H_{eff}(j\Omega)X_c(j\Omega), \tag{3.36}$$

where

$$H_{eff}(j\Omega) = \begin{cases} H(e^{j\Omega T}), & |\Omega| < \pi/T, \\ 0, & |\Omega| \geq \pi/T. \end{cases} \tag{3.37}$$

That is, the overall continuous-time system is equivalent to a linear time-invariant system whose *effective* frequency response is given by Eq. (3.37).

It is important to emphasize that the linear and time-invariant behavior of the system of Fig. 3.9 depends on two factors. First, the discrete-time system must be linear and time-invariant. Second, the input signal must be bandlimited and the sampling rate must be high enough so that any aliased components are removed by the discrete-time system. As a simple illustration of this second condition being violated, consider the case when $x_c(t)$ is a single unit-amplitude pulse whose duration is less than the sampling period. If the pulse is unity at $t = 0$, then $x[n] = \delta[n]$. However, it is clearly possible to shift the pulse so that it is not aligned with any of the sampling times, i.e., $x[n] = 0$ for all n. Obviously such a pulse, being timelimited, is not bandlimited. Even if the discrete-time system is the identity system, such that $y[n] = x[n]$, the overall system will not be time-invariant. In general, if the

discrete-time system in Fig. 3.9 is linear and time-invariant and the sampling frequency is above the Nyquist rate associated with the bandwidth of the input $x_c(t)$, then the overall system is guaranteed to be equivalent to a linear time-invariant continuous-time system with an effective frequency response given by Eq. (3.37). Furthermore, Eq. (3.37) is valid even if some aliasing occurs in the C/D converter as long as $H(e^{j\omega})$ does not pass the aliased components. Example 3.1 is a simple illustration of this.

Example 3.1 The Ideal Lowpass Filter

Consider Fig. 3.9, with the linear time-invariant discrete-time system having frequency response

$$H(e^{j\omega}) = \begin{cases} 1, & |\omega| < \omega_c, \\ 0, & \omega_c < |\omega| \le \pi. \end{cases} \tag{3.38}$$

This frequency response is, of course, periodic with period 2π, as shown in Fig. 3.10(a). For bandlimited inputs sampled above the Nyquist rate, it follows from Eq. (3.37) that the overall system of Fig. 3.9 will behave as a linear time-invariant continuous-time system with frequency response

$$H_{eff}(j\Omega) = \begin{cases} 1, & |\Omega T| < \omega_c \text{ or } |\Omega| < \omega_c/T, \\ 0, & |\Omega T| > \omega_c \text{ or } |\Omega| > \omega_c/T. \end{cases} \tag{3.39}$$

As shown in Fig. 3.10(b), this effective frequency response is that of an ideal lowpass filter with cutoff frequency $\Omega_c = \omega_c/T$.

As an interpretation of this result, consider the graphical illustration given in Fig. 3.11. Figure 3.11(a) indicates the Fourier transform of a bandlimited signal. Figure 3.11(b) shows the Fourier transform of the intermediate modulated impulse train, which is identical to $X(e^{j\Omega T})$, the discrete-time Fourier transform of the sequence of samples evaluated for $\omega = \Omega T$. In Fig. 3.11(c), the discrete-time Fourier transform of the sequence of samples and the frequency response of the discrete-time system are both plotted as a function of the normalized discrete-time frequency variable ω. Figure 3.11(d) shows $Y(e^{j\omega}) = H(e^{j\omega})X(e^{j\omega})$, the Fourier transform of the output of the discrete-time

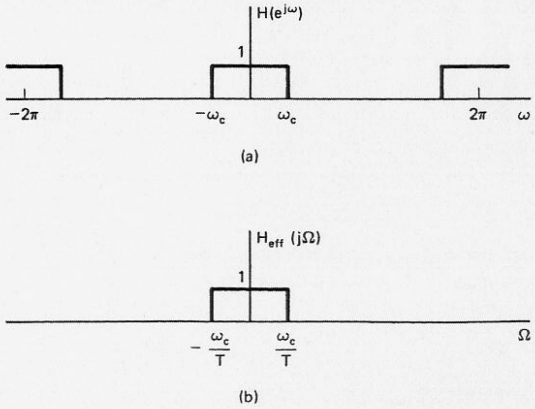


Figure 3.10 (a) Frequency response of the discrete-time system in Fig. 3.9. (b) Corresponding frequency response for bandlimited inputs.

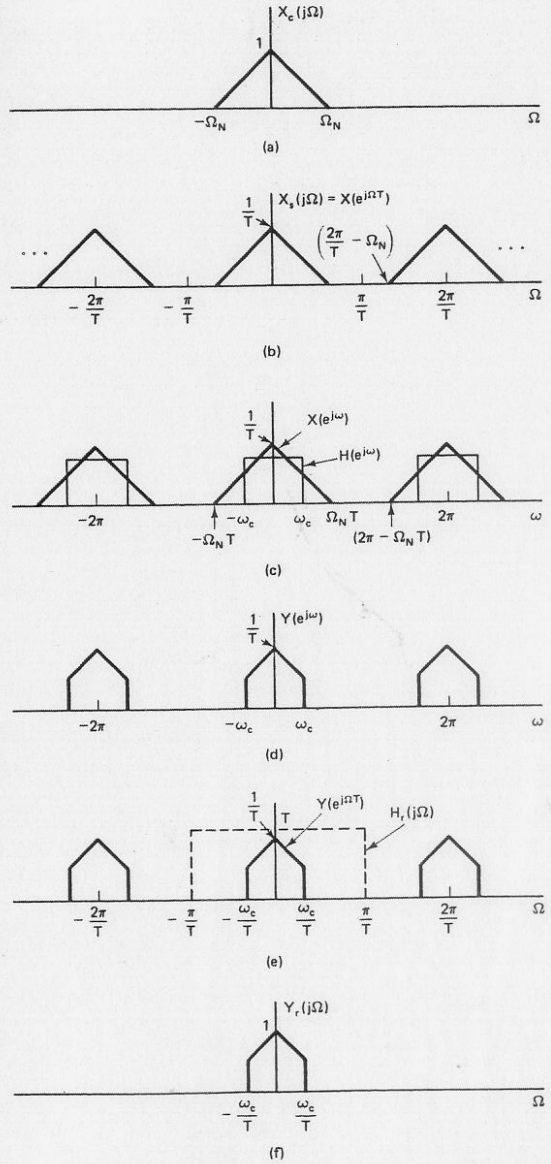


Figure 3.11 (a) Fourier transform of a bandlimited input signal. (b) Fourier transform of sampled input plotted as a function of continuous-time frequency Ω. (c) Fourier transform $X(e^{j\omega})$ of sequence of samples and frequency response $H(e^{j\omega})$ of discrete-time system plotted vs. ω. (d) Fourier transform of output of discrete-time system. (e) Fourier transform of output of discrete-time system and frequency response of ideal reconstruction filter plotted vs. Ω. (f) Fourier transform of output.

system. Figure 3.11(e) illustrates the Fourier transform of the output of the discrete-time system as a function of continuous-time frequency Ω together with the frequency response of the ideal reconstruction filter $H_r(j\Omega)$ of the D/C converter. Finally, Fig. 3.11(f) shows the resulting Fourier transform of the output of the D/C converter. By comparing Figs. 3.11(a) and 3.11(f), we see that the system behaves as a linear time-invariant system with frequency response given by Eq. (3.39) and plotted in Fig. 3.10(b).

Several important points are illustrated in this example. First note that the ideal lowpass discrete-time filter with discrete-time cutoff frequency ω_c has the effect of an ideal lowpass filter with cutoff frequency $\Omega_c = \omega_c/T$ when used in the configuration of Fig. 3.9. This cutoff frequency depends on both ω_c and T. In particular, by using a fixed discrete-time lowpass filter but varying the sampling period T, an equivalent continuous-time lowpass filter with a variable cutoff frequency can be implemented. For example, if T were chosen so that $\Omega_N T < \omega_c$, then the output of the system of Fig. 3.9 would be $y_r(t) = x_c(t)$. Also, as illustrated in Problem 3.11, Eq. (3.39) will be valid even if some aliasing is present in Figs. 3.11(b) and (c), as long as these distorted (aliased) components are eliminated by the filter $H(e^{j\omega})$. In particular, from Fig. 3.11(c), we see that for no aliasing to be present in the output, we require that

$$(2\pi - \Omega_N T) > \omega_c \tag{3.40}$$

as compared with the Nyquist requirement that

$$(2\pi - \Omega_N T) > \Omega_N T. \tag{3.41}$$

As another example of continuous-time processing using a discrete-time system, let us consider the implementation of an ideal differentiator for bandlimited signals.

Example 3.2 Ideal Bandlimited Differentiator

The ideal continuous-time differentiator system is defined by

$$y_c(t) = \frac{d}{dt}[x_c(t)], \tag{3.42}$$

with corresponding frequency response

$$H_c(j\Omega) = j\Omega. \tag{3.43}$$

Since we are considering a realization in the form of Fig. 3.9, the inputs are restricted to be bandlimited. For processing bandlimited signals, it is sufficient that

$$H_{eff}(j\Omega) = \begin{cases} j\Omega, & |\Omega| < \pi/T, \\ 0, & |\Omega| \geq \pi/T, \end{cases} \tag{3.44}$$

as depicted in Fig. 3.12(a). The corresponding discrete-time system has frequency response

$$H(e^{j\omega}) = j\omega/T, \qquad |\omega| < \pi, \tag{3.45}$$

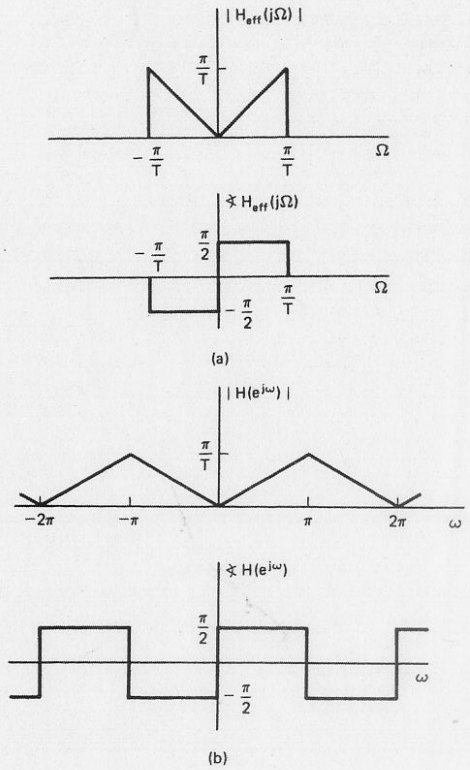


Figure 3.12 (a) Frequency response of a continuous-time ideal bandlimited differentiator $H_c(j\Omega) = j\Omega$, $|\Omega| < \Omega_c$. (b) Frequency response of discrete-time filter to implement a continuous-time bandlimited differentiator.

and is periodic with period 2π. This frequency response is plotted in Fig. 3.12(b). The corresponding impulse response can be shown to be

$$h[n] = \frac{\pi n \cos \pi n - \sin \pi n}{\pi n^2 T}, \qquad -\infty < n < \infty,$$

$$h[n] = \begin{cases} 0, & n = 0, \\ \dfrac{\cos \pi n}{nT}, & n \neq 0. \end{cases} \tag{3.46}$$

Thus, if a discrete-time system with this impulse response were used in the configuration of Fig. 3.9, the output for every bandlimited input would be the derivative of the input.

3.4.2 Impulse Invariance

We have shown that the cascade system of Fig. 3.9 can be equivalent to a linear time-invariant system for bandlimited input signals. Let us now assume that, as depicted in

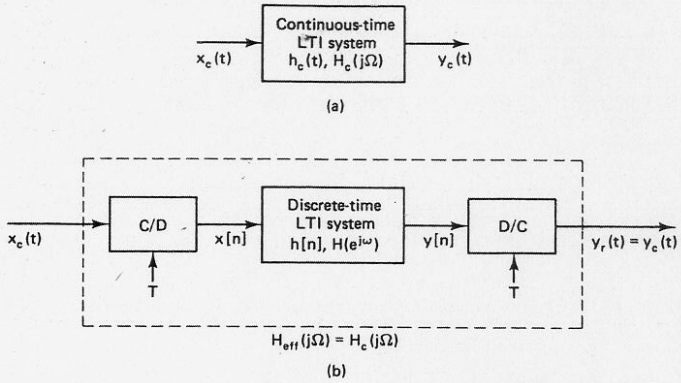


Figure 3.13 (a) Continuous-time LTI system. (b) Equivalent system for bandlimited inputs.

Figure 3.13, we are given a desired continuous-time system that we wish to implement in the form of Fig. 3.9. With $H_c(j\Omega)$ bandlimited, Eq. (3.37) specifies how to choose $H(e^{j\omega})$ so that $H_{eff}(j\Omega) = H_c(j\Omega)$. Specifically,

$$H(e^{j\omega}) = H_c(j\omega/T), \qquad |\omega| < \pi, \tag{3.47}$$

with the further requirement that T be chosen such that

$$H_c(j\Omega) = 0, \qquad |\Omega| \geq \pi/T. \tag{3.48}$$

Under the constraints of Eqs. (3.47) and (3.48), there is also a straightforward and useful relationship between the continuous-time impulse response $h_c(t)$ and the discrete-time impulse response $h[n]$. In particular, as we verify shortly,

$$h[n] = Th_c(nT), \tag{3.49}$$

i.e., the impulse response of the discrete-time system is a scaled, sampled version of $h_c(t)$. When $h[n]$ and $h_c(t)$ are related through Eq. (3.49), the discrete-time system is said to be an *impulse-invariant* version of the continuous-time system.

Equation (3.49) is a direct consequence of the discussion in Section 3.2. Specifically, with $x[n]$ and $x_c(t)$ replaced by $h[n]$ and $h_c(t)$ in Eq. (3.16), i.e.,

$$h[n] = h_c(nT), \tag{3.50}$$

Eq. (3.20) becomes

$$H(e^{j\omega}) = \frac{1}{T}\sum_{k=-\infty}^{\infty} H_c\left(j\frac{\omega}{T} - j\frac{2\pi k}{T}\right) \tag{3.51}$$

or, equivalently,

$$H(e^{j\omega}) = \frac{1}{T}H_c\left(j\frac{\omega}{T}\right), \qquad |\omega| < \pi. \tag{3.52}$$

Modifying Eqs. (3.50) and (3.52) to account for the scale factor of T in Eq. (3.49), we have

$$h[n] = Th_c(nT), \tag{3.53}$$

$$H(e^{j\omega}) = H_c\left(j\frac{\omega}{T}\right), \qquad |\omega| < \pi. \tag{3.54}$$

3.5 CONTINUOUS-TIME PROCESSING OF DISCRETE-TIME SIGNALS

In Section 3.4 we discussed and analyzed the use of discrete-time systems for processing continuous-time signals in the configuration of Fig. 3.9. In this section we consider the complementary situation depicted in Fig. 3.14, which is appropriately referred to as continuous-time processing of discrete-time signals. While the system of Fig. 3.14 is not typically used to implement discrete-time systems, it provides a useful interpretation of certain discrete-time systems.

From the definition of the ideal D/C converter, $X_c(j\Omega)$ and therefore also $Y_c(j\Omega)$ will necessarily be zero for $|\Omega| \geq \pi/T$. Thus the C/D converter samples $y_c(t)$ without aliasing, and we can express $x_c(t)$ and $y_c(t)$ as

$$x_c(t) = \sum_{n=-\infty}^{\infty} x[n]\frac{\sin[\pi(t-nT)/T]}{\pi(t-nT)/T} \tag{3.55}$$

and

$$y_c(t) = \sum_{n=-\infty}^{\infty} y[n]\frac{\sin[\pi(t-nT)/T]}{\pi(t-nT)/T}, \tag{3.56}$$

where $x[n] = x_c(nT)$ and $y[n] = y_c(nT)$. The frequency-domain relationships for Fig. 3.14 are

$$X_c(j\Omega) = TX(e^{j\Omega T}), \qquad |\Omega| < \pi/T, \tag{3.57a}$$

$$Y_c(j\Omega) = H_c(j\Omega)X_c(j\Omega), \qquad |\Omega| < \pi/T, \tag{3.57b}$$

$$Y(e^{j\omega}) = \frac{1}{T}Y_c(j\omega/T), \qquad |\omega| < \pi. \tag{3.57c}$$

Therefore, by substituting Eqs. (3.57a) and (3.57b) into Eq. (3.57c) it follows that the overall system behaves as a discrete-time system whose frequency response is

$$H(e^{j\omega}) = H_c(j\omega/T), \qquad |\omega| < \pi, \tag{3.58}$$

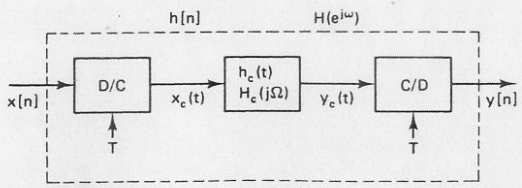


Figure 3.14 Continuous-time processing of discrete-time signals.

or, equivalently, the overall frequency response of the system in Fig. 3.14 will be equal to a given $H(e^{j\omega})$ if the frequency response of the continuous-time system is

$$H_c(j\Omega) = H(e^{j\Omega T}), \qquad |\Omega| < \pi/T. \tag{3.59}$$

Since $X_c(j\Omega) = 0$ for $|\Omega| \geq \pi/T$, $H_c(j\Omega)$ may be chosen arbitrarily above π/T. A convenient, but arbitrary, choice is $H_c(j\Omega) = 0$ for $|\Omega| \geq \pi/T$.

With this representation of a discrete-time system we can focus on the equivalent effect of the continuous-time system on the bandlimited continuous-time signal $x_c(t)$. This is illustrated in Examples 3.3 and 3.4.

