Introduction

Digital signal processing is an area of science and engineering that has developed rapidly over the past 40 years. This rapid development is a result of the significant advances in digital computer technology and integrated-circuit fabrication. The digital computers and associated digital hardware of four decades ago were relatively large and expensive and, as a consequence, their use was limited to general-purpose non-real-time (off-line) scientific computations and business applications. The rapid developments in integrated-circuit technology, starting with medium-scale integration (MSI) and progressing to large-scale integration (LSI), and now, very-large-scale integration (VLSI) of electronic circuits has spurred the development of powerful, smaller, faster, and cheaper digital computers and special-purpose digital hardware. These inexpensive and relatively fast digital circuits have made it possible to construct highly sophisticated digital systems capable of performing complex digital signal processing functions and tasks, which are usually too difficult and/or too expensive to be performed by analog circuitry or analog signal processing systems. Hence many of the signal processing tasks that were conventionally performed by analog means are realized today by less expensive and often more reliable digital hardware.

We do not wish to imply that digital signal processing is the proper solution for all signal processing problems. Indeed, for many signals with extremely wide bandwidths, real-time processing is a requirement. For such signals, analog or, perhaps, optical signal processing is the only possible solution. However, where digital circuits are available and have sufficient speed to perform the signal processing, they are usually preferable.

Not only do digital circuits yield cheaper and more reliable systems for signal processing, they have other advantages as well. In particular, digital processing hardware allows programmable operations. Through software, one can more eas-

From Chapter 1 of *Digital Signal Processing: Principles, Algorithms, and Applications*, Fourth Edition.
John G. Proakis, Dimitris G. Manolakis. Copyright © 2007 by Pearson Education, Inc. All rights reserved.

ily modify the signal processing functions to be performed by the hardware. Thus digital hardware and associated software provide a greater degree of flexibility in system design. Also, there is often a higher order of precision achievable with digital hardware and software compared with analog circuits and analog signal processing systems. For all these reasons, there has been an explosive growth in digital signal processing theory and applications over the past three decades.

We begin by introducing some of with the process of converting an analog signal to digital form suitable for digital processing. As we shall see, digital processing of analog signals has some drawbacks. First, and foremost, conversion of an analog signal to digital form, accomplished by sampling the signal and quantizing the samples, results in a distortion that prevents us from reconstructing the original analog signal from the quantized samples. Control of the amount of this distortion is achieved by proper choice of the sampling rate and the precision in the quantization process. Second, there are finite precision effectsthat must be considered in the digital processing of the quantized samples.

1 Signals, Systems, and Signal Processing

A *signal* is defined as any physical quantity that varies with time, space, or any other independent variable or variables. Mathematically, we describe a signal as a function of one or more independent variables. For example, the functions

$$\begin{aligned} s_1(t) &= 5t \\ s_2(t) &= 20t^2 \end{aligned} \tag{1.1}$$

describe two signals, one that varies linearly with the independent variable t (time) and a second that varies quadratically with t. As another example, consider the function

$$s(x, y) = 3x + 2xy + 10y^2 \tag{1.2}$$

This function describes a signal of two independent variables x and y that could represent the two spatial coordinates in a plane.

The signals described by (1.1) and (1.2) belong to a class of signals that are precisely defined by specifying the functional dependence on the independent variable. However, there are cases where such a functional relationship is unknown or too highly complicated to be of any practical use.

For example, a speech signal (see Fig. 1.1) cannot be described functionally by expressions such as (1.1). In general, a segment of speech may be represented to

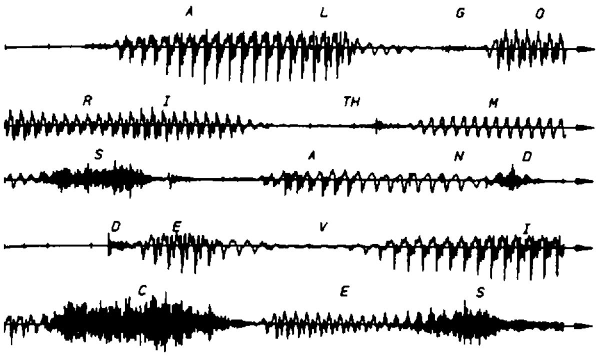


Figure 1.1
Example of a speech signal.