Example 3.3 Noninteger Delay

Let us consider a discrete-time system with frequency response $H(e^{j\omega})$ given by

$$H(e^{j\omega}) = e^{-j\omega\Delta}, \qquad |\omega| < \pi. \tag{3.60}$$

When Δ is an integer, this system has a straightforward interpretation as a delay of Δ, i.e.,

$$y[n] = x[n - \Delta]. \tag{3.61}$$

When Δ is not an integer, Eq. (3.61) has no formal meaning. However, using the system of Fig. 3.14, a useful time-domain interpretation can be applied to the system specified by Eq. (3.60). Consider $H_c(j\Omega)$ in Fig. 3.14 chosen as

$$H_c(j\Omega) = H(e^{j\Omega T}) = e^{-j\Omega\Delta T}. \tag{3.62}$$

Then from Eq. (3.59), the overall discrete-time system in Fig. 3.14 will have the frequency response given by Eq. (3.60), whether or not Δ is an integer. To interpret the system of Eq. (3.60), we note that from Eq. (3.62),

$$y_c(t) = x_c(t - \Delta T). \tag{3.63}$$

Furthermore, $x_c(t)$ is the bandlimited interpolation of $x[n]$, and $y[n]$ is $y_c(t)$ sampled. Thus, for example, if $\Delta = \frac{1}{2}$, $y[n]$ would be the values of the bandlimited interpolation halfway between the input sequence values. This is illustrated in Fig. 3.15. Formally, from Eqs. (3.56), (3.63), and (3.55a),

$$y[n] = x_c(nT - \Delta T) = \sum_{k=-\infty}^{\infty} x[k] \left. \frac{\sin[\pi(t - \Delta T - kT)/T]}{\pi(t - \Delta T - kT)/T} \right|_{t=nT}. \tag{3.64}$$

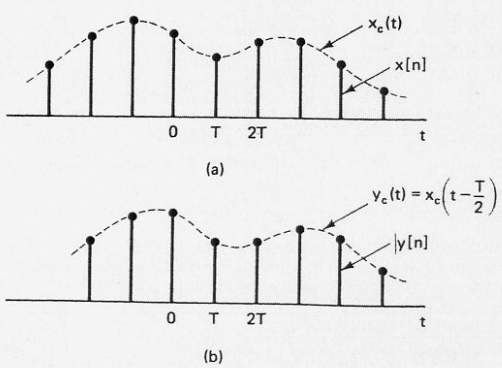


Figure 3.15 Continuous-time processing of the discrete-time sequence in part (a) can produce a new sequence with a "half-sample" delay, as in part (b).

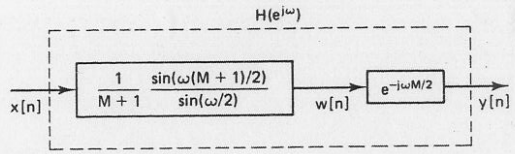


Figure 3.16 The moving average system represented as a cascade of two systems.

The noninteger delay represented by Eq. (3.64) has considerable practical significance since such a factor often arises in the frequency-domain representation of systems. When such a term is found in the frequency response of a causal discrete-time system, it can be interpreted in the light of this example. This interpretation is illustrated in Example 3.4.

Example 3.4 Moving Average

In Example 2.15 we considered the moving average system and obtained its frequency response. For the case $M_1 = 0$ and $M_2 = M$, the frequency response is

$$H(e^{j\omega}) = \frac{1}{(M+1)} \frac{\sin[\omega(M+1)/2]}{\sin(\omega/2)} e^{-j\omega M/2}, \qquad |\omega| < \pi. \tag{3.65}$$

Let us interpret this as the cascade of two systems, as indicated in Fig. 3.16. The first system imposes a frequency-domain amplitude weighting. The second system represents the linear phase term in Eq. (3.65). If M is an even integer (meaning the moving average of an odd number of samples), then the linear phase term corresponds to an integer delay, i.e.,

$$y[n] = w[n - M/2]. \tag{3.66}$$

However, if M is odd, the linear phase term corresponds to a noninteger delay, specifically, an integer plus one-half sample interval. This noninteger delay can be interpreted in terms of the discussion in Example 3.3; i.e., $y[n]$ is equivalent to bandlimited interpolation of $w[n]$ followed by a continuous-time delay of $MT/2$ (where T is the assumed but arbitrary sampling period associated with the D/C interpolation of $w[n]$) followed by C/D conversion again with sampling period T.

3.6 CHANGING THE SAMPLING RATE USING DISCRETE-TIME PROCESSING

We have seen that a continuous-time signal $x_c(t)$ can be represented by a discrete-time signal consisting of a sequence of samples

$$x[n] = x_c(nT). \tag{3.67}$$

Alternatively, our previous discussion has shown that even if $x[n]$ was not obtained originally by sampling, we can always use Eq. (3.24) to find a continuous-time bandlimited signal $x_r(t)$ whose samples are $x[n] = x_c(nT)$.

It is often necessary to change the sampling rate of a discrete-time signal, i.e., to obtain a new discrete-time representation of the underlying continuous-time signal of the form

$$x'[n] = x_c(nT'), \tag{3.68}$$

where $T' \neq T$. One approach to obtaining the sequences $x'[n]$ from $x[n]$ is to reconstruct $x_c(t)$ from $x[n]$ using Eq. (3.24) and then resample $x_c(t)$ with period T' to obtain $x'[n]$. Often this is not a desirable approach because of the nonideal analog reconstruction filter, D/A converter, and A/D converter that would be used in a practical implementation. Thus, it is of interest to consider methods of changing the sampling rate that involve only discrete-time operations.

3.6.1 Sampling Rate Reduction by an Integer Factor

The sampling rate of a sequence can be reduced by "sampling" it, i.e., by defining a new sequence

$$x_d[n] = x[nM] = x_c(nMT). \tag{3.69}$$

Equation (3.69) defines the system depicted in Fig. 3.17, which is called a *sampling rate compressor* (Crochiere and Rabiner, 1983) or simply a *compressor*. From Eq. (3.69) it is clear that $x_d[n]$ could be obtained directly from $x_c(t)$ by sampling with period $T' = MT$. Furthermore, if $X_c(j\Omega) = 0$ for $|\Omega| > \Omega_N$, then $x_d[n]$ is an exact representation of $x_c(t)$ if $\pi/T' = \pi/(MT) > \Omega_N$. That is, the sampling rate can be reduced by a factor of M without aliasing if the original sampling rate was at least M times the Nyquist rate or if the bandwidth of the sequence is first reduced by a factor of M by discrete-time filtering. In general, the operation of reducing the sampling rate (including any prefiltering) will be called *downsampling*.

As in the case of sampling a continuous-time signal, it is useful to obtain a frequency-domain relation between the input and output of the compressor. This time, however, it will be a relationship between discrete-time Fourier transforms. Although several methods can be used to derive the desired result, we will base our derivation on the results already obtained for sampling continuous-time signals. First recall that the discrete-time Fourier transform of $x[n] = x_c(nT)$ is

$$X(e^{j\omega}) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c\left(j\frac{\omega}{T} - j\frac{2\pi k}{T}\right). \tag{3.70}$$

Similarly, the discrete-time Fourier transform of $x_d[n] = x[nM] = x_c(nT')$ with $T' = MT$ is

$$X_d(e^{j\omega}) = \frac{1}{T'}\sum_{r=-\infty}^{\infty} X_c\left(j\frac{\omega}{T'} - j\frac{2\pi r}{T'}\right). \tag{3.71}$$

Now, since $T' = MT$, we can write Eq. (3.71) as

$$X_d(e^{j\omega}) = \frac{1}{MT}\sum_{r=-\infty}^{\infty} X_c\left(j\frac{\omega}{MT} - j\frac{2\pi r}{MT}\right). \tag{3.72}$$

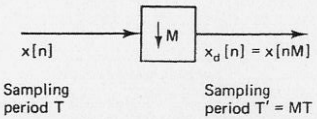


Figure 3.17 Representation of downsampler or discrete-time sampler.

To see the relationship between Eqs. (3.72) and (3.70), note that the summation index r in Eq. (3.72) can be expressed as

$$r = i + kM, \tag{3.73}$$

where k and i are integers such that $-\infty < k < \infty$ and $0 \le i \le M-1$. Clearly r is still an integer ranging from $-\infty$ to ∞, but now Eq. (3.72) can be expressed as

$$X_d(e^{j\omega}) = \frac{1}{M}\sum_{i=0}^{M-1}\left[\frac{1}{T}\sum_{k=-\infty}^{\infty} X_c\left(j\frac{\omega}{MT} - j\frac{2\pi k}{T} - j\frac{2\pi i}{MT}\right)\right]. \tag{3.74}$$

The term inside the square brackets in Eq. (3.74) is recognized from Eq. (3.70) as

$$X(e^{j(\omega-2\pi i)/M}) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_c\left(j\frac{\omega - 2\pi i}{MT} - j\frac{2\pi k}{T}\right). \tag{3.75}$$

Thus we can express Eq. (3.74) as

$$X_d(e^{j\omega}) = \frac{1}{M}\sum_{i=0}^{M-1} X(e^{j(\omega/M - 2\pi i/M)}). \tag{3.76}$$

The analogy between Eqs. (3.70) and (3.76) is clear. Equation (3.70) expresses the Fourier transform of the sequence of samples $x[n]$ (period T) in terms of the Fourier transform of the continuous-time signal $x_c(t)$. Equation (3.76) expresses the Fourier transform of the discrete-time sampled sequence $x_d[n]$ (sampling period M) in terms of the Fourier transform of the sequence $x[n]$. If we compare Eqs. (3.71) and (3.76) we see that $X_d(e^{j\omega})$ can be thought of as being composed of either an infinite set of copies of $X_c(j\Omega)$, frequency scaled through $\omega = \Omega T'$ and shifted by integer multiples of $2\pi/T'$ (Eq. 3.71), or M copies of the periodic Fourier transform $X(e^{j\omega})$, frequency scaled by M and shifted by integer multiples of $2\pi/M$ (Eq. 3.76). Either interpretation makes it clear that $X_d(e^{j\omega})$ is periodic with period 2π (as are all discrete-time Fourier transforms) and that aliasing can be avoided by ensuring that $X(e^{j\omega})$ is bandlimited, i.e.,

$$X(e^{j\omega}) = 0, \qquad \omega_N \le |\omega| \le \pi, \tag{3.77}$$

and $2\pi/M \ge 2\omega_N$.

Downsampling is illustrated in Fig. 3.18. Figure 3.18(a) shows the Fourier transform of a bandlimited continuous-time signal, and Fig. 3.18(b) shows the Fourier transform of the impulse train of samples obtained with sampling period T. Figure 3.18(c) shows $X(e^{j\omega})$ and is related to Fig. 3.18(b) through Eq. (3.18). As we have already seen, Figs. 3.18(b) and (c) differ only in a relabeling of the frequency axis. Figure 3.18(d) shows the discrete-time Fourier transform of the downsampled sequence when $M = 2$. We have plotted this Fourier transform as a function of normalized frequency $\omega = \Omega T'$. Finally, Fig. 3.18(e) shows the discrete-time Fourier transform of the downsampled sequence plotted as a function of the continuous-time

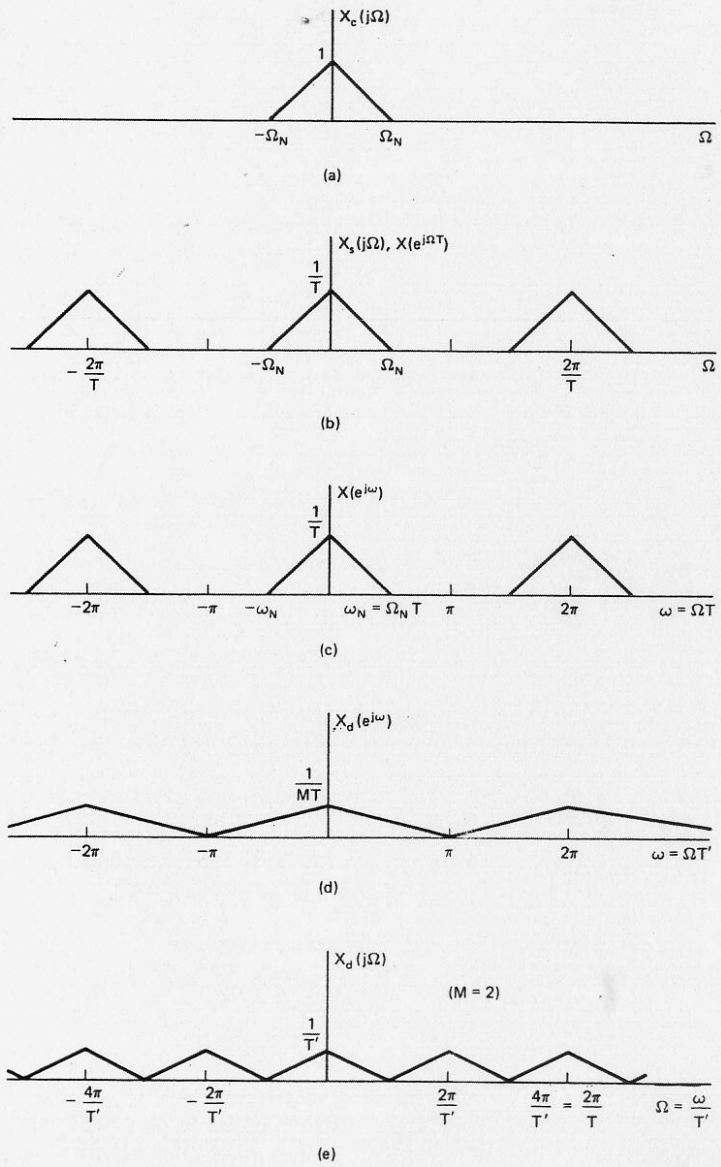


Figure 3.18 Frequency-domain illustration of downsampling.

frequency variable Ω. Figure 3.18(e) is identical to Fig. 3.18(d) except for the relabeling of the frequency axis through the relation $\Omega = \omega/T'$.

In this example, $2\pi/T = 4\Omega_N$, i.e., the original sampling rate is exactly twice the minimum rate to avoid aliasing. Thus, when the original sampled sequence is downsampled by a factor of $M = 2$, no aliasing results. If the downsampling factor is more than 2 in this case, aliasing will result, as illustrated in Fig. 3.19.

Figure 3.19(a) shows the continuous-time Fourier transform of $x_c(t)$, and Fig. 3.19(b) shows the discrete-time Fourier transform of the sequence $x[n] = x_c(nT)$, when $2\pi/T = 4\Omega_N$. Thus, $\omega_N = \Omega_N T = \pi/2$. Now if we downsample by a factor of $M = 3$, we obtain the sequence $x_d[n] = x[3n] = x_c(n3T)$ whose discrete-time Fourier transform is plotted in Fig. 3.19(c) with normalized frequency $\omega = \Omega T'$. Note that because $M\omega_N = 3\pi/2$, which is greater than π, aliasing occurs. In general, to avoid aliasing in downsampling by a factor of M requires that

$$\omega_N M < \pi \qquad \text{or} \qquad \omega_N < \pi/M. \tag{3.78}$$

If this condition does not hold, aliasing occurs, but it may be tolerable for some applications. In other cases, downsampling can be done without aliasing if we are willing to reduce the bandwidth of the signal $x[n]$ before downsampling. Thus, if $x[n]$ is filtered by an ideal lowpass filter with cutoff frequency π/M, then the output $\tilde{x}[n]$ can be decimated without aliasing, as illustrated in Figs. 3.19(d), (e), and (f). Note that the sequence $\tilde{x}_d[n] = \tilde{x}[nM]$ no longer represents the original underlying continuous-time signal $x_c(t)$. Rather $\tilde{x}_d[n] = \tilde{x}_c(nT')$, where $T' = MT$, and $\tilde{x}_c(t)$ is obtained from $x_c(t)$ by lowpass filtering with cutoff frequency $\Omega_c = \pi/T' = \pi/(MT)$.

From the preceding discussion we see that a general system for downsampling by a factor of M is the one shown in Fig. 3.20. Such a system is called a *decimator*, and downsampling by lowpass filtering followed by compression has been termed *decimation* (Crochiere and Rabiner, 1983).

3.6.2 Increasing the Sampling Rate by an Integer Factor

We have seen that reduction of the sampling rate of a discrete-time signal by an integer factor involves sampling the sequence in a manner analogous to sampling a continuous-time signal. Not surprisingly, increasing the sampling rate involves operations analogous to D/C conversion. To see this, consider a signal $x[n]$ whose sampling rate we wish to increase by a factor of L. If we consider the underlying continuous-time signal $x_c(t)$, the objective is to obtain samples

$$x_i[n] = x_c(nT'), \tag{3.79}$$

where $T' = T/L$, from the sequence of samples

$$x[n] = x_c(nT). \tag{3.80}$$

We will refer to the operation of increasing the sampling rate as *upsampling*.

It is clear from Eqs. (3.79) and (3.80) that

$$x_i[n] = x[n/L] = x_c(nT/L), \qquad n = 0, \pm L, \pm 2L, \ldots \tag{3.81}$$

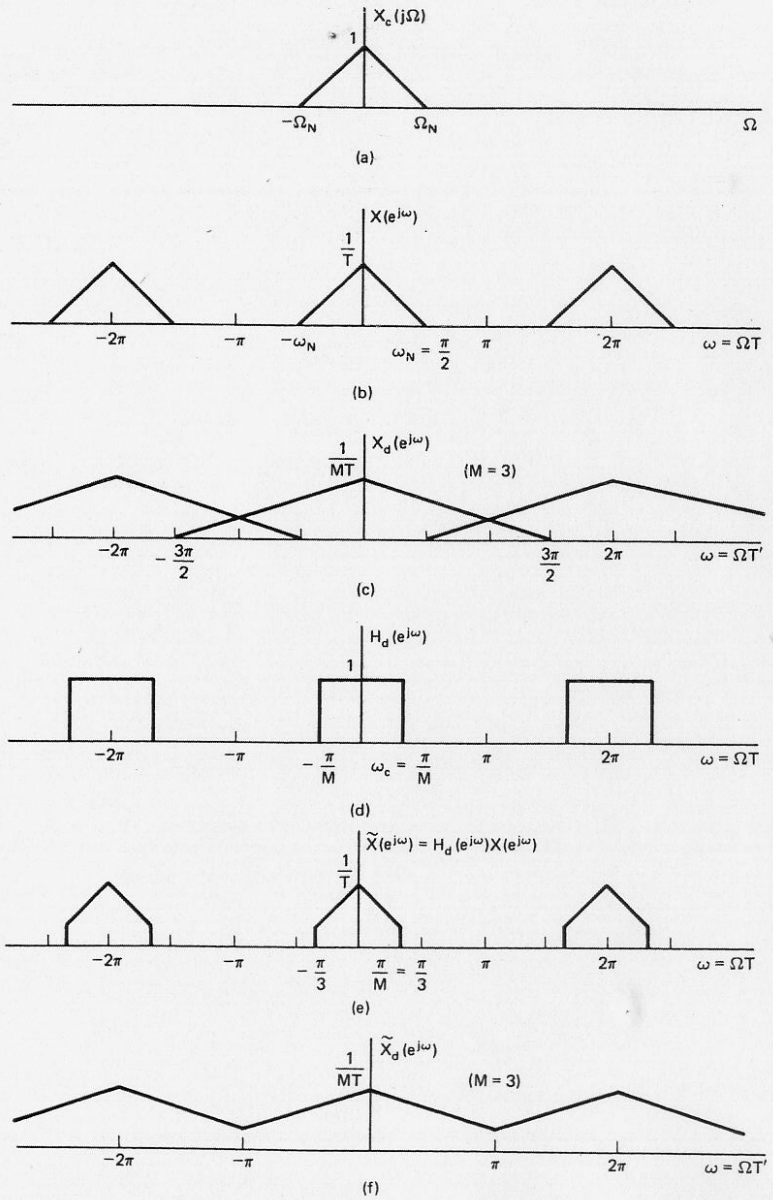


Figure 3.19 (a)–(c) Downsampling with aliasing. (d)–(f) Downsampling with prefiltering to avoid aliasing.

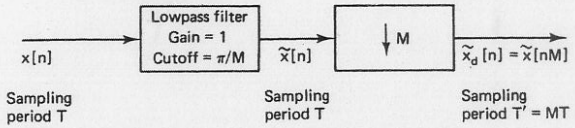


Figure 3.20 General system for sampling rate reduction by M.

Figure 3.21 shows a system for obtaining $x_i[n]$ from $x[n]$ using only discrete-time processing. The system on the left is called a *sampling rate expander* (Crochiere and Rabiner, 1983) or simply an *expander*. Its output is

$$x_e[n] = \begin{cases} x[n/L], & n = 0, \pm L, \pm 2L, \ldots, \\ 0, & \text{otherwise}, \end{cases} \tag{3.82}$$

or, equivalently,

$$x_e[n] = \sum_{k=-\infty}^{\infty} x[k]\delta[n - kL]. \tag{3.83}$$

The system on the right is a lowpass discrete-time filter with cutoff frequency π/L and gain L. This system plays a role similar to the ideal D/C converter in Fig. 3.8(b). First we create a discrete-time impulse train $x_e[n]$ and then we lowpass filter to reconstruct the sequence.

The operation of the system in Fig. 3.21 is most easily understood in the frequency domain. The Fourier transform of $x_e[n]$ can be expressed as

$$\begin{aligned} X_e(e^{j\omega}) &= \sum_{n=-\infty}^{\infty} \left(\sum_{k=-\infty}^{\infty} x[k]\delta[n - kL] \right) e^{-j\omega n} \\ &= \sum_{k=-\infty}^{\infty} x[k]e^{-j\omega Lk} = X(e^{j\omega L}). \end{aligned} \tag{3.84}$$

Thus the Fourier transform of the output of the expander is a frequency-scaled version of the Fourier transform of the input, i.e., ω is replaced by ωL so that ω is now normalized by

$$\omega = \Omega T'. \tag{3.85}$$

This effect is illustrated in Fig. 3.22. Figure 3.22(a) shows a bandlimited continuous-time Fourier transform, and Fig. 3.22(b) shows the discrete-time Fourier transform of the sequence $x[n] = x_c(nT)$, where $\pi/T = \Omega_N$. Figure 3.22(c) shows $X_e(e^{j\omega})$ according to Eq. (3.84), with $L = 2$, and Fig. 3.22(e) shows the Fourier transform of the desired signal $x_i[n]$. We see that $X_i(e^{j\omega})$ can be obtained from $X_e(e^{j\omega})$ by correcting the amplitude scale from $1/T$ to $1/T'$ and by removing all the frequency-scaled images of $X_c(j\Omega)$ except at integer multiples of 2π. For the case depicted in Fig. 3.22 this requires a lowpass filter with a gain of 2 and cutoff frequency $\pi/2$, as shown in Fig.

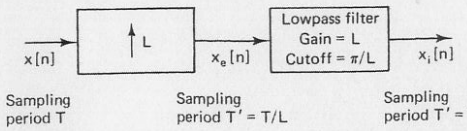


Figure 3.21′ General system for sampling rate increase by L.

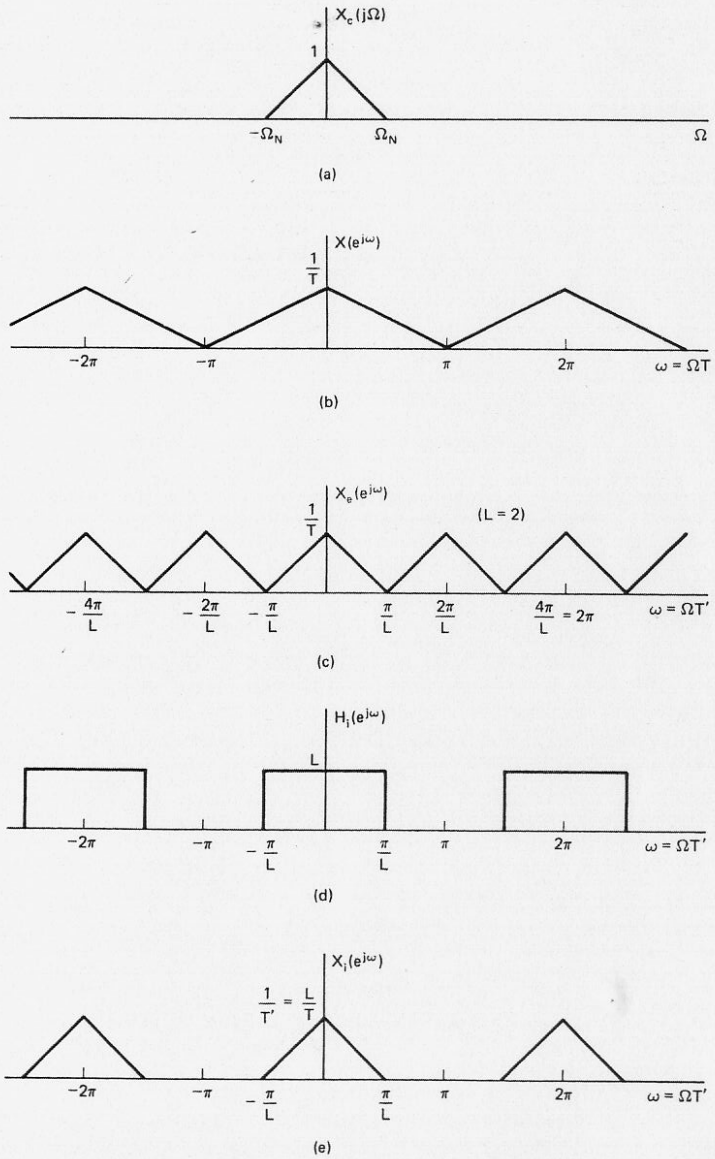


Figure 3.22 Frequency-domain illustration of interpolation.

3.22(d). In general, the required gain would be L, since $L(1/T) = [1/(T/L)] = 1/T'$, and the cutoff frequency would be π/L.

This example shows that the system of Fig. 3.21 does indeed give an output satisfying Eq. (3.79) if the input sequence $x[n] = x_c(nT)$ was obtained by sampling without aliasing. That system is therefore called an *interpolator* since it fills in the missing samples, and the operation of upsampling is therefore considered to be synonymous with *interpolation*.

As in the case of the D/C converter, it is possible to obtain an interpolation formula for $x_i[n]$ in terms of $x[n]$. First note that the impulse response of the lowpass filter in Fig. 3.21 is

$$h_i[n] = \frac{\sin(\pi n/L)}{\pi n/L}. \tag{3.86}$$

Using Eq. (3.83), we obtain

$$x_i[n] = \sum_{k=-\infty}^{\infty} x[k] \frac{\sin[\pi(n-kL)/L]}{\pi(n-kL)/L}. \tag{3.87}$$

The impulse response $h_i[n]$ has the properties

$$\begin{aligned} h_i[0] &= 1, \\ h_i[n] &= 0, \qquad n = \pm L, \pm 2L, \ldots. \end{aligned} \tag{3.88}$$

Thus for the ideal lowpass interpolation filter we have

$$x_i[n] = x[n/L] = x_c(nT/L) = x_c(nT'), \qquad n = 0, \pm L, \pm 2L, \ldots, \tag{3.89}$$

as desired. The fact that $x_i[n] = x_c(nT')$ for all n follows from our frequency-domain argument.

In practice, ideal lowpass filters cannot be implemented exactly, but we will see in Chapter 7 that very good approximations can be designed. (Also see Schafer and Rabiner, 1973, and Oetken et al., 1975.) In some cases very simple interpolation procedures are adequate. Since linear interpolation is often used, although generally it is not very accurate, it is worthwhile to examine linear interpolation in the present context.

Linear interpolation can be accomplished by the system of Fig. 3.21 if the filter has impulse response

$$h_{lin}[n] = \begin{cases} 1 - |n|/L, & |n| \le L, \\ 0, & \text{otherwise}, \end{cases} \tag{3.90}$$

as shown in Fig. 3.23 for $L = 5$. With this filter, the interpolated output will be

$$x_{lin}[n] = \sum_{k=-\infty}^{\infty} x_e[k] h_{lin}[n-k] = \sum_{k=-\infty}^{\infty} x[k] h_{lin}[n-kL]. \tag{3.91}$$

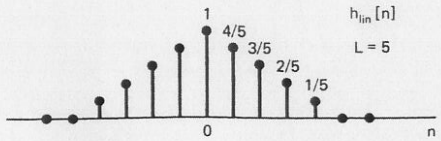


Figure 3.23 Impulse response for linear interpolation.

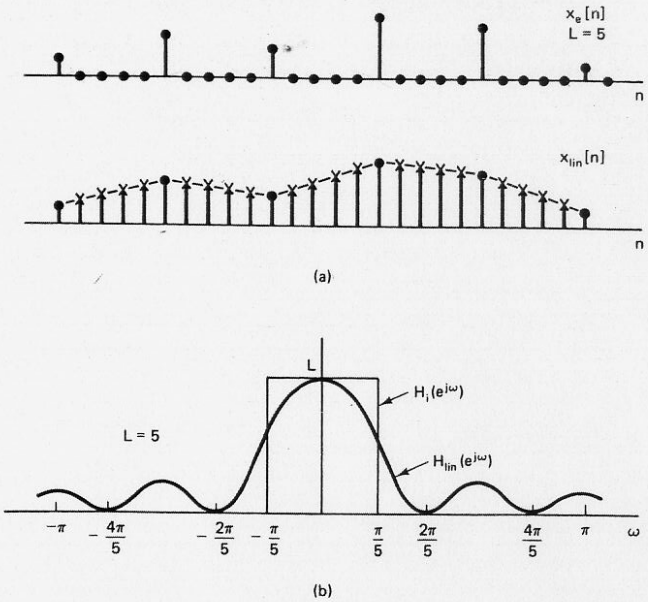


Figure 3.24 (a) Illustration of linear interpolation by filtering. (b) Frequency response of linear interpolator compared with ideal lowpass interpolation filter.

Figure 3.24(a) depicts $x_e[n]$ and $x_{lin}[n]$ for the case $L = 5$. From this figure we see that $x_{lin}[n]$ is identical to the sequence obtained by linear interpolation between the samples. Note that

$$h_{lin}[0] = 1, \qquad h_{lin}[n] = 0, \quad n = \pm L, \pm 2L, \ldots, \tag{3.92}$$

so that

$$x_{lin}[n] = x[n/L] \quad \text{at} \quad n = 0, \pm L, \pm 2L, \ldots. \tag{3.93}$$

The amount of distortion in the intervening samples can be gauged by comparing the frequency response of the linear interpolator to the ideal lowpass interpolator for a factor-of-L interpolation. It can be shown (see Problem 3.22) that

$$H_{lin}(e^{j\omega}) = \frac{1}{L}\left[\frac{\sin(\omega L/2)}{\sin(\omega/2)}\right]^2. \tag{3.94}$$

This function is plotted in Fig. 3.24(b) for $L = 5$ together with the ideal lowpass interpolation filter. It is clear from this figure that if the original signal is sampled at the Nyquist rate, linear interpolation will not be very good since the output of the filter will contain considerable energy in the band $\pi/L < |\omega| \le \pi$. However, if the original sampling rate is much higher than the Nyquist rate, then the linear

interpolator will be more successful in removing the frequency-scaled images of $X_c(j\Omega)$ at multiples of $2\pi/L$. This is intuitively reasonable since if the original sampling rate greatly exceeds the Nyquist rate, the signal will not vary significantly between samples, and thus linear interpolation should be reasonably accurate.

3.6.3 Changing the Sampling Rate by a Noninteger Factor

We have shown how to increase or decrease the sampling rate of a sequence by an integer factor. By combining decimation and interpolation it is possible to change the sampling rate by a noninteger factor. Specifically, consider Fig. 3.25(a), which shows an interpolator that decreases the sampling period from T to T/L, followed by a decimator, which increases the sampling period by M, producing an output sequence $\tilde{x}_d[n]$ that has an effective sampling period of $T' = TM/L$. By choosing L and M appropriately, we can approach arbitrarily close to any desired ratio of sampling periods. For example, if $L = 100$ and $M = 101$, then $T' = 1.01T$.

If $M > L$ there is a net increase in the sampling period (decrease in sampling rate) and if $M < L$, the opposite is true. From Fig. 3.25(a) we see that the interpolation and decimation filters are in cascade, so that they can be combined as shown in Fig. 3.25(b) into one lowpass filter with gain L and cutoff equal to the minimum of π/L and π/M. If $M > L$, then π/M is the dominant cutoff frequency and there is a net reduction in sampling rate. As pointed out in Section 3.6.1, if $x[n]$ was obtained by sampling at the Nyquist rate, the sequence $\tilde{x}_d[n]$ will represent a lowpass-filtered version of the original underlying bandlimited signal if we are to

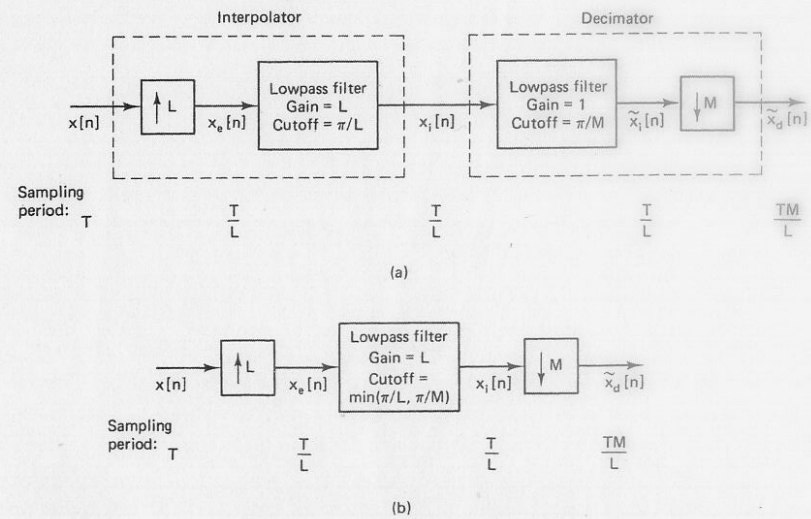


Figure 3.25 (a) System for changing the sampling rate by a noninteger factor. (b) Simplified system in which the decimation and interpolation filters are combined.

avoid aliasing. On the other hand, if $M < L$, then π/L is the dominant cutoff frequency and there will be no need to further limit the bandwidth of the signal below the original Nyquist frequency.

3.7 PRACTICAL CONSIDERATIONS

So far in this chapter our discussions of the representation of continuous-time signals by discrete-time signals have focused on idealized models of periodic sampling and bandlimited interpolation. We have formalized our discussions in terms of an idealized sampling system that we have called the *ideal continuous-to-discrete (C/D) converter* and an idealized bandlimited interpolator system called the *ideal discrete-to-continuous (D/C) converter*. These idealized conversion systems allow us to concentrate on the essential mathematical details of the relationship between a bandlimited signal and its samples. For example, in Section 3.4 we used the idealized C/D and D/C conversion systems to show that linear time-invariant discrete-time systems can be used in the configuration of Fig. 3.26(a) to implement linear time-invariant continuous-time systems if the input is bandlimited and the sampling rate exceeds the Nyquist rate. In a practical setting, continuous-time signals are not precisely bandlimited, ideal filters cannot be realized, and the ideal C/D and D/C converters can only be approximated. The block diagram of Fig. 3.26(b) shows a more realistic model for digital processing of continuous-time (analog) signals. In this section we will examine some of the imperfections introduced by each of the components of the system in Fig. 3.26(b). While it is difficult to carry out an exact analysis, our approximate analysis will indicate the effects of each imperfection and suggest that the system of Fig. 3.26(b) can very closely approximate the system of Fig. 3.26(a).

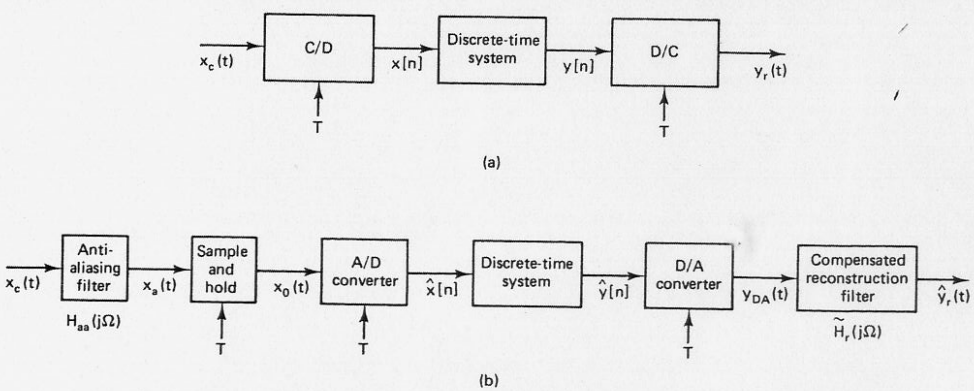


Figure 3.26 (a) Discrete-time filtering of continuous-time signals. (b) A more realistic model.

3.7.1 Prefiltering to Avoid Aliasing

In most cases, it is desirable to minimize the sampling rate of a discrete-time system for processing analog systems. This is because the amount of arithmetic processing required to implement the system is proportional to the number of samples to be processed. If the input is not bandlimited or if the Nyquist frequency of the input is too high, prefiltering is often used. An example of such a situation occurs in processing speech signals, where often only the low-frequency band up to about 3–4 kHz is required for intelligibility even though the speech signal may have significant frequency content in the 4–20 kHz range. Also, even if the signal is naturally bandlimited, wideband additive noise may fill in the higher-frequency range, and as a result of sampling, these noise components would be aliased into the low-frequency band. If we wish to avoid aliasing, the input signal must be forced to be bandlimited to frequencies below one-half the desired sampling rate. This can be accomplished by lowpass filtering the continuous-time signal prior to C/D conversion, as shown in Fig. 3.27. In this context the lowpass filter that precedes the C/D converter is called an *anti-aliasing filter*. Ideally, the frequency response of the anti-aliasing filter would be

$$H_{aa}(j\Omega) = \begin{cases} 1, & |\Omega| < \Omega_c < \pi/T, \\ 0, & |\Omega| > \Omega_c. \end{cases} \tag{3.95}$$

From the discussion of Section 3.4.1, it follows that the overall system from the output of the anti-aliasing filter $x_a(t)$ to the output $y_r(t)$ will always behave as a linear time-invariant system, since the input to the C/D converter $x_a(t)$ is forced by the anti-aliasing filter to be bandlimited to frequencies below π/T radians/s. Thus the overall effective frequency response of Fig. 3.27 will be the product of $H_{aa}(j\Omega)$ and the effective frequency response from $x_a(t)$ to $y_r(t)$. Combining Eqs. (3.95) and (3.37) gives

$$H_{eff}(j\Omega) = \begin{cases} H(e^{j\Omega T}), & |\Omega| < \pi/T, \\ 0, & |\Omega| > \pi/T. \end{cases} \tag{3.96}$$

Thus for an ideal lowpass anti-aliasing filter, the system of Fig. 3.27 behaves as a linear time-invariant system with frequency response given by Eq. (3.96) even when $X_c(j\Omega)$ is not bandlimited. In practice, the frequency response $H_{aa}(j\Omega)$ cannot be ideally bandlimited, but $H_{aa}(j\Omega)$ can be made small for $|\Omega| > \pi/T$ so that aliasing is minimized. In this case, the overall frequency response of the system in Fig. 3.27 would be approximately

$$H_{eff}(j\Omega) \approx H_{aa}(j\Omega)H(e^{j\Omega T}). \tag{3.97}$$

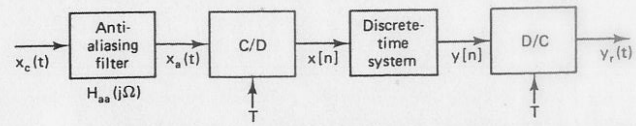


Figure 3.27 Use of prefiltering to avoid aliasing.