a high degree of accuracy as a sum of several sinusoids of different amplitudes and frequencies, that is, as

$$\sum_{i=1}^{N} A_i(t) \sin[2\pi F_i(t)t + \theta_i(t)] \tag{1.3}$$

where $\{A_i(t)\}$, $\{F_i(t)\}$, and $\{\theta_i(t)\}$ are the sets of (possibly time-varying) amplitudes, frequencies, and phases, respectively, of the sinusoids. In fact, one way to interpret the information content or message conveyed by any short time segment of the speech signal is to measure the amplitudes, frequencies, and phases contained in the short time segment of the signal.

Another example of a natural signal is an electrocardiogram (ECG). Such a signal provides a doctor with information about the condition of the patient's heart. Similarly, an electroencephalogram (EEG) signal provides information about the activity of the brain.

Speech, electrocardiogram, and electroencephalogram signals are examples of information-bearing signals that evolve as functions of a single independent variable, namely, time. An example of a signal that is a function of two independent variables is an image signal. The independent variables in this case are the spatial coordinates. These are but a few examples of the countless number of natural signals encountered in practice.

Associated with natural signals are the means by which such signals are generated. For example, speech signals are generated by forcing air through the vocal cords. Images are obtained by exposing a photographic film to a scene or an object. Thus signal generation is usually associated with a *system* that responds to a stimulus or force. In a speech signal, the system consists of the vocal cords and the vocal tract, also called the vocal cavity. The stimulus in combination with the system is called a *signal source*. Thus we have speech sources, images sources, and various other types of signal sources.

A *system* may also be defined as a physical device that performs an operation on a signal. For example, a filter used to reduce the noise and interference corrupting a desired information-bearing signal is called a system. In this case the filter performs some operation(s) on the signal, which has the effect of reducing (filtering) the noise and interference from the desired information-bearing signal.

When we pass a signal through a system, as in filtering, we say that we have processed the signal. In this case the processing of the signal involves filtering the noise and interference from the desired signal. In general, the system is characterized by the type of operation that it performs on the signal. For example, if the operation is linear, the system is called linear. If the operation on the signal is nonlinear, the system is said to be nonlinear, and so forth. Such operations are usually referred to as *signal processing*.

For our purposes, it is convenient to broaden the definition of a system to include not only physical devices, but also software realizations of operations on a signal. In digital processing of signals on a digital computer, the operations performed on a signal consist of a number of mathematical operations as specified by a software program. In this case, the program represents an implementation of the system in *software*. Thus we have a system that is realized on a digital computer by means of a sequence of mathematical operations; that is, we have a digital signal processing system realized in software. For example, a digital computer can be programmed to perform digital filtering. Alternatively, the digital processing on the signal may be performed by digital *hardware* (logic circuits) configured to perform the desired specified operations. In such a realization, we have a physical device that performs the specified operations. In a broader sense, a digital system can be implemented as a combination of digital hardware and software, each of which performs its own set of specified operations.

This text deals with the processing of signals by digital means, either in software or in hardware. Since many of the signals encountered in practice are analog, we must also consider the problem of converting an analog siganl into a digital signal for processing. Thus we deal primarily with digital systems. The operations performed by such a system can usually be specified mathematically. The method or set of rules for implementing the system by a program that performs the corresponding mathematical operations is called an *algorithm*. Usually, there are many ways or algorithms by which a system can be implemented, either in software or in hardware, to perform the desired operations and computations. In practice, we have an interest in devising algorithms that are computationally efficient, fast, and easily implemented. Thus a major topic in the study of digital signal processing is the discussion of efficient algorithms for performing such operations as filtering, correlation, and spectral analysis.

1.1 Basic Elements of a Digital Signal Processing System

Most of the signals encountered in science and engineering are analog in nature. That is, the signals are functions of a continuous variable, such as time or space, and usually take on values in a continuous range. Such signals may be processed directly by appropriate analog systems (such as filters, frequency analyzers, or frequency multipliers) for the purpose of changing their characteristics or extracting some desired information. In such a case we say that the signal has been processed directly in its analog form, as illustrated in Fig. 1.2. Both the input signal and the output signal are in analog form.