To achieve negligibly small frequency response above π/T it would be necessary for $H_{aa}(j\Omega)$ to begin to "roll off," i.e., begin to introduce attenuation, at frequencies below π/T. Equation (3.97) suggests that the rolloff of the anti-aliasing filter (and other linear time-invariant distortions to be discussed later) could be at least partially compensated for by taking them into account in the design of the discrete-time system. This is illustrated in Problem 3.32.

3.7.2 Analog-to-Digital (A/D) Conversion

Sampling operation

An ideal C/D converter converts a continuous-time signal into a discrete-time signal, where each sample is known with infinite precision. As an approximation to this for digital signal processing, the system of Fig. 3.28 converts a continuous-time (analog) signal into a digital signal, i.e., a sequence of finite-precision or quantized samples. The two systems in Fig. 3.28 correspond to systems that are available as physical devices. The A/D converter is a physical device that converts a voltage or current amplitude at its input into a binary code representing a quantized amplitude value closest to the amplitude of the input. Under the control of an external clock, the A/D converter can be caused to start and complete an A/D conversion every T seconds. However, the conversion is not instantaneous, and for this reason a high-performance A/D system typically includes a sample and hold as in Fig. 3.28. The ideal sample-and-hold system is the system whose output is

$$x_0(t) = \sum_{n=-\infty}^{\infty} x[n]h_0(t - nT), \tag{3.98}$$

where $x[n] = x_a(nT)$ are the ideal samples of $x_a(t)$, and $h_0(t)$ is the impulse response of the zero-order-hold system, i.e.,

$$h_0(t) = \begin{cases} 1, & 0 < t < T, \\ 0, & \text{otherwise.} \end{cases} \tag{3.99}$$

If we note that Eq. (3.98) has the equivalent form

$$x_0(t) = h_0(t) * \sum_{n=-\infty}^{\infty} x_a(nT)\delta(t - nT), \tag{3.100}$$

we see that the ideal sample and hold is equivalent to impulse train modulation followed by linear filtering with the zero-order-hold system as depicted in Fig. 3.29(a). The relationship between the Fourier transform of $x_0(t)$ and the Fourier transform of $x_a(t)$ can be worked out following the style of analysis of Section 3.2, and

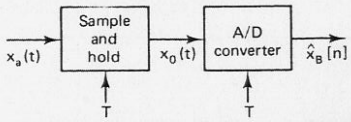


Figure 3.28 Physical configuration for analog-to-digital conversion.

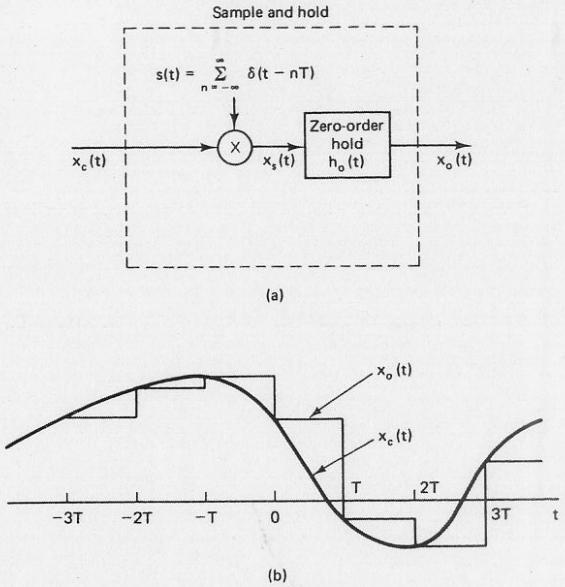


Figure 3.29 (a) Representation of an ideal sample and hold. (b) Typical input and output signals for the sample and hold.

we will do a similar analysis when we discuss the D/A converter. However, the analysis is unnecessary at this point since everything we need to know about the behavior of the system can be seen from the time-domain expression. Specifically, the output of the zero-order hold is a staircase waveform where the sample values are held constant during the sampling period of T seconds. This is illustrated in Fig. 3.29(b). Physical sample-and-hold circuits are designed to sample $x_a(t)$ as nearly instantaneously as possible and to hold the sample value as nearly constant as possible until the next sample is taken. The purpose of this is to provide the constant input voltage (or current) required by the A/D converter. The details of the wide variety of A/D conversion processes and the details of sample-and-hold and A/D circuit implementations are outside the scope of this book. Many practical issues arise in obtaining a sample and hold that samples quickly and holds the sample value constant with no decay or "glitches." Likewise, many practical concerns dictate the speed and accuracy of conversion of A/D converter circuits. Such questions are considered in Hnatek (1976) and Schmid (1976), and details of performance of a specific product are available in manufacturers' specification and data sheets. Our concern in this section is the analysis of the quantization effects in A/D conversion.

Since the purpose of the sample and hold in Fig. 3.28 is to implement ideal sampling and to hold the sample value for quantization by the A/D converter, we can represent the system of Fig. 3.28 by the system of Fig. 3.30, where the ideal C/D converter represents the sampling performed by the sample and hold, and, as we will describe later, the quantizer and coder together represent the operation of the A/D converter.

purpose of sampling hold
hold sample for T sec & constant

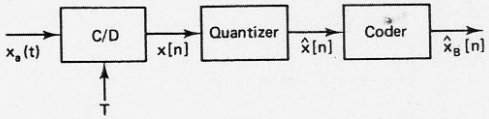


Figure 3.30 Mathematical representation of the system in Fig. 3.28.

The quantizer is a nonlinear system whose purpose is to transform the input sample $x[n]$ into one of a finite set of prescribed values. We represent this operation as

$$\hat{x}[n] = Q(x[n]) \tag{3.101}$$

and refer to $\hat{x}[n]$ as the *quantized sample*. Quantizers can be defined with either uniformly or nonuniformly spaced quantization levels; however, in the context of signal processing the quantization steps usually are uniform. Figure 3.31 shows a typical uniform quantizer characteristic† in which the sample values are *rounded* to the nearest quantization level.

Several features of Fig. 3.31 should be emphasized. First note that this quantizer would be appropriate for a signal whose samples are both positive and negative (bipolar). If it is known that the input samples are always positive (or negative), then a different distribution of the quantization levels would be appropriate. Next note that the quantizer of Fig. 3.31 has an even number of quantization levels. With an even number of levels, it is not possible to have a quantization level at zero amplitude and also have an equal number of positive and negative quantization levels. Generally the number of quantization levels will be a power of 2, but the number will be much greater than eight so this difference is usually inconsequential.

Figure 3.31 also depicts coding of the quantization levels. Since there are eight quantization levels, we can label them by a binary code of 3 bits. (In general 2^{B+1} levels can be coded with a $(B+1)$-bit binary code.) In principle, any assignment of symbols can be used, and many binary coding schemes exist, each with its own advantages and disadvantages depending on the application. For example, the right-hand column of binary numbers in Fig. 3.31 illustrates the *offset binary* coding scheme, in which the binary symbols are assigned in numeric order starting with the most negative quantization level. However, in digital signal processing, we generally wish to use a binary code that permits us to do arithmetic directly with the code words as scaled representations of the quantized samples.

The left-hand column in Fig. 3.31 shows an assignment according to the two's complement binary number system. This very convenient system for representing signed numbers is used in most computers and microprocessors and thus it is perhaps the most convenient labeling of the quantization levels. Note incidentally that the offset binary code can be converted to two's complement code by simply complementing the most significant bit.

In the two's-complement system, the leftmost or most significant bit is considered as the sign bit, and we can consider the remaining bits as representing either binary integers or fractions. We will assume the latter; i.e., we assume a binary

† Such quantizers are also called *linear* quantizers because of the linear progression of quantization steps.

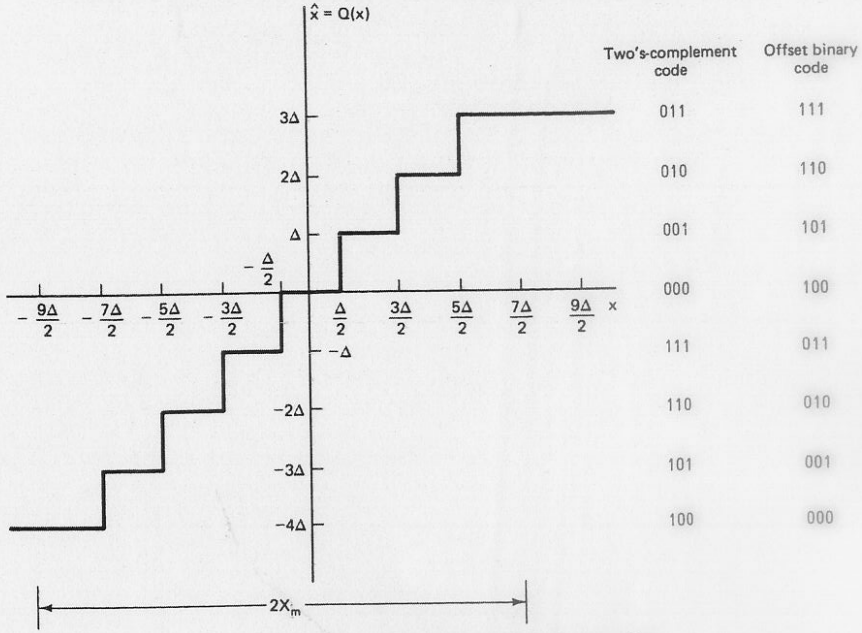


Figure 3.31 Typical quantizer for A/D conversion.

fraction point between the two most significant bits. Then for the two's-complement interpretation, the binary symbols have the following meaning for $B = 2$:

Binary Symbol	Numeric Value, $\hat{x}_B$
$0_\wedge 1\ 1$	3/4
$0_\wedge 1\ 0$	1/2
$0_\wedge 0\ 1$	1/4
$0_\wedge 0\ 0$	0
$1_\wedge 1\ 1$	−1/4
$1_\wedge 1\ 0$	−1/2
$1_\wedge 0\ 1$	−3/4
$1_\wedge 0\ 0$	−1

In general if we have a $(B+1)$-bit binary two's-complement fraction of the form

$$a_0 a_1 a_2 \ldots a_B,$$

then its value is

$$-a_0 2^0 + a_1 2^{-1} + a_2 2^{-2} + \cdots + a_B 2^{-B}.$$

The relationship between the code words and the quantized levels depends on the parameter X_m in Fig. 3.31. In general this parameter is called the full-scale level of

the A/D converter. Typical values are 10, 5, or 1 volt. From Fig. 3.31 we see that the step size Δ of the quantizer would in general be

$$\Delta = \frac{2X_m}{2^{B+1}} = \frac{X_m}{2^B}. \tag{3.102}$$

The smallest quantization levels ($\pm\Delta$) correspond to the least significant bit of the binary code word. Furthermore, the numeric relationship between the code words and the quantized samples is

$$\hat{x}[n] = X_m \hat{x}_B[n] \tag{3.103}$$

since we have assumed that $\hat{x}_B[n]$ is a binary number such that $-1 \leq \hat{x}_B[n] < 1$ (for two's complement). In this scheme the binary coded samples $\hat{x}_B[n]$ are directly proportional to the quantized samples (in two's complement binary), and therefore they can be used as a numeric representation of the amplitude of the samples. Indeed, it is generally appropriate to assume that the input signal is normalized to X_m, so that the numeric values of $\hat{x}[n]$ and $\hat{x}_B[n]$ are identical and there is no need to distinguish between the quantized samples and the binary coded samples.

Figure 3.32 shows a simple example of quantization and coding of the samples of a sine wave using a 3-bit quantizer. The unquantized samples $x[n]$ are shown with solid dots and the quantized samples $x[n]$ are shown with open circles. Also shown is

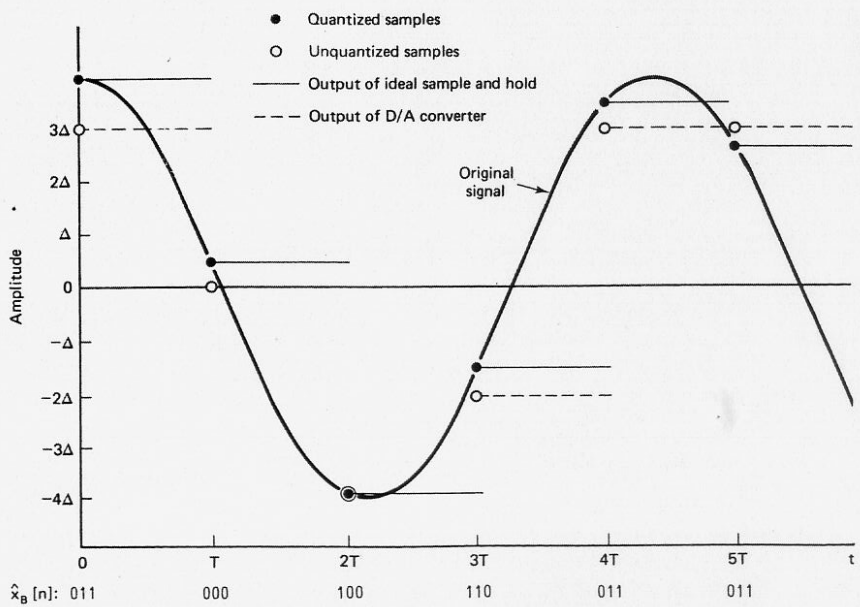


Figure 3.32 Sampling, quantization, coding, and D/A conversion with a 3-bit quantizer.

the output of an ideal sample and hold. The dotted lines labeled "output of D/A converter" will be discussed later. Figure 3.32 also shows the 3-bit code words that represent each sample. Note that since the analog input $x_a(t)$ exceeds the full-scale value of the quantizer, some of the positive samples are "clipped."

Although much of the preceding discussion pertains to two's-complement coding of the quantization levels, the basic principles of quantization and coding in A/D conversion are the same regardless of the binary code used to represent the samples. A more detailed discussion of the binary arithmetic systems used in digital computing can be found in texts on computer arithmetic (see, for example, Knuth, 1981; Cavanagh, 1984). We now turn to an analysis of the effect of quantization. Since this analysis does not depend on the assignment of binary code words, it will lead to rather general conclusions.

3.7.3 Analysis of Quantization Errors

From Figs. 3.31 and 3.32 we see that the quantized sample $\hat{x}[n]$ will in general be different from the true sample value $x[n]$. The difference between them is the *quantization error*, defined as

$$e[n] = \hat{x}[n] - x[n]. \tag{3.104}$$

For example, for the 3-bit quantizer of Fig. 3.31, if $\Delta/2 < x[n] \leq 3\Delta/2$, then $\hat{x}[n] = \Delta$, and it follows that

$$-\Delta/2 < e[n] \leq \Delta/2. \tag{3.105}$$

In the case of Fig. 3.31, Eq. (3.105) holds whenever

$$-9\Delta/2 < x[n] \leq 7\Delta/2. \tag{3.106}$$

In the general case of a $(B+1)$-bit quantizer with Δ given by Eq. (3.102), the quantization error satisfies Eq. (3.105) whenever

$$(-X_m - \Delta/2) < x[n] \leq (X_m - \Delta/2). \tag{3.107}$$

If $x[n]$ is outside this range, as it is for the sample at $t = 0$ in Fig. 3.32, then the quantization error is larger in magnitude than $\Delta/2$ and such samples are said to be clipped.

A simplified but useful model of the quantizer is depicted in Fig. 3.33. In this model the quantization error samples are thought of as an additive noise signal. This model is exactly equivalent to the quantizer if we know $e[n]$. In most cases $e[n]$ is not known, and a statistical model based on Fig. 3.33 may then be useful in representing

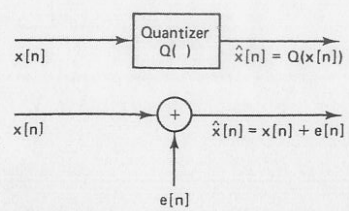


Figure 3.33 Additive noise model for quantizer.

the effects of quantization. We will also use such a model in Chapters 6 and 9 to describe the effects of quantization in signal processing algorithms. The statistical representation of quantization errors is based on the following assumptions:

1. The error sequence $e[n]$ is a sample sequence of a stationary random process.
2. The error sequence is uncorrelated with the sequence $x[n]$.
3. The random variables of the error process are uncorrelated; i.e., the error is a white-noise process.
4. The probability distribution of the error process is uniform over the range of quantization error.

As we will see, these assumptions lead to a rather simple analysis of quantization effects. It is easy to find situations where these assumptions are clearly not valid. For example, if $x_a(t)$ is a step function, the assumptions would not be justified. However, when the signal is a complicated signal, such as speech or music where the signal fluctuates rapidly in a somewhat unpredictable manner, these assumptions are more realistic. Experiments have shown that as the signal becomes more complicated, the measured correlation between the signal and the quantization error decreases, and the error also becomes uncorrelated (see Bennett, 1948; Widrow, 1956, 1961). In a heuristic sense, the assumptions of the statistical model appear to be valid if the signal is sufficiently complex and the quantization steps sufficiently small so that the amplitude of the signal is likely to traverse many quantization steps from sample to sample.

As an illustration, Fig. 3.34(a) shows a sequence of unquantized samples of a speech waveform. Figures 3.34(b) and 3.34(c) show the quantization-noise samples after quantization of the samples of Fig. 3.34(a) with an 8-bit and a 3-bit quantizer, respectively. In the 8-bit case, the quantization noise has been magnified by a factor of 32. Comparison of the noise sequences with the signal supports the preceding assertions about the quantization-noise properties in the finely quantized (8-bit) case. In Chapter 11 this will be demonstrated further when we calculate the power density spectrum and autocorrelation of a quantization noise sequence.