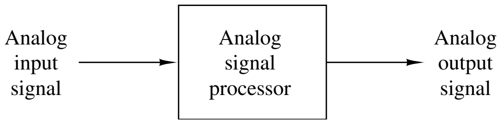


Figure 1.2
Analog signal processing.

Digital signal processing provides an alternative method for processing the analog signal, as illustrated in Fig. 1.3. To perform the processing digitally, there is a need for an interface between the analog signal and the digital processor. This interface is called an *analog-to-digital* (*A/D*) *converter*. The output of the A/D converter is a digital signal that is appropriate as an input to the digital processor.

The digital signal processor may be a large programmable digital computer or a small microprocessor programmed to perform the desired operations on the input signal. It may also be a hardwired digital processor configured to perform a specified set of operations on the input signal. Programmable machines provide the flexibility to change the signal processing operations through a change in the software, whereas hardwired machines are difficult to reconfigure. Consequently, programmable signal processors are in very common use. On the other hand, when signal processing operations are well defined, a hardwired implementation of the operations can be optimized, resulting in a cheaper signal processor and, usually, one that runs faster than its programmable counterpart. In applications where the digital output from the digital signal processor is to be given to the user in analog form, such as in speech communications, we must provide another interface from the digital domain to the analog domain. Such an interface is called a *digital-to-analog (D/A) converter*. Thus the signal is provided to the user in analog form, as illustrated in the block diagram of Fig. 1.3. However, there are other practical applications involving signal analysis, where the desired information is conveyed in digital form and no D/A converter is required. For example, in the digital processing of radar signals, the information extracted from the radar signal, such as the position of the aircraft and its speed, may simply be printed on paper. There is no need for a D/A converter in this case.

1.2 Advantages of Digital over Analog Signal Processing

There are many reasons why digital signal processing of an analog signal may be preferable to processing the signal directly in the analog domain, as mentioned briefly earlier. First, a digital programmable system allows flexibility in reconfiguring the digital signal processing operations simply by changing the program. Reconfigu-

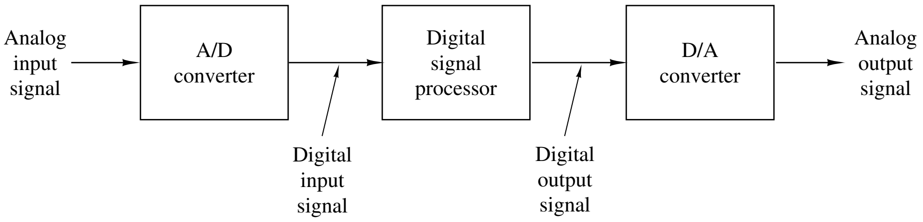


Figure 1.3 Block diagram of a digital signal processing system.

ration of an analog system usually implies a redesign of the hardware followed by testing and verification to see that it operates properly.

Accuracy considerations also play an important role in determining the form of the signal processor. Tolerances in analog circuit components make it extremely difficult for the system designer to control the accuracy of an analog signal processing system. On the other hand, a digital system provides much better control of accuracy requirements. Such requirements, in turn, result in specifying the accuracy requirements in the A/D converter and the digital signal processor, in terms of word length, floating-point versus fixed-point arithmetic, and similar factors.

Digital signals are easily stored on magnetic media (tape or disk) without deterioration or loss of signal fidelity beyond that introduced in the A/D conversion. As a consequence, the signals become transportable and can be processed off-line in a remote laboratory. The digital signal processing method also allows for the implementation of more sophisticated signal processing algorithms. It is usually very difficult to perform precise mathematical operations on signals in analog form but these same operations can be routinely implemented on a digital computer using software.

In some cases a digital implementation of the signal processing system is cheaper than its analog counterpart. The lower cost may be due to the fact that the digital hardware is cheaper, or perhaps it is a result of the flexibility for modifications provided by the digital implementation.

As a consequence of these advantages, digital signal processing has been applied in practical systems covering a broad range of disciplines. We cite, for example, the application of digital signal processing techniques in speech processing and signal transmission on telephone channels, in image processing and transmission, in seismology and geophysics, in oil exploration, in the detection of nuclear explosions, in the processing of signals received from outer space, and in a vast variety of other applications.