For quantizers that round the sample value to the nearest quantization level, as shown in Fig. 3.31, the amplitude of the quantization noise is in the range

$$-\Delta/2 < e[n] \leq \Delta/2. \tag{3.108}$$

For small Δ, it is reasonable to assume that $e[n]$ is a random variable uniformly distributed from $-\Delta/2$ to $\Delta/2$. Therefore the first-order probability density for the quantization noise is as shown in Fig. 3.35. (If truncation rather than rounding is used in implementing quantization, then the error would always be negative and we would assume a uniform probability density from $-\Delta$ to 0.) To complete the statistical model for the quantization noise, we assume that successive noise samples are uncorrelated with each other and that $e[n]$ is uncorrelated with $x[n]$. Thus $e[n]$ is assumed to be a uniformly distributed white-noise sequence. The mean value of $e[n]$ is zero and its variance is

$$\sigma_e^2 = \frac{\Delta^2}{12}. \tag{3.109}$$

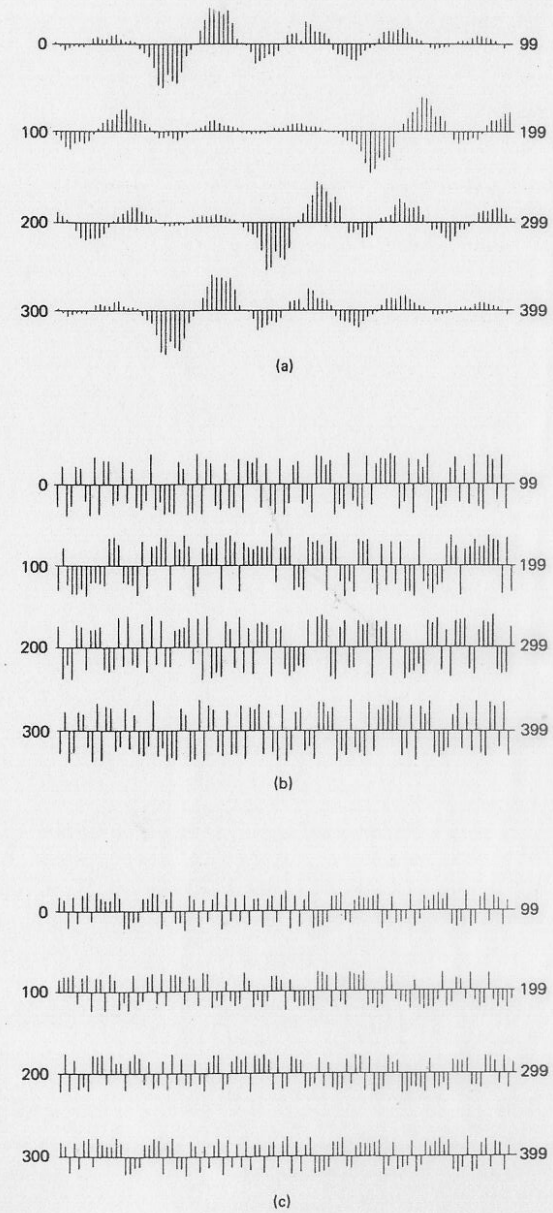


Figure 3.34 Example of quantization noise. Quantization of the speech waveform in part (a) with an 8-bit and 3-bit quantizer results in the quantization noise waveforms shown in parts (b) and (c), respectively. The quantization noise in (b) is magnified by a factor of 5 relative to (a). The quantization noise in (c) is magnified by a factor of 100 relative to (a).

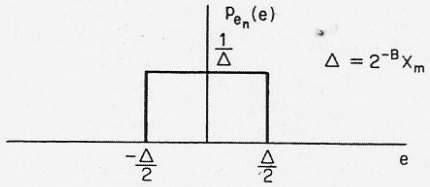


Figure 3.35 Probability density function of quantization error for a rounding quantizer such as in Fig. 3.31.

For a $(B+1)$-bit quantizer with full-scale value X_m, the noise variance or power is

$$\sigma_e^2 = \frac{2^{-2B}X_m^2}{12}. \tag{3.110}$$

A common measure of the amount of degradation of a signal by additive noise is the signal-to-noise ratio, defined as the ratio of signal variance (power) to noise variance. Expressed in decibels, the signal-to-noise ratio of a $(B+1)$-bit quantizer is

$$\begin{aligned} \text{SNR} &= 10\log_{10}\left(\frac{\sigma_x^2}{\sigma_e^2}\right) = 10\log_{10}\left(\frac{12\cdot 2^{2B}\sigma_x^2}{X_m^2}\right) \\ &= 6.02B + 10.8 - 20\log_{10}\left(\frac{X_m}{\sigma_x}\right). \end{aligned} \tag{3.111}$$

From Eq. (3.111) we see that the signal-to-noise ratio increases approximately 6 dB for each bit added to the wordlength of the quantized samples, i.e., for each doubling of the number of quantization levels. It is particularly instructive to consider the term

$$-20\log_{10}\left(\frac{X_m}{\sigma_x}\right) \tag{3.112}$$

in Eq. (3.111). First recall that X_m is a parameter of the quantizer and it would usually be fixed in a practical system. The quantity σ_x is the rms value of the signal amplitude, and it would necessarily be less than the peak amplitude of the signal. For example, if $x_a(t)$ is a sine wave of peak amplitude X_p, then $\sigma_x = X_p/\sqrt{2}$. If σ_x is too large, the peak signal amplitude will exceed the full-scale amplitude X_m of the A/D converter. In this case Eq. (3.111) is no longer valid and severe distortion results. If, on the other hand, σ_x is too small, then the term in Eq. (3.112) will become large and negative, thereby decreasing the signal-to-noise ratio in Eq. (3.111). In fact, it is easily seen that when σ_x is halved, SNR decreases by 6 dB. Thus it is very important that the signal amplitude be carefully matched to the full-scale amplitude of the A/D converter.

For analog signals such as speech or music, the distribution of amplitudes tends to be concentrated about zero and falls off rapidly with increasing amplitude. In such cases, the probability that the magnitude of a sample will exceed three or four times the rms value is very low. For example, if the signal amplitude has a Gaussian distribution, only 0.064 percent of the samples would have an amplitude greater than $4\sigma_x$. Thus, to avoid peak clipping (as assumed in our statistical model), we might set

the gain of filters and amplifiers preceding the A/D converter so that $\sigma_x = X_m/4$. Using this value of σ_x in Eq. (3.111) gives

$$\text{SNR} \approx 6B - 1.25 \text{ dB}. \tag{3.113}$$

For example, obtaining a signal-to-noise ratio of about 90–96 dB for use in high-quality music recording and playback requires 16-bit quantization, but it should be remembered that such performance is obtained only if the input signal is carefully matched to the full-scale range of the A/D converter.

This tradeoff between peak signal amplitude and absolute size of the quantization noise is fundamental to any quantization process. We will see its importance again in Chapter 6 when we discuss roundoff noise in implementing discrete-time linear systems.

3.7.4 D/A Conversion

In Section 3.3 we discussed how a bandlimited signal can be reconstructed from a sequence of samples using ideal lowpass filtering. In terms of Fourier transforms, the reconstruction is represented as

$$X_r(j\Omega) = X(e^{j\Omega T})H_r(j\Omega), \tag{3.114}$$

where $X(e^{j\omega})$ is the discrete-time Fourier transform of the sequence of samples and $X_r(j\Omega)$ is the Fourier transform of the reconstructed continuous-time signal. The ideal reconstruction filter is

$$H_r(j\Omega) = \begin{cases} T, & |\Omega| < \pi/T, \\ 0, & |\Omega| > \pi/T. \end{cases} \tag{3.115}$$

In the time domain the corresponding relation between $x_r(t)$ and $x[n]$, where $H_r(j\Omega)$ is the ideal reconstruction filter, is

$$x_r(t) = \sum_{n=-\infty}^{\infty} x[n]\frac{\sin[\pi(t-nT)/T]}{\pi(t-nT)/T}. \tag{3.116}$$

The system that takes the sequence $x[n]$ as input and produces $x_r(t)$ as output is called the *ideal D/C converter*. A physically realizable counterpart to the ideal D/C converter is a *digital-to-analog converter* (D/A converter) followed by an approximate lowpass filter. As depicted in Fig. 3.36(a), a D/A converter takes a sequence of binary code words as its input and produces a continuous-time output of the form

$$\begin{aligned} x_{DA}(t) &= \sum_{n=-\infty}^{\infty} X_m \hat{x}_B[n]h_0(t-nT) \\ &= \sum_{n=-\infty}^{\infty} \hat{x}[n]h_0(t-nT), \end{aligned} \tag{3.117}$$

where $h_0(t)$ is the impulse response of the zero-order hold given by Eq. (3.99). The dotted lines in Fig. 3.32 show the output of a D/A converter for the quantized examples of the sine wave. Note that the D/A converter holds the quantized sample

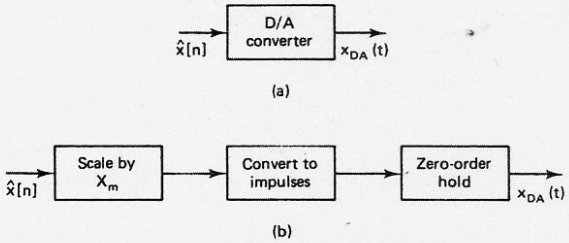


Figure 3.36 (a) Block diagram of D/A converter. (b) Representation in terms of a zero-order hold.

for one sample period in the same way that the sample and hold holds the unquantized input sample. If we use the additive-noise model to represent the effects of quantization, Eq. (3.117) becomes

$$x_{DA}(t) = \sum_{n=-\infty}^{\infty} x[n]h_0(t-nT) + \sum_{n=-\infty}^{\infty} e[n]h_0(t-nT). \tag{3.118}$$

To simplify our discussion, we define

$$x_0(t) = \sum_{n=-\infty}^{\infty} x[n]h_0(t-nT), \tag{3.119a}$$

$$e_0(t) = \sum_{n=-\infty}^{\infty} e[n]h_0(t-nT), \tag{3.119b}$$

so that Eq. (3.118) can be written as

$$x_{DA}(t) = x_0(t) + e_0(t). \tag{3.120}$$

The signal component $x_0(t)$ is related to the input signal $x_a(t)$ since $x[n] = x_a(nT)$. The noise signal $e_0(t)$ depends on the quantization-noise samples $e[n]$ in the same way that $x_0(t)$ depends on the unquantized signal samples. The Fourier transform of Eq. (3.119a) is

$$\begin{aligned} X_0(j\Omega) &= \sum_{n=-\infty}^{\infty} x[n]H_0(j\Omega)e^{-j\Omega nT} \\ &= \left(\sum_{n=-\infty}^{\infty} x[n]e^{-j\Omega Tn}\right)H_0(j\Omega) \\ &= X(e^{j\Omega T})H_0(j\Omega), \end{aligned} \tag{3.121}$$

Now since

$$X(e^{j\Omega T}) = \frac{1}{T}\sum_{k=-\infty}^{\infty} X_a(j\Omega + j2\pi k/T), \tag{3.122}$$

it follows that

$$X_0(j\Omega) = \left[\frac{1}{T}\sum_{k=-\infty}^{\infty} X_a(j\Omega + j2\pi k/T)\right]H_0(j\Omega). \tag{3.123}$$

If $X_a(j\Omega)$ is bandlimited to frequencies below π/T, the shifted copies of $X_a(j\Omega)$ do not overlap in Eq. (3.123), and if we define a compensated reconstruction filter as

$$\tilde{H}_r(j\Omega) = \frac{H_r(j\Omega)}{H_0(j\Omega)}, \tag{3.124}$$

then the output of the filter will be $x_a(t)$ if the input is $x_0(t)$. The frequency response of the zero-order-hold filter is easily shown to be

$$H_0(j\Omega) = \frac{2\sin(\Omega T/2)}{\Omega}e^{-j\Omega T/2}. \tag{3.125}$$

Therefore the compensated reconstruction filter is

$$\tilde{H}_r(j\Omega) = \begin{cases} \dfrac{\Omega T/2}{\sin(\Omega T/2)}e^{j\Omega T/2}, & |\Omega| < \pi/T, \\ 0, & |\Omega| > \pi/T. \end{cases} \tag{3.126}$$

Figure 3.37(a) shows $|H_0(j\Omega)|$ as given by Eq. (3.125) compared with the ideal interpolation filter $H_r(j\Omega)$ as given by Eq. (3.115). Both filters have a gain of T at $\Omega = 0$, but the zero-order hold, although lowpass in nature, does not cut off sharply at $\Omega = \pi/T$. Figure 3.37(b) shows the magnitude of the frequency response of the ideal compensated reconstruction filter to be used following a zero-order-hold reconstruction system such as a D/A converter. The phase response would ideally correspond to an advance time shift of $T/2$ s to compensate for the delay of that amount introduced

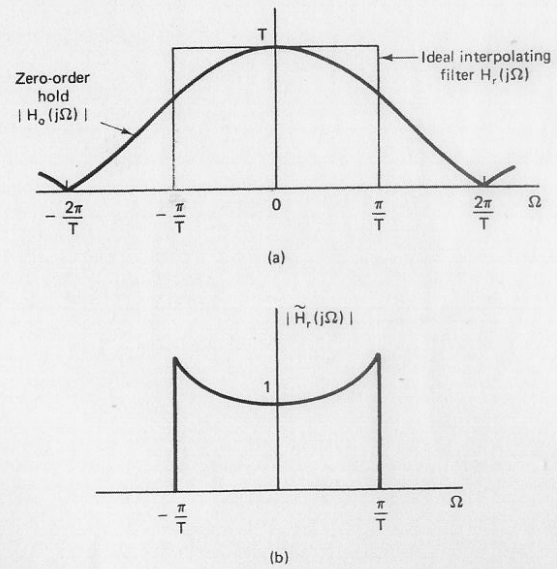


Figure 3.37 (a) Frequency response of zero-order hold compared with ideal interpolating filter. (b) Ideal compensated reconstruction filter for use with a zero-order hold output.

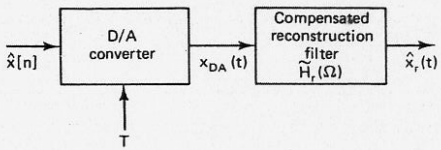


Figure 3.38 Physical configuration for digital-to-analog conversion.

by the zero-order hold. Since such time advance cannot be realized in practical real-time approximations to the ideal compensated reconstruction filter, only the magnitude response would normally be compensated, and often even this compensation is neglected since the gain of the zero-order hold drops to only $2/\pi$ (or -4 dB) at $\Omega = \pi/T$.

Figure 3.38 shows a D/A converter followed by an ideal compensated reconstruction filter. As can be seen from the preceding discussion, with the ideal compensated reconstruction filter following the D/A converter, the reconstructed output signal would be

$$\begin{aligned}\hat{x}_r(t) &= \sum_{n=-\infty}^{\infty} \hat{x}[n] \frac{\sin[\pi(t-nT)/T]}{\pi(t-nT)/T} \\ &= \sum_{n=-\infty}^{\infty} x[n] \frac{\sin[\pi(t-nT)/T]}{\pi(t-nT)/T} + \sum_{n=-\infty}^{\infty} e[n] \frac{\sin[\pi(t-nT)/T]}{\pi(t-nT)/T}.\end{aligned} \tag{3.127}$$

In other words, the output would be

$$\hat{x}_r(t) = x_a(t) + e_a(t), \tag{3.128}$$

where $e_a(t)$ would be a bandlimited white-noise signal.

Returning to a consideration of Fig. 3.26(b), we are now in a position to understand the behavior of systems for digital processing of analog signals. If we assume that the output of the anti-aliasing filter is bandlimited to frequencies below π/T, that $\tilde{H}_r(j\Omega)$ is similarly bandlimited, and that the discrete-time system is linear and time-invariant, then the output of the overall system will be of the form

$$\hat{y}_r(t) = y_a(t) + e_a(t), \tag{3.129}$$

where

$$Y_a(j\Omega) = \tilde{H}_r(j\Omega)H_0(j\Omega)H(e^{j\Omega T})H_{aa}(j\Omega)X_c(j\Omega), \tag{3.130}$$

where $H_{aa}(j\Omega)$, $H_0(j\Omega)$ and $\tilde{H}_r(j\Omega)$ are the frequency responses of the anti-aliasing filter, the zero-order hold, and the reconstruction lowpass filter, respectively. $H(e^{j\Omega T})$ is the frequency response of the discrete-time system. Similarly, assuming that the quantization noise introduced by the A/D converter is white noise with variance $\sigma_e^2 = \Delta^2/12$, it can be shown that the power spectrum of the output noise is

$$P_{e_a}(j\Omega) = |\tilde{H}_r(j\Omega)H_0(j\Omega)H(e^{j\Omega T})|^2\sigma_e^2, \tag{3.131}$$

i.e., the input quantization noise is changed by the successive stages of discrete- and continuous-time filtering. From Eq. (3.130) it follows that under the assumption of negligible aliasing, the overall effective frequency response from $x_c(t)$ to $\hat{y}_r(t)$ is

$$H_{eff}(j\Omega) = \tilde{H}_r(j\Omega)H_0(j\Omega)H(e^{j\Omega T})H_{aa}(j\Omega). \tag{3.132}$$

If the anti-aliasing filter is ideal, as in Eq. (3.95), and if the compensation of the reconstruction filter is ideal, as in Eq. (3.126), then the effective frequency response is as given in Eq. (3.96). Otherwise Eq. (3.132) provides a reasonable model for the effective response. Note that Eq. (3.132) suggests that compensation for imperfections in any of the four terms in Eq. (3.130) can in principle be included in any of the other terms; e.g., the discrete-time system can include appropriate compensation for the anti-aliasing filter or the zero-order hold or the reconstruction filter or all of these.

In addition to the filtering supplied by Eq. (3.132), Eq. (3.129) reminds us that the output will also be contaminated by the filtered quantization noise. In Chapter 6 we will see that noise can also be introduced in the implementation of the discrete-time linear system.

3.7.5 Application of Decimation and Interpolation to A/D and D/A Conversion

Throughout this section we have discussed idealized filters for preventing aliasing and for reconstructing a smoothly interpolated analog signal from the output of a D/A converter. Our underlying assumption has been that these filters could be adequately approximated, and indeed it is true that such sharp cutoff analog filters can be realized using active networks and integrated circuits. However, in applications involving powerful but inexpensive digital processors, the continuous-time filters may account for a major part of the cost of a system for discrete-time processing of analog signals. Sharp cutoff filters are difficult and expensive to implement, and if the system is to operate with a variable sampling rate, adjustable filters would be required. Thus, it is desirable to eliminate the continuous-time filters or simplify the requirements on them.

One approach is depicted in Fig. 3.39. First, the input continuous-time signal is sampled at a very high sampling rate—far above the Nyquist rate of the signal. Often such high sampling rates permit very inexpensive A/D conversion for low-bandwidth signals, and a very simple lowpass filter is sufficient to avoid aliasing. Then, as indicated in Fig. 3.39(a), the sampling rate can be reduced to a sampling rate as low as the Nyquist rate by the decimation techniques discussed in Section 3.6. As specified in Fig. 3.20, the sampling rate reduction must include a sharp cutoff digital filter. Any desired digital processing can then be done at the low sampling rate to minimize computation, as depicted in Fig. 3.39(b). If D/A conversion is required, the output of the discrete-time processor can be interpolated to a much higher sampling rate and then converted to a continuous-time output by using a D/A converter followed by a very simple reconstruction filter, since the discrete-time interpolation filter ensures that the output will be bandlimited to frequencies much lower than the output sampling frequency. This is depicted in Fig. 3.39(c). Note also that the compensation

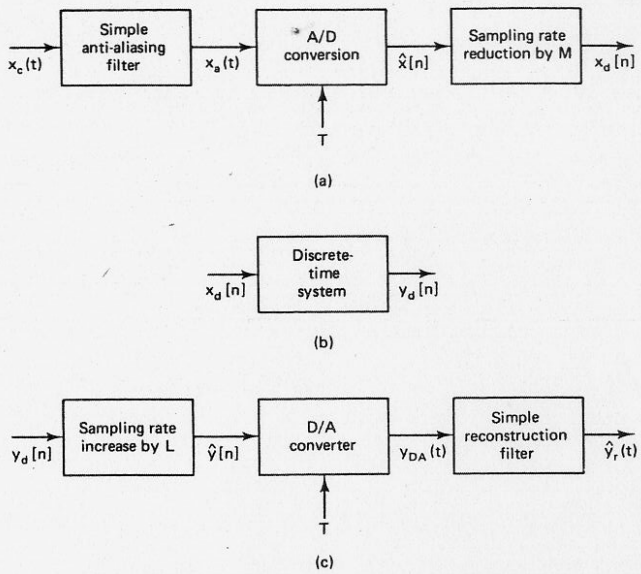


Figure 3.39 The requirements on the analog lowpass filters in sampling and reconstruction can be relaxed by use of a sampling rate much higher than the Nyquist rate. The sampling rate reduction in part (a) is done as indicated in Fig. 3.20. The sampling rate increase in part (c) is done as indicated in Fig. 3.21.

for the zero-order hold and the rolloff of the simple reconstruction filter can be in the filter required for the sampling rate increase.

The use of decimation in A/D conversion is illustrated in Fig. 3.40. Figure 3.40(a) shows the Fourier transform of a signal that occupies the band $|\Omega| < \Omega_N$ plus the Fourier transform of what might correspond to high-frequency "noise." Also shown (dotted line) is the frequency response of a simple anti-aliasing filter, which does not cut off sharply but eventually falls to zero at frequencies above a frequency Ω_c. Figure 3.40(b) shows the Fourier transform of the output of this filter. If the signal $x_a(t)$ is sampled with period T such that $(2\pi/T - \Omega_c) > \Omega_N$, then the discrete-time Fourier transform of the sequence $\hat{x}[n]$ will be as shown in Fig. 3.40(c). Note that the "noise" will be aliased, but aliasing will not affect the signal band $|\omega| < \omega_N = \Omega_N T$. Now if T and T' are chosen so that $T' = MT$ and $\pi/T' = \Omega_N$, then $\hat{x}[n]$ can be filtered by a sharp cutoff discrete-time filter (shown idealized in Fig. 3.40c) with unity gain and cutoff frequency π/M. The output of the discrete-time filter can be downsampled by M to obtain the sampled sequence $x_d[n]$ whose Fourier transform is shown in Fig. 3.40(d). Thus all the sharp cutoff filtering has been done by a discrete-time system, and only nominal continuous-time filtering is required.

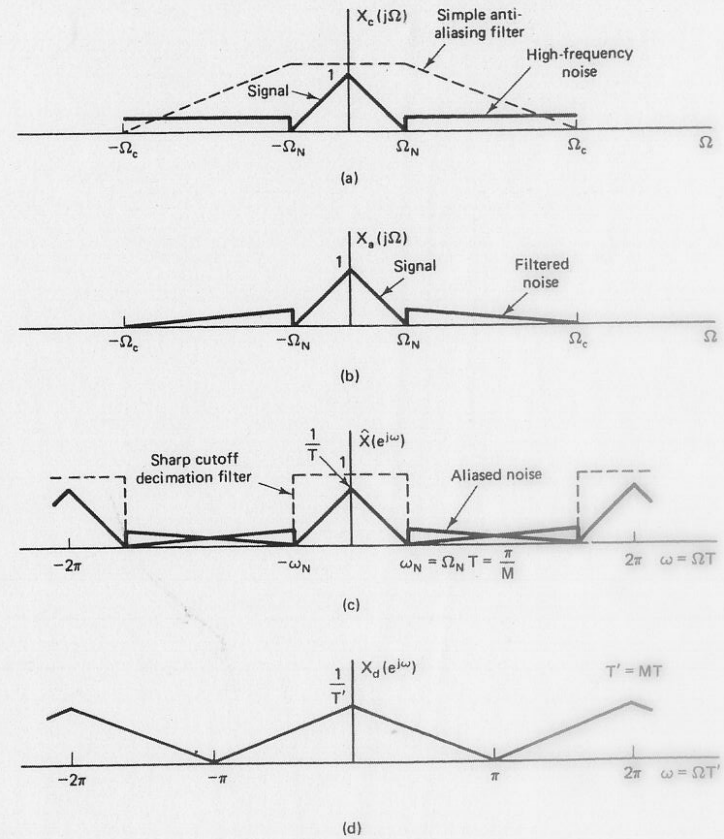


Figure 3.40 Use of decimation in A/D conversion.

To simplify the continuous-time filtering in D/A conversion, we can use the system of Fig. 3.39(c). Suppose that the input $y_d[n]$ has discrete-time Fourier transform as in Fig. 3.41(a), where Nyquist rate sampling (with period $T' = MT$) is assumed. Figure 3.41(b) shows the Fourier transform of the interpolated sequence (sampling period $T'' = T'/L$) together with the frequency response of the discrete-time interpolation filter, perhaps including compensation for a zero-order hold or reconstruction filtering. Figure 3.41(c) shows a simple reconstruction filter that will suffice to select the desired central image, producing an output with Fourier transform shown in Fig. 3.41(d). Note that we require only $\Omega_c < (2\pi/T'' - \Omega_N)$. Thus, we have used a sharp cutoff discrete-time system to provide the bandlimiting necessary for accurate reconstruction of the continuous-time output signal.

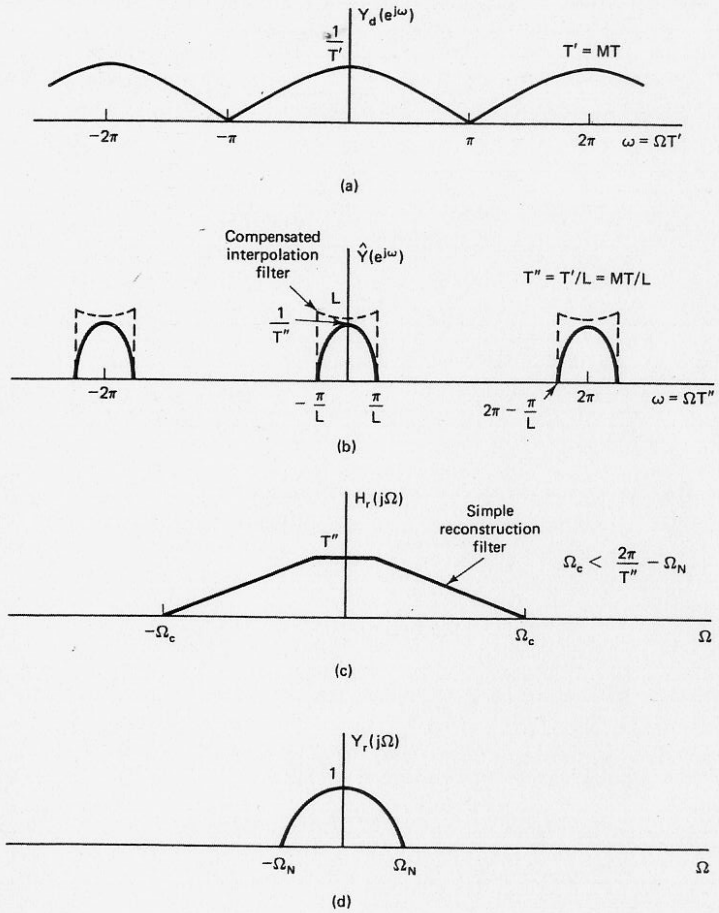


Figure 3.41 Use of interpolation in D/A conversion.

3.8 SUMMARY

In this chapter, we developed and explored the relationship between continuous-time signals and the discrete-time sequence obtained by periodic sampling. The fundamental theorem that allows the continuous-time signal to be represented by a sequence of samples is the Nyquist theorem, which states that for a bandlimited signal, periodic samples are a sufficient representation as long as the sampling rate is sufficiently high relative to the highest frequency in the continuous-time signal. Under this condition,

the continuous-time signal can be reconstructed from the samples by lowpass filtering, corresponding to bandlimited interpolation. If the sampling rate is too low relative to the bandwidth of the signal, then aliasing results.

The ability to represent signals by sampling permits the discrete-time processing of continuous-time signals. This is accomplished by first sampling, then applying discrete-time processing, and reconstructing a continuous-time signal from the result. Examples given were lowpass filtering and differentiation.

A particularly important class of processing is the class corresponding to sampling rate changes. Downsampling a discrete-time signal corresponds in the frequency domain to a replication of the discrete-time spectrum and rescaling of the frequency axis, which may require additional bandlimiting to avoid aliasing. Upsampling corresponds to effectively increasing the sampling rate and is also represented in the frequency domain by a rescaling of the frequency axis. By combining upsampling and downsampling by integer amounts, noninteger sampling rate conversion can be achieved.

In the final sections of the chapter, we explored a number of practical considerations associated with the discrete-time processing of continuous-time signals, including the use of prefiltering to avoid aliasing, quantization error in analog-to-digital conversion, and some issues associated with the filtering used in sampling and reconstruction of the continuous-time signals.

PROBLEMS

3.1. The sequence

$$x[n] = \cos\left(\frac{\pi}{4}n\right), \qquad -\infty < n < \infty,$$

was obtained by sampling an analog signal

$$x_c(t) = \cos(\Omega_0 t), \qquad -\infty < t < \infty,$$

at a sampling rate of 1000 samples/s. What are two possible values of Ω_0 that could have resulted in the sequence $x[n]$?

3.2. Let $h_c(t)$ denote the impulse response of a linear time-invariant continuous-time filter and $h_d[n]$ the impulse response of a linear time-invariant discrete-time filter.

(a) If

$$h_c(t) = \begin{cases} e^{-at}, & t \geq 0, \\ 0, & t < 0, \end{cases}$$

determine the continuous-time filter frequency response and sketch its magnitude.

(b) If $h_d[n] = Th_c(nT)$ with $h_c(t)$ as in part (a), determine the discrete-time filter frequency response and sketch its magnitude.

(c) For a given value of a, determine as a function of T, the minimum magnitude of the discrete-time filter frequency response.

3.3. A simple model of a multipath communication channel is indicated in Fig. P3.3-1. Assume that $s_c(t)$ is bandlimited such that $S_c(j\Omega) = 0$ for $|\Omega| \geq \pi/T$ and that $x_c(t)$ is

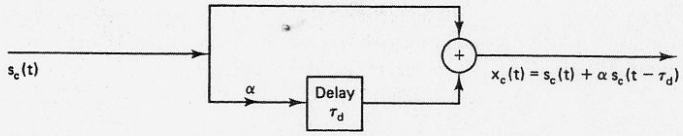


Figure P3.3-1

sampled with a sampling period T to obtain the sequence

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

(a) Determine the Fourier transform of $x_c(t)$ and the Fourier transform of $x[n]$ in terms of $S_c(j\Omega)$.

(b) We want to simulate the multipath system with a discrete-time system by choosing $H(e^{j\omega})$ in Fig. P3.3-2 so that the output $r[n] = x_c(nT)$ when the input is $s[n] = s_c(nT)$. Determine $H(e^{j\omega})$ in terms of T and τ_d.

(c) Determine the impulse response $h[n]$ in Fig. P3.3-2 when (i) $\tau_d = T$ and (ii) $\tau_d = T/2$.

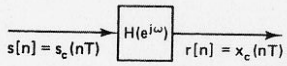


Figure P3.3-2

3.4. Consider the system in Fig. P3.4 with the following relations:

$$X_c(j\Omega) = 0, \qquad |\Omega| \geq 2\pi \times 10^4, \tag{P3.4-1}$$

$$x[n] = x_c(nT), \tag{P3.4-2}$$

$$y[n] = T \sum_{k=-\infty}^{n} x[k]. \tag{P3.4-3}$$

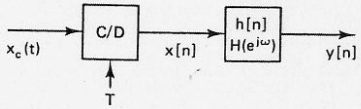


Figure P3.4

(a) For this system, what is the maximum allowable value of T if aliasing is to be avoided, i.e., so that $x_c(t)$ can be recovered from $x[n]$.

(b) Determine $h[n]$.

(c) In terms of $X(e^{j\omega})$, what is the value of $y[n]$ for $n = \infty$?

(d) Determine if there is any value of T for which

$$y[n]\Big|_{n=\infty} = \int_{-\infty}^{\infty} x_c(t)\, dt. \tag{P3.4-4}$$

If there is such a value for T, determine the *maximum* value. If there is not, explain and specify how T would be chosen so that the equality in Eq. (P3.4-4) is best approximated.

3.5. A complex bandpass analog signal $x_c(t)$ has the Fourier transform shown in Fig. P3.5, where $(\Omega_2 - \Omega_1) = \Delta\Omega$. This signal is sampled to produce the sequence $x[n] = x_c(nT)$.

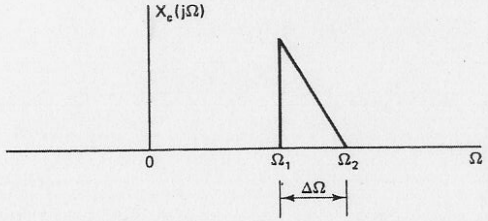


Figure P3.5

(a) Sketch the Fourier transform $X(e^{j\omega})$ of the sequence $x[n]$ for $T = \pi/\Omega_2$.

(b) What is the *lowest* sampling frequency that can be used without incurring any aliasing distortion, i.e., so that $x_c(t)$ can be recovered from $x[n]$?

(c) Draw the block diagram of a system that can be used to recover $x_c(t)$ from $x[n]$ if the sampling rate is greater than or equal to the rate determined in part (b). Assume that (complex) ideal filters are available.

3.6. An analog signal $x_c(t)$, with Fourier transform $X_c(j\Omega)$ shown in Fig. P3.6, is sampled with sampling period $T = 2\pi/\Omega_0$ to form the sequence $x[n] = x_c(nT)$.

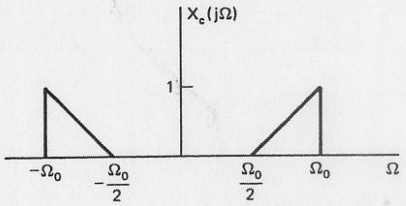


Figure P3.6

(a) Sketch the Fourier transform $X(e^{j\omega})$ for $|\omega| < \pi$.

(b) The signal $x[n]$ is to be transmitted across a digital channel. At the receiver, the original signal $x_c(t)$ must be recovered. Draw a block diagram of the recovery system and specify its characteristics. Assume that ideal filters are available.

(c) In terms of Ω_0, for what range of values for T can $x_c(t)$ be recovered from $x[n]$?

3.7. In many applications discrete-time random signals arise through periodic sampling of continuous-time random signals. We are concerned in this problem with a derivation of the sampling theorem for random signals. Consider a continuous-time, stationary, random process defined by the random variables $\{x_a(t)\}$, where t is a continuous variable. The autocorrelation function is defined as

$$\phi_{x_c x_c}(\tau) = \mathscr{E}\{x(t)x^*(t+\tau)\}$$

and the power density spectrum is

$$P_{x_c x_c}(\Omega) = \int_{-\infty}^{\infty} \phi_{x_c x_c}(\tau) e^{-j\Omega\tau}\, d\tau.$$

A discrete-time random process obtained by periodic sampling is defined by the set of random variables $\{x[n]\}$, where $x[n] = x_a(nT)$ and T is the sampling period.

(a) What is the relationship between $\phi_{xx}[n]$ and $\phi_{x_c x_c}(\tau)$?
(b) Express the power density spectrum of the discrete-time process in terms of the power density spectrum of the continuous-time process.
(c) Under what condition is the discrete-time power density spectrum a faithful representation of the continuous-time power density spectrum?

3.8. Consider a continuous-time random process $x_c(t)$ with a bandlimited power density spectrum $P_{x_c x_c}(\Omega)$ as depicted in Fig. P3.8-1. Suppose that we sample $x_c(t)$ to obtain the discrete-time random process $x[n] = x_c(nT)$.

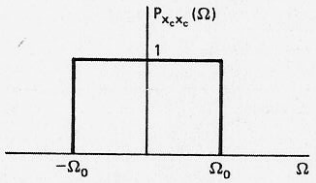


Figure P3.8-1

(a) What is the autocorrelation sequence of the discrete-time random process?
(b) For the continuous-time power density spectrum in Fig. P3.8-1, how should T be chosen so that the discrete-time process is white, i.e., so that the power spectrum is constant for all ω?
(c) If the continuous-time power density spectrum is as shown in Fig. P3.8-2, how should T be chosen so that the discrete-time process is white?

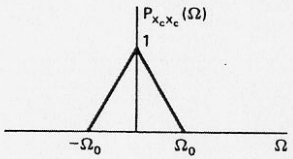


Figure P3.8-2

(d) What is the general requirement on the continuous-time process and the sampling period such that the discrete-time process is white?

3.9. In Fig. P3.9, assume that $X_c(j\Omega) = 0$, $|\Omega| \geq \pi/T_1$. For the general case in which $T_1 \neq T_2$ in the system, express $y_c(t)$ in terms of $x_c(t)$. Is the basic relationship different for $T_1 > T_2$ and $T_1 < T_2$?