As already indicated, however, digital implementation has its limitations. One practical limitation is the speed of operation of A/D converters and digital signal processors. We shall see that signals having extremely wide bandwidths require fast-sampling-rate A/D converters and fast digital signal processors. Hence there are analog signals with large bandwidths for which a digital processing approach is beyond the state of the art of digital hardware.

2 Classification of Signals

The methods we use in processing a signal or in analyzing the response of a system to a signal depend heavily on the characteristic attributes of the specific signal. There are techniques that apply only to specific families of signals. Consequently, any investigation in signal processing should start with a classification of the signals involved in the specific application.

2.1 Multichannel and Multidimensional Signals

As explained in Section 1, a signal is described by a function of one or more independent variables. The value of the function (i.e., the dependent variable) can be a real-valued scalar quantity, a complex-valued quantity, or perhaps a vector. For example, the signal

$$s_1(t) = A \sin 3\pi t$$

is a real-valued signal. However, the signal

$$s_2(t) = Ae^{j3\pi t} = A \cos 3\pi t + jA \sin 3\pi t$$

is complex valued.

In some applications, signals are generated by multiple sources or multiple sensors. Such signals, in turn, can be represented in vector form. Figure 2.1 shows the three components of a vector signal that represents the ground acceleration due to an earthquake. This acceleration is the result of three basic types of elastic waves. The primary (P) waves and the secondary (S) waves propagate within the body of

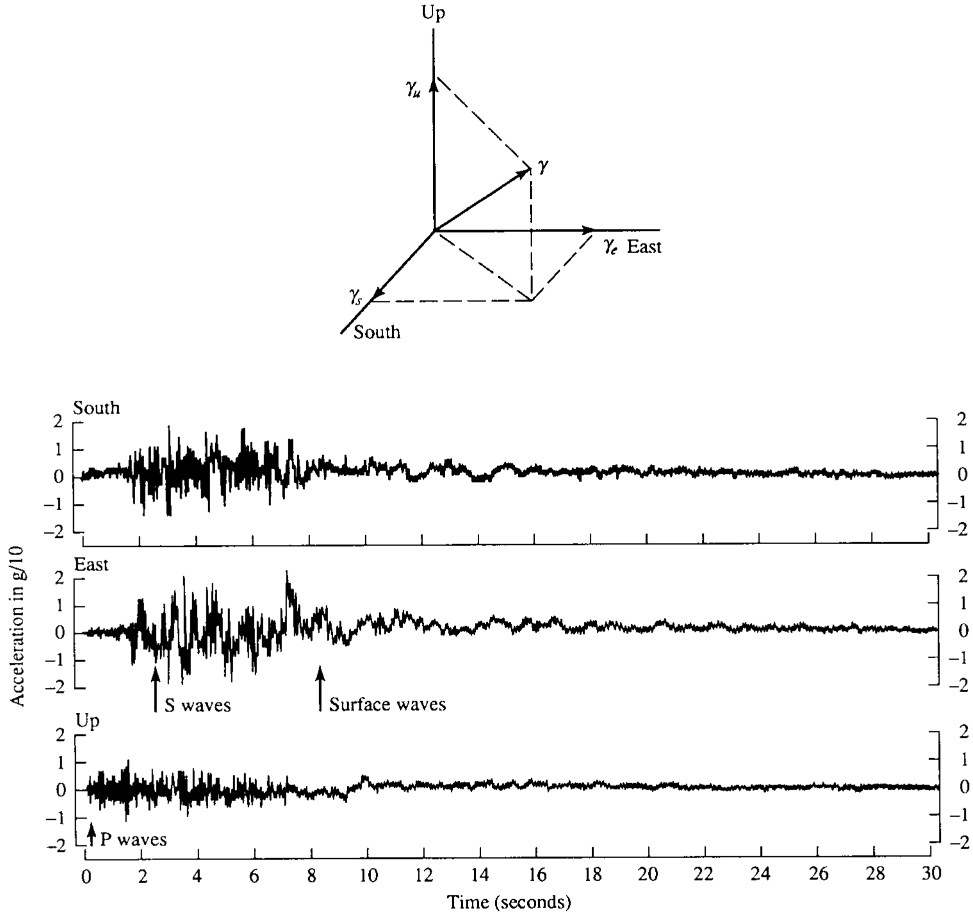


Figure 2.1 Three components of ground acceleration measured a few kilometers from the epicenter of an earthquake. (From *Earthquakes*, by B. A. Bold, ©1988 by W. H. Freeman and Company. Reprinted with permission of the publisher.)

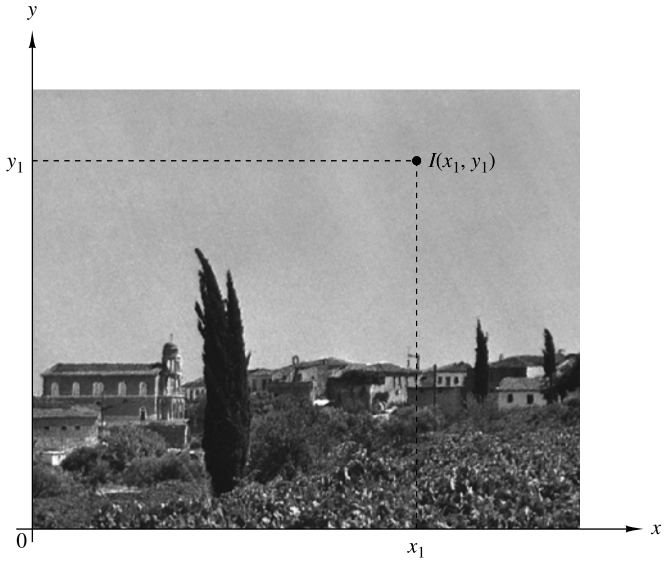


Figure 2.2
Example of a two-dimensional signal.

rock and are longitudinal and transversal, respectively. The third type of elastic wave is called the surface wave, because it propagates near the ground surface. If $s_k(t)$, $k = 1, 2, 3$, denotes the electrical signal from the kth sensor as a function of time, the set of $p = 3$ signals can be represented by a vector $\mathbf{S}_3(t)$, where

$$\mathbf{S}_3(t) = \begin{bmatrix} s_1(t) \\ s_2(t) \\ s_3(t) \end{bmatrix}$$

We refer to such a vector of signals as a *multichannel signal.*In electrocardiography, for example, 3-lead and 12-lead electrocardiograms (ECG) are often used in practice, which result in 3-channel and 12-channel signals.

Let us now turn our attention to the independent variable(s). If the signal is a function of a single independent variable, the signal is called a *one-dimensional* signal. On the other hand, a signal is called *M-dimensional* if its value is a function of M independent variables.

The picture shown in Fig. 2.2 is an example of a two-dimensional signal, since the intensity or brightness $I(x, y)$ at each point is a function of two independent variables. On the other hand, a black-and-white television picture may be represented as $I(x, y, t)$ since the brightness is a function of time. Hence the TV picture may be treated as a three-dimensional signal. In contrast, a color TV picture may be described by three intensity functions of the form $I_r(x, y, t)$, $I_g(x, y, t)$, and $I_b(x, y, t)$, corresponding to the brightness of the three principal colors (red, green, blue) as functions of time. Hence the color TV picture is a three-channel, three-dimensional signal, which can be represented by the vector

$$\mathbf{I}(x, y, t) = \begin{bmatrix} I_r(x, y, t) \\ I_g(x, y, t) \\ I_b(x, y, t) \end{bmatrix}$$

In this text we deal mainly with single-channel, one-dimensional real- or complex-valued signals and we refer to them simply as signals. In mathematical terms these signals are described by a function of a single independent variable. Although the independent variable need not be time, it is common practice to use t as the independent variable. In many cases the signal processing operations and algorithms developed in this text for one-dimensional, single-channel signals can be extended to multichannel and multidimensional signals.