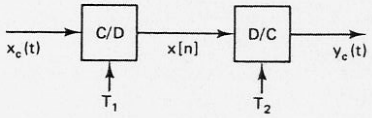


Figure P3.9

3.10. In the system of Fig. P3.10, $X_c(j\Omega)$ and $H(e^{j\omega})$ are as shown. Sketch and label the Fourier transform of $y_c(t)$ for each of the following cases:
(a) $1/T_1 = 1/T_2 = 10^4$
(b) $1/T_1 = 1/T_2 = 2 \times 10^4$
(c) $1/T_1 = 2 \times 10^4$, $1/T_2 = 10^4$
(d) $1/T_1 = 10^4$, $1/T_2 = 2 \times 10^4$

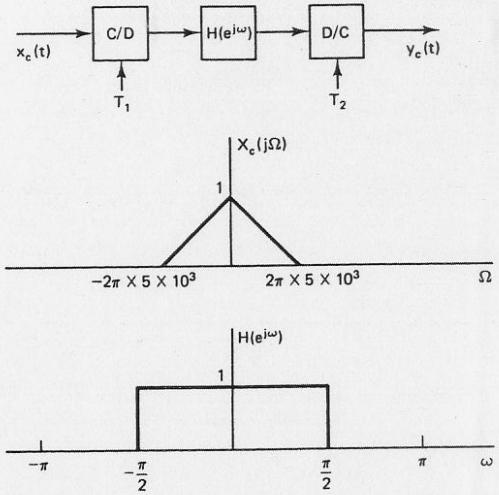


Figure P3.10

3.11. Figure P3.11-1 shows the overall system for filtering a continuous-time signal using a discrete-time filter. The frequency responses of the reconstruction filter $H_r(j\Omega)$ and the discrete-time filter $H(e^{j\omega})$ are shown in Fig. P3.11-2.

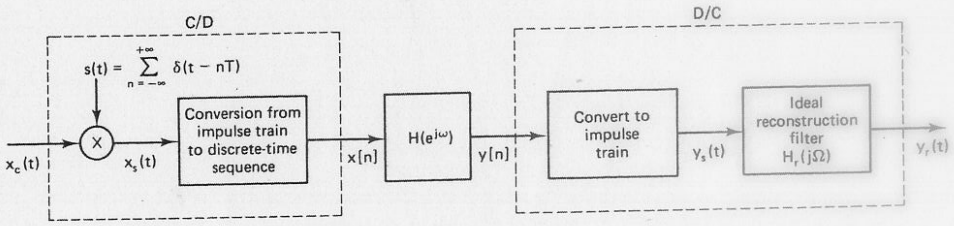


Figure P3.11-1

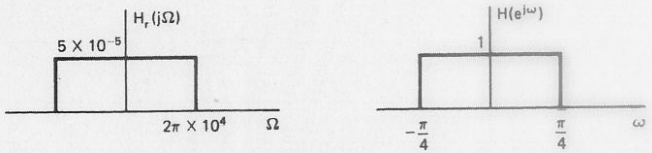


Figure P3.11-2

(a) For $X_c(j\Omega)$ as shown in Fig. P3.11-3 and $1/T = 20$ kHz, sketch $X_s(j\Omega)$ and $X(e^{j\omega})$.

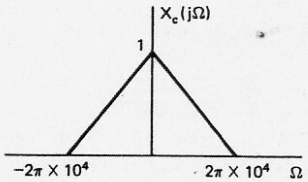


Figure P3.11-3

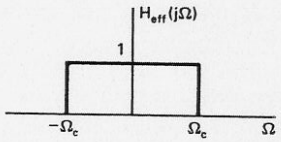


Figure P3.11-4

For a certain range of values for T the overall system, with input $x_c(t)$ and output $y_c(t)$, is equivalent to a continuous-time lowpass filter with frequency response $H_{eff}(j\Omega)$ sketched in Fig. P3.11-4.

(b) Determine the range of values of T for which this is true.

(c) For the range of values determined in (b), sketch Ω_c as a function of $1/T$.

Note: This is one way of implementing a variable-cutoff continuous-time filter using fixed continuous-time and discrete-time filters and a variable sampling rate.

3.12. In the system of Fig. 3.9, assume that the discrete-time system is linear and time-invariant and

$$X_c(j\Omega) = 0 \qquad \text{for} \qquad |\Omega| \geq 4000\pi.$$

Determine the largest possible value for T and the corresponding frequency response $H(e^{j\omega})$ for the discrete-time system so that

$$Y_c(j\Omega) = \begin{cases} |\Omega| X_c(j\Omega), & 1000\pi < |\Omega| < 2000\pi, \\ 0, & \text{otherwise.} \end{cases}$$

3.13. In the system of Fig. 3.9, assume that $X_c(j\Omega) = 0$ for $|\Omega| > \pi/T$. Determine and plot the magnitude and phase of the discrete-time LTI system such that the output $y_r(t)$ is the running integral of the input, i.e.,

$$y_r(t) = \int_{-\infty}^{t} x_c(\tau)\, d\tau.$$

3.14. Consider the system of Fig. 3.9, with the discrete-time system an ideal lowpass filter with cutoff frequency $\pi/8$ radians/s.

(a) If $x_c(t)$ is bandlimited to 5 kHz, what is the maximum value of T to avoid aliasing in the C/D converter?

(b) If $1/T = 10$ kHz, what will the cutoff frequency of the effective continuous-time filter be?

(c) Repeat part (b) for $1/T = 20$ kHz.

3.15. A bandlimited continuous-time signal is known to contain a 60-Hz component, which we want to remove by processing with the system of Fig. 3.9 where $T = 10^{-4}$.

(a) What is the highest frequency that the analog signal can contain if aliasing is to be avoided?

(b) The discrete-time system to be used has frequency response

$$H(e^{j\omega}) = \frac{[1 - e^{-j(\omega-\omega_0)}][1 - e^{-j(\omega+\omega_0)}]}{[1 - 0.9e^{-j(\omega-\omega_0)}][1 - 0.9e^{-j(\omega+\omega_0)}]}.$$

Sketch the magnitude and phase of $H(e^{j\omega})$.

(c) What value should be chosen for ω_0 to eliminate the 60-Hz component?

3.16. Consider the system in Fig. 3.9 with

$$X_c(j\Omega) = 0 \qquad \text{for} \qquad |\Omega| \geq 2\pi(1000)$$

and the discrete-time system a squarer, i.e., $y[n] = x^2[n]$. What is the largest value of T such that $y_c(t) = x_c^2(t)$?

3.17. This problem explores the effect of interchanging the order of two operations on a signal, namely, sampling and performing a zero-memory nonlinear operation.

(a) Consider the two signal processing systems in Fig. P3.17-1, where the C/D and D/C converters are ideal. The mapping $g[x] = x^2$ represents a zero-memory nonlinear device. For the two systems in Fig. P3.17-1, sketch the signal spectra at points 1, 2, and 3 when the sampling rate is selected to be $1/T = 2f_m$ samples/s and $x_c(t)$ has the spectral characteristic (bandlimited to f_m hertz) shown in Fig. P3.17-2. Is $y_1(t) = y_2(t)$? If not, why not? Is $y_1(t) = x^2(t)$? Explain your answer.

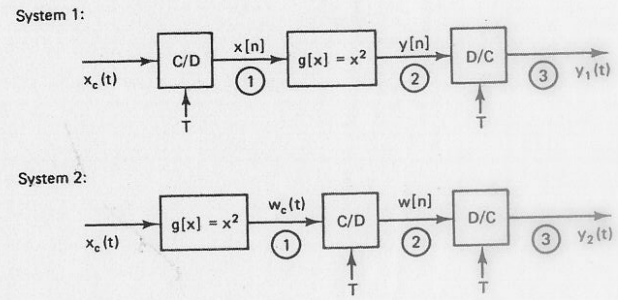


Figure P3.17-1

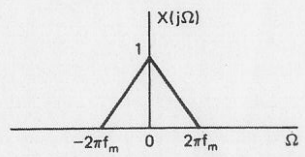


Figure P3.17-2

(b) Consider system 1, and let $x(t) = A \cos 30\pi t$. Let the sampling rate be $1/T = 40$ samples/s. Is $y_1(t) = x_c^2(t)$? Explain why or why not.

(c) Consider the signal processing system shown in Fig. P3.17-3, where $g[x] = x^3$ and

$g[x] = x^3$

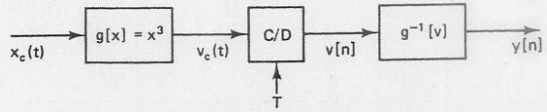


Figure P3.17-3

$g^{-1}[v]$ is the (unique) inverse, i.e., $g^{-1}[g(x)] = x$. Let $x(t) = A \cos 30\pi t$ and $1/T =$ 40 samples/s. Express $v[n]$ in terms of $x[n]$. Is there spectral aliasing? Express $y[n]$ in terms of $x[n]$. What conclusion can you reach from this example? (You may want to use the identity

$$\cos^3 \Omega_0 t = \tfrac{3}{4} \cos \Omega_0 t + \tfrac{1}{4} \cos 3\Omega_0 t.)$$

(d) One practical problem is that of digitizing a signal having a large dynamic range. Suppose we compress the dynamic range by passing the signal through a zero-memory nonlinear device prior to A/D conversion and then expand it back after A/D conversion. What is the impact of the nonlinear operation prior to the A/D converter in our choice of the sampling rate?

3.18. For the LTI system in Fig. P3.18-1, $H(e^{j\omega})$ is

$$H(e^{j\omega}) = e^{-j\omega/2}, \qquad |\omega| \le \pi \quad \text{(half-sample delay).}$$

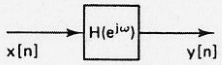


Figure P3.18-1

(a) Determine a choice for T and $h_c(t)$ in the system of Fig. 3.14 so that the system in Fig. 3.14 is equivalent to the system in Fig. P3.18-1 with $H(e^{j\omega})$ as specified.

(b) Determine and sketch $y[n]$ when the input sequence is

$$x[n] = \cos\left(\frac{5\pi}{2} n - \frac{\pi}{4}\right)$$

as sketched in Fig. P3.18-2.

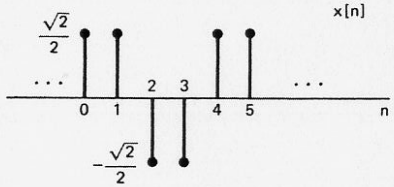


Figure P3.18-2

3.19. Consider the system of Fig. 3.14 with the continuous-time LTI system causal and characterized by the linear constant-coefficient differential equation

$$\frac{d^2 y_c(t)}{dt^2} + 4\frac{dy_c(t)}{dt} + 3y_c(t) = x_c(t).$$

The overall system is equivalent to a causal discrete-time linear time-invariant system. Determine the impulse response $h[n]$ of the equivalent discrete-time system.

3.20. Consider the sequence $x[n]$ whose Fourier transform $X(e^{j\omega})$ is shown in Fig. P3.20. Sketch $X_s(e^{j\omega})$ and $X_d(e^{j\omega})$ for $M = 3$ and for $\omega_H = \pi/2$ and $\omega_H = \pi/4$, where

$$x_s[n] = \begin{cases} x[n], & n = Mk, \quad k = 0, \pm 1, \pm 2, \ldots, \\ 0, & \text{otherwise,} \end{cases}$$

and

$$x_d[n] = x_s[Mn] = x[Mn].$$

What is the maximum value of ω_H to avoid aliasing when $M = 3$?

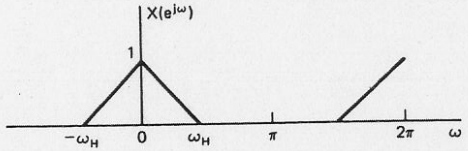


Figure P3.20

3.21. In Fig. P3.21, $x[n] = x_c(nT)$ and $y[n] = x[2n]$.

(a) In the system shown in Fig. P3.21, if $x_c(t)$ has a Fourier transform such that $X_c(j\Omega) = 0$, $|\Omega| > 2\pi \times 100$, what value of T is required so that

$$X(e^{j\omega}) = 0, \qquad \frac{\pi}{2} < |\omega| \le \pi?$$

(b) How should T' be chosen so that $y_c(t) = x_c(t)$?

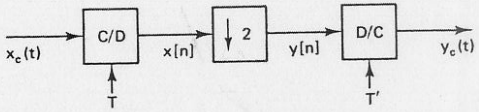


Figure P3.21

3.22. Figure 3.21 depicts a system for interpolating a signal by a factor of L, where

$$x_e[n] = \begin{cases} x[n/L], & n = 0, \ \pm L, \ \pm 2L, \ \text{etc.,} \\ 0, & \text{otherwise,} \end{cases}$$

and the lowpass filter interpolates between the nonzero values of $x_e[n]$ to generate the upsampled or interpolated signal $x_i[n]$. When the lowpass filter is ideal, the interpolation is referred to as bandlimited interpolation. As indicated in Section 3.6.2, simple interpolation procedures are often adequate. Two simple procedures often used are zero-order-hold and linear interpolation. For zero-order-hold interpolation, each value of $x[n]$ is simply repeated L times, i.e.,

$$x_i[n] = \begin{cases} x_e[0], & n = 0, \ 1, \ \ldots, \ L-1 \\ x_e[L], & n = L, \ L+1, \ \ldots, \ 2L-1 \\ x_e[2L], & n = 2L, \ 2L+1, \ \ldots, \\ \vdots \end{cases}$$

Linear interpolation is described in Section 3.6.2.

(a) Determine an appropriate choice for the impulse response of the lowpass filter in Fig. 3.21 to implement zero-order-hold interpolation. Also determine the corresponding frequency response.

(b) Equation (3.90) specifies the impulse response for linear interpolation. Determine the corresponding frequency response. (You may find it helpful to use the fact that $h_{\text{lin}}[n]$ is triangular and consequently corresponds to the convolution of two rectangular sequences.)

(c) Sketch the magnitude of the filter frequency response for zero-order-hold and linear interpolation. Which is a better approximation to ideal bandlimited interpolation?

3.23. Suppose you obtained a sequence $s[n]$ by filtering a speech signal $s_c(t)$ with a continuous-time lowpass filter with a cutoff frequency of 5 kHz and then sampling it at a 10-kHz rate as shown in Fig. P3.23-1. Unfortunately, the speech signal $s_c(t)$ was destroyed once the

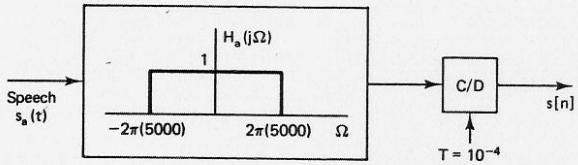


Figure P3.23-1

sequence $s[n]$ was stored on magnetic tape. Later, you find that what you should have done is followed the process shown in Fig. P3.23-2. Develop a method to obtain $s_1[n]$ from $s[n]$ using discrete-time processing. Your method may require a very large amount of computation but should *not* require a C/D or D/C converter. If your method uses a discrete-time filter, you should specify the frequency response of the filter.

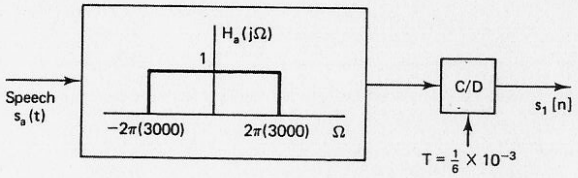


Figure P3.23-2

3.24. Consider the system shown in Fig. P3.24-1, where

$$H(e^{j\omega}) = \begin{cases} 1, & |\omega| < \pi/L, \\ 0, & \pi/L < |\omega| \le \pi. \end{cases}$$

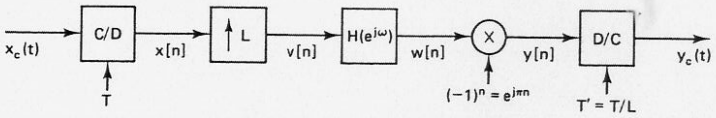


Figure P3.24-1

Sketch $Y_c(j\Omega)$ if $X_c(j\Omega)$ is as shown in Fig. P3.24-2.

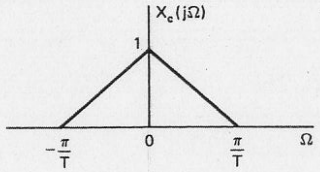


Figure P3.24-2

3.25. We wish to compute the autocorrelation function of an upsampled signal, as indicated in Fig. P3.25-1. It is suggested that this can equivalently be accomplished with the system of Fig. P3.25-2. Can $H_2(e^{j\omega})$ be chosen so that $\phi_3[n] = \phi_1[n]$? If not, why not? If so, specify $H_2(e^{j\omega})$.

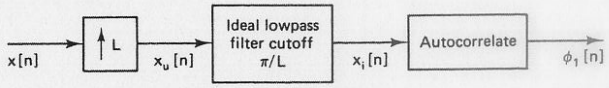


Figure P3.25-1

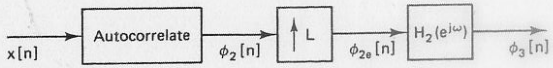


Figure P3.25-2

3.26. The system shown in Fig. P3.26 approximately interpolates the sequence $x[n]$ by a factor L. Suppose that the linear filter has impulse response $h[n]$ such that $h[n] = h[-n]$ and $h[n] = 0$ for $|n| > (RL - 1)$, where R and L are integers; i.e., the impulse response is symmetric and of length $(2RL - 1)$ samples.

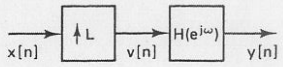


Figure P3.26

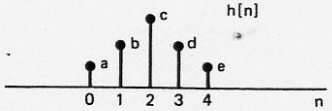
(a) In answering the following, do not be concerned about the causality of the system. It can be made causal by including some delay. Specifically, how much delay must be inserted to make the system causal?

(b) What conditions must be satisfied by $h[n]$ in order that $y[n] = x[n/L]$ for $n = 0, \pm L, \pm 2L, \pm 3L, \ldots$?

(c) By exploiting the symmetry of the impulse response, show that each sample of $y[n]$ can be computed with no more than RL multiplications.

(d) By taking advantage of the fact that multiplications by zero need not be done, show that only $2R$ multiplications per output sample are required.

3.27. We are interested in upsampling a sequence by a factor of 2, using a system of the form of Fig. 3.21. However, the lowpass filter in Fig. 3.21 is to be approximated by a 5-point filter with impulse response $h[n]$ indicated in Fig. P3.27-1. In this system the output $y_1[n]$ is obtained by direct convolution of $h[n]$ with $w[n]$.



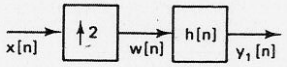


Figure P3.27-1

(a) A proposed implementation of the system with this choice of $h[n]$ is shown in Fig. P3.27-2. The three impulse responses $h_1[n]$, $h_2[n]$, and $h_3[n]$ are all restricted to be zero outside the range $0 \le n \le 2$. Determine and clearly justify a choice for $h_1[n]$, $h_2[n]$, and $h_3[n]$ so that $y_1[n] = y_2[n]$ for any $x[n]$, i.e., so that the two systems are identical.

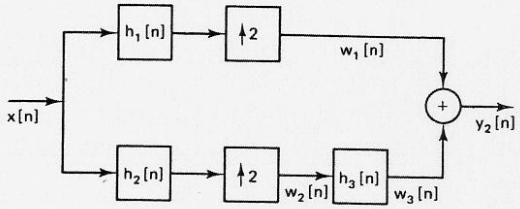


Figure P3.27-2

(b) Determine the number of multiplications per output point required in the system of Fig. P3.27-1 and in the system of Fig. P3.27-2. You should find that the system of Fig. P3.27-2 is more efficient.

3.28. In the system in Fig. P3.28,

$$X_c(j\Omega) = 0, \qquad |\Omega| \ge \pi/T,$$

and

$$H(e^{j\omega}) = \begin{cases} e^{-j\omega}, & |\omega| < \pi/L, \\ 0, & \pi/L < |\omega| \le \pi. \end{cases}$$

How is $y[n]$ related to the input signal $x_c(t)$?

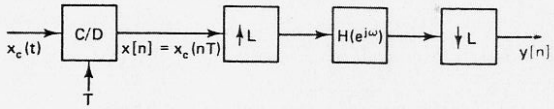


Figure P3.28

3.29. Consider the analysis/synthesis system shown in Fig. P3.29-1. The lowpass filter $h_0[n]$ is identical in the analyzer and synthesizer, and the highpass filter $h_1[n]$ is identical in the analyzer and synthesizer. The Fourier transforms of $h_0[n]$ and $h_1[n]$ are related by

$$H_1(e^{j\omega}) = H_0(e^{j(\omega+\pi)}).$$

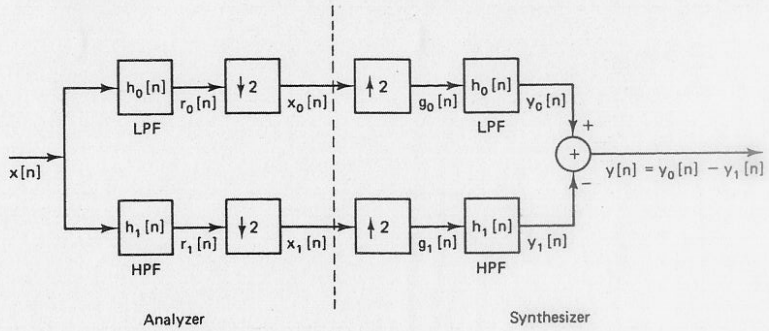


Figure P3.29-1

(a) If $X(e^{j\omega})$ and $H_0(e^{j\omega})$ are as shown in Fig. P3.29-2, sketch (to within a scale factor) $X_0(e^{j\omega})$, $G_0(e^{j\omega})$, and $Y_0(e^{j\omega})$.

(b) Write a general expression for $G_0(e^{j\omega})$ in terms of $X(e^{j\omega})$ and $H_0(e^{j\omega})$. Do *not* assume that $X(e^{j\omega})$ and $H_0(e^{j\omega})$ are as shown in Fig. P3.29-2.

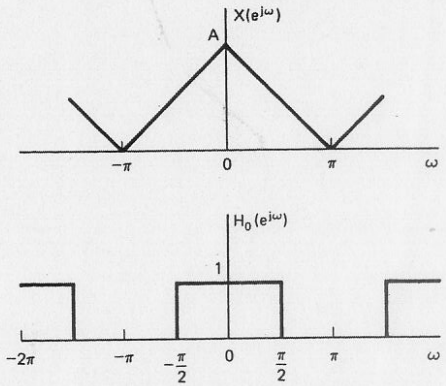


Figure P3.29-2

(c) Determine a set of conditions on $H_0(e^{j\omega})$ that is as general as possible and that will guarantee that $y[n]$ is proportional to $x[n - n_d]$ for any stable input $x[n]$.

Note: Analyzer/synthesizer filter banks of the form developed in this problem are called quadrature mirror filter banks. For further reading, see Crochiere and Rabiner (1983), pp. 378–392.

3.30. Consider a real-valued sequence $x[n]$ for which

$$X(e^{j\omega}) = 0, \qquad \frac{\pi}{3} \le |\omega| \le \pi.$$

One sequence value of $x[n]$ may have been corrupted and we would like to approximately or exactly recover it. With $\hat{x}[n]$ denoting the corrupted signal,

$$\hat{x}[n] = x[n] \qquad \text{for} \qquad n \ne n_0$$

and $\hat{x}[n_0]$ is real but not related to $x[n_0]$. In each of the following three cases, specify a practical algorithm for exactly or approximately recovering $x[n]$ from $\hat{x}[n]$.

(a) The value of n_0 is known.

(b) The exact value of n_0 is *not* known, but we know that n_0 is an even number.

(c) Nothing about n_0 is known.

3.31. Communication systems often require conversion from time-division multiplexing (TDM) to frequency-division multiplexing (FDM). In this problem we examine a simple example of such a system. The block diagram of the system to be studied is shown in Fig. P3.31-1. The TDM input is assumed to be the sequence of interleaved samples

$$w[n] = \begin{cases} x_1[n/2] & \text{for } n \text{ an even integer,} \\ x_2[(n-1)/2] & \text{for } n \text{ an odd integer.} \end{cases}$$

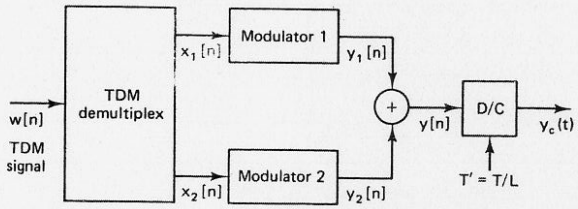


Figure P3.31-1

Assume that the sequences $x_1[n] = x_{c1}(nT)$ and $x_2[n] = x_{c2}(nT)$ have been obtained by sampling without aliasing the continuous-time signals $x_{c1}(t)$ and $x_{c2}(t)$, respectively. Also assume that these two signals have the same highest frequency, Ω_N, and that the sampling period is $T = \pi/\Omega_N$.

(a) Draw a block diagram of a system that produces $x_1[n]$ and $x_2[n]$ as outputs, i.e., obtain a system for TDM demultiplexing using simple operations. State whether or not your system is linear, time-invariant, causal, and stable.

The kth modulator system ($k = 1$ or 2) is defined by the block diagram in Fig. P3.31-2. The lowpass filter $H_l(e^{j\omega})$, which is the same for both channels, has gain L and cutoff frequency π/L, and the highpass filters $H_k(e^{j\omega})$ have unity gain and cutoff frequency ω_k. The modulator frequencies are such that

$$\omega_2 = \omega_1 + \pi/L \qquad \text{and} \qquad \omega_2 + \pi/L \le \pi \qquad (\text{assume } \omega_1 > \pi/2).$$

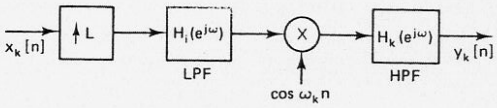


Figure P3.31-2

(b) Find ω_1 and L so that after ideal D/C conversion with sampling period T/L, the Fourier transform of $y_c(t)$ is zero except in the band of frequencies

$$2\pi \times 10^5 \le |\Omega| \le 2\pi \times 10^5 + 2\Omega_N.$$

Assume that $\Omega_N = 2\pi \times 5 \times 10^3$.

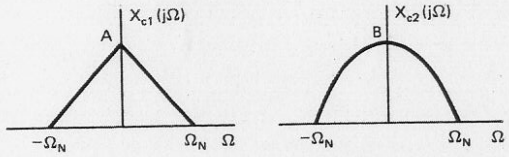


Figure P3.31-3

(c) Assume that the continuous-time Fourier transforms of the two original input signals are as sketched in Fig. P3.31-3. Sketch the Fourier transforms at each point in the system. This may help in determining the desired parameters in part (b).

(d) Based on your solution to parts (a)–(c), discuss how the system could be generalized to handle M equal-bandwidth channels.

3.32. In Section 3.7.1 we considered the use of prefiltering to avoid aliasing. In practice, the anti-aliasing filter cannot be ideal. However, the nonideal characteristics can be at least partially compensated for with a discrete-time system applied to the sequence $x[n]$ which is the output of the C/D converter.

Consider the two systems in Fig. P3.32-1. The anti-aliasing filters $H_{ideal}(j\Omega)$ and $H_{aa}(j\Omega)$ are shown in Fig. P3.32-2. $H(e^{j\omega})$ in Fig. P3.32-1 is to be specified to compensate for the nonideal characteristics of $H_{aa}(j\Omega)$.

Sketch $H(e^{j\omega})$ so that the two sequences $x[n]$ and $w[n]$ are identical.

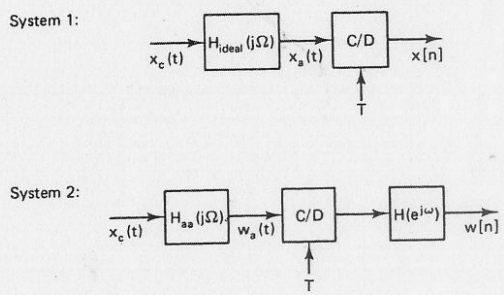


Figure P3.32-1

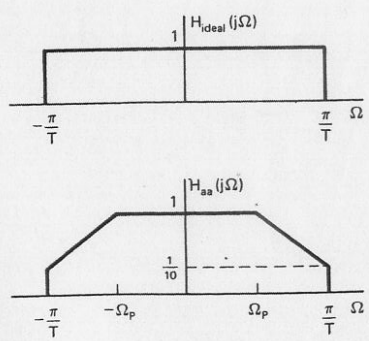


Figure P3.32-2

3.33. In this problem we consider some of the properties of the various number representations discussed in the text. Consider a number x such that $|x| < 1$ and further assume that $|x|$ can be represented as a binary fraction with b bits to the right of the binary point. We will find it convenient to introduce the notation $\simeq$ to mean "is represented by."

For sign and magnitude,

$$x \simeq \begin{cases} |x|, & x \geq 0, \\ 1 + |x|, & x \leq 0 \end{cases}$$

For one's complement,

$$x \simeq \begin{cases} |x|, & x \geq 0 \\ 2 - 2^{-b} - |x|, & x \leq 0 \end{cases}$$

For two's complement,

$$x \simeq \begin{cases} |x|, & x \geq 0 \\ 2 - |x|, & x < 0 \end{cases}$$

(a) Show that in all three representations, if x is represented by a $(B + 1)$-bit binary number, the bit immediately to the left of the binary point (the sign bit) is 0 if $x > 0$ and 1 if $x < 0$.

(b) Show that the following algorithm is sufficient to obtain the one's-complement representation of a negative number x from its magnitude $|x|$: Obtain the negative of $|x|$ by changing every 1 to a 0 and every 0 to a 1, including the sign bit.

(c) Show that the following algorithms serve to generate the two's-complement representation of a negative number x from its magnitude, $|x|$.

(i) Find the one's-complement representation and add 2^{-B}.

(ii) Starting at the right, examine the bits of $|x|$ in turn. For each 0 in $|x|$, place a 0 in x. When the first 1 is encountered in $|x|$, place a 1 in x. Thereafter, change each 0 to a 1 and each 1 to a zero, for all bits including the sign bit.

3.34. The two's-complement representation of binary numbers leads to very attractive simplifications of arithmetic processes. To illustrate this, consider some details of two's-complement addition. We use the notation $\simeq$ to mean "is represented by." Thus a number x has the two's-complement representation

$$x \simeq \begin{cases} |x|, & x \geq 0, \\ 2 - |x|, & x < 0, \end{cases}$$

where $|x| < 1$ and $(B + 1)$ bits are used in the representation of x. Two's-complement addition is performed as follows:

1. All numbers are treated as $(B + 1)$-bit *unsigned* binary numbers.
2. Addition is simple binary addition.
3. Carries past the sign bit are ignored; i.e., if the sum is greater than 2, the carry is ignored. Thus addition is implemented modulo 2.

(a) Using the above notation and definitions, write complete expressions for the two's-complement addition of two numbers x_1 and x_2, where $|x_1|$ and $|x_2| < 1$. Consider all possible cases; i.e., note that x_1 and x_2 can each be either $+$ or $-$ and also that $|x_1|$ can be either greater than or less than $|x_2|$.

(b) Note that when two numbers of like sign are added, the resultant magnitude may be greater than 1. This condition is called *overflow*. Show that overflow is indicated whenever the result of adding two numbers of like sign has the opposite sign.

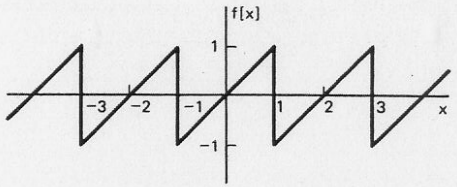


Figure P3.34

(c) Show that two's-complement addition of x_1 and x_2 is equivalent to $f[x_1 + x_2]$, where $f[x]$ is as depicted in Fig. P3.34.

(d) Suppose that $x_1 = 5/8$, $x_2 = 3/4$, and $x_3 = -(1/2)$. Find the two's-complement representation of each number, and add the two's-complement representations together in the sequence $(x_1 + x_2) + x_3$. Note that overflow occurs in the addition $(x_1 + x_2)$, but the final result is correct. Show that in general any number of overflows can occur in the process of accumulating the sum of three or more two's-complement numbers, and the result will be correct if the correct sum has magnitude less than unity.

3.35. In a floating-point representation of numbers, the number is represented in the form

$$F = 2^c M, \qquad \tfrac{1}{2} \leq |M| < 1,$$

where c is commonly called the characteristic and M the mantissa. Since $1/2 \leq |M| < 1$, M is a fixed-point fraction. Negative floating-point numbers can be represented by representing the mantissa as a floating-point fraction in either sign-and-magnitude, one's-complement, or two's-complement notation.

Consider a floating-point number F that is to be quantized by quantizing the mantissa to B bits, excluding sign, so that the value of the least significant bit in the mantissa is 2^{-B}.

Let $Q[F]$ represent the quantized value. It is convenient to represent $Q[F]$ as $Q[F] = F(1 + \varepsilon)$, so that the error $E = Q[F] - F$ is

$$E = \varepsilon F.$$

(a) Assuming that F is a positive number, show that for rounding, $-2^{-B} < \varepsilon \leq 2^{-B}$ and for truncation of the mantissa, $-2 \cdot 2^{-B} < \varepsilon \leq 0$.

(b) Assuming that F is a negative number, determine an upper and lower bound on ε for (i) rounding, (ii) sign-and-magnitude truncation, (iii) one's-complement truncation, and (iv) two's-complement truncation.

3.36. As discussed in Section 3.7.2, to process sequences on a digital computer, we must quantize the amplitude of the sequence to a set of discrete levels. This quantization can be expressed in terms of passing the input sequence $x[n]$ through a quantizer $Q(x)$ that has an input/output relation as shown in Fig. 3.31.

As discussed in Section 3.7.3, if the quantization interval Δ is small compared with changes in the level of the input sequence, we can assume that the output of the quantizer $y[n]$ is of the form

$$y[n] = x[n] + e[n],$$

where $e[n] = Q(x[n]) - x[n]$ and $e[n]$ is a stationary random process with a first-order probability density uniformly distributed between $-\Delta/2$ and $\Delta/2$, uncorrelated from sample to sample and uncorrelated with $x[n]$, so that $\mathscr{E}\{e[n]x[m]\} = 0$ for all m and n.

Let $x[n]$ be a stationary white-noise process, with zero mean and variance σ_x^2.

(a) Find the mean, variance, and autocorrelation sequence of $e[n]$.

(b) What is the signal-to-quantizing-noise ratio σ_x^2/σ_e^2?

(c) The quantized signal $y[n]$ is to be filtered by a digital filter with impulse response $h[n] = \frac{1}{2}[a^n + (-a)^n]u[n]$. Determine the variance of the noise produced at the output due to the input quantization noise and determine the signal-to-noise ratio at the output.

In some cases we may want to use nonlinear quantization steps, for example, logarithmically spaced quantization steps. This can be accomplished by applying uniform quantization to the logarithm of the input as depicted in Fig. P3.36, where $Q[\cdot]$ is a uniform quantizer as specified in Fig. 3.31. In this case, if we assume that Δ is small compared with changes in the sequence $\ln(x[n])$, then we can assume that the output of the quantizer is

$$\ln(y[n]) = \ln(x[n]) + e[n].$$

Thus

$$y[n] = x[n] \cdot \exp(e[n]).$$

For small e we can approximate $\exp(e[n])$ by $(1 + e[n])$, so that

$$y[n] \simeq x[n](1 + e[n]) = x[n] + f[n]. \tag{P3.36}$$

This equation will be used to describe the effect of logarithmic quantization. We assume $e[n]$ to be a stationary random process, uncorrelated from sample to sample, independent of the signal $x[n]$, and with first-order probability density uniformly distributed between $\pm\Delta/2$.

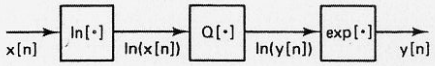


Figure P3.36

(d) Determine the mean, variance, and autocorrelation sequence of the *additive* noise $f[n]$ defined in Eq. (P3.36).

(e) What is the signal-to-quantizing-noise ratio σ_x^2/σ_f^2? Note that in this case σ_x^2/σ_f^2 is independent of σ_x^2. Within the limits of our assumption, therefore, the signal-to-quantizing-noise ratio is independent of the input signal level, whereas for linear quantization the ratio σ_x^2/σ_e^2 depends directly on σ_x^2.

(f) The quantized signal $y[n]$ is to be filtered by means of a digital filter with impulse response $h[n] = \frac{1}{2}[a^n + (-a)^n]u[n]$. Determine the variance of the noise produced at the output due to the input quantization noise and determine the signal-to-noise ratio at the output.

The z-Transform

4

4.0 INTRODUCTION

We have seen that the Fourier transform plays a key role in representing and analyzing discrete-time signals and systems. In this chapter, we consider a generalization of the Fourier transform referred to as the *z-transform*. The z-transform for discrete-time signals is the counterpart of the Laplace transform for continuous-time signals, and they each have a similar relationship to the corresponding Fourier transform. A principal motivation for introducing this generalization is that the Fourier transform does not converge for all sequences and it is useful to have a generalization of the Fourier transform that encompasses a broader class of signals. A second advantage is that in analytical problems the z-transform notation is often more convenient than the Fourier transform notation. Still another advantage is that the z-transform allows us to bring the power of complex variable theory to bear on problems of discrete-time signals and systems.

In this chapter, we define the z-transform representation of a sequence and study how the properties of a sequence are related to the properties of its z-transform. In discussing the z-transform we use a number of results from the theory of complex variables. In applying these results we will be precise, but we will not attempt to maintain a high degree of mathematical rigor.

4.1 z-TRANSFORM

The Fourier transform $X(e^{j\omega})$ of a sequence $x[n]$ was defined in Chapter 2 as

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n]e^{-j\omega n}. \tag{4.1}$$

The z-transform $X(z)$ of a sequence $x[n]$ is defined as

$$X(z) = \sum_{n=-\infty}^{\infty} x[n]z^{-n}. \tag{4.2}$$