2.2 Continuous-Time Versus Discrete-Time Signals

Signals can be further classified into four different categories depending on the characteristics of the time (independent) variable and the values they take. *Continuous-time signals* or *analog signals* are defined for every value of time and they take on values in the continuous interval (a, b), where a can be $-\infty$ and b can be ∞. Mathematically, these signals can be described by functions of a continuous variable. The speech waveform in Fig. 1.1 and the signals $x_1(t) = \cos \pi t$, $x_2(t) = e^{-|t|}$, $-\infty < t < \infty$ are examples of analog signals. *Discrete-time signals*are defined only at certain specific values of time. These time instants need not be equidistant, but in practice they are usually taken at equally spaced intervals for computational convenience and mathematical tractability. The signal $x(t_n) = e^{-|t_n|}$, $n = 0, \pm 1, \pm 2, \ldots$ provides an example of a discrete-time signal. If we use the index n of the discrete-time instants as the independent variable, the signal value becomes a function of an integer variable (i.e., a sequence of numbers). Thus a discrete-time signal can be represented mathematically by a sequence of real or complex numbers. To emphasize the discrete-time nature of a signal, we shall denote such a signal as $x(n)$ instead of $x(t)$. If the time instants t_n are equally spaced (i.e., $t_n = nT$), the notation $x(nT)$ is also used. For example, the sequence

$$x(n) = \begin{cases} 0.8^n, & \text{if } n \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{2.1}$$

is a discrete-time signal, which is represented graphically as in Fig. 2.3.

In applications, discrete-time signals may arise in two ways:

1. By selecting values of an analog signal at discrete-time instants. This process is called *sampling* and is discussed in more detail in Section 4. All measuring instruments that take measurements at a regular interval of time provide

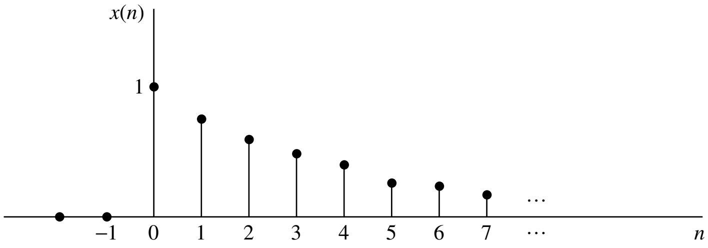


Figure 2.3 Graphical representation of the discrete time signal $x(n) = 0.8^n$ for $n > 0$ and $x(n) = 0$ for $n < 0$.

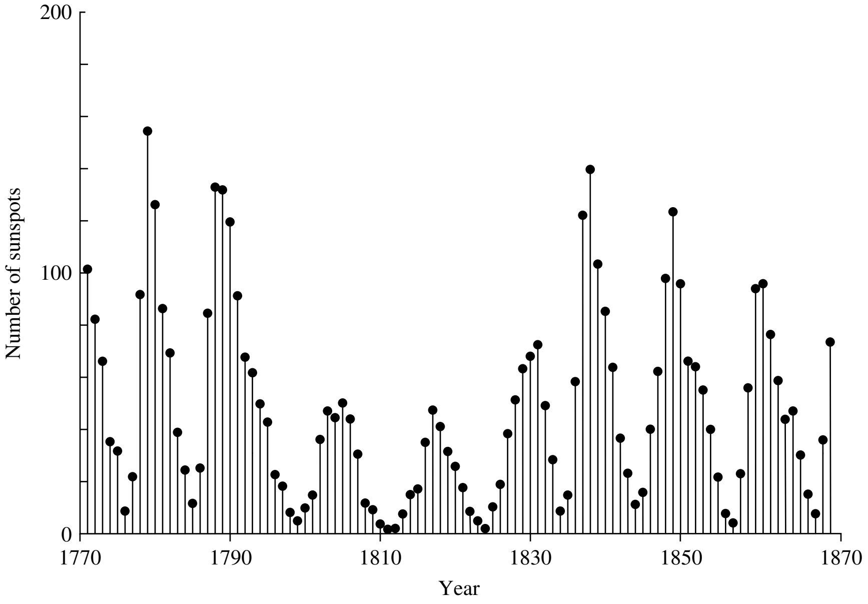


Figure 2.4 Wölfer annual sunspot numbers (1770–1869).

discrete-time signals. For example, the signal $x(n)$ in Fig. 2.3 can be obtained by sampling the analog signal $x(t) = 0.8^t$, $t \geq 0$ and $x(t) = 0$, $t < 0$ once every second.

2. By accumulating a variable over a period of time. For example, counting the number of cars using a given street every hour, or recording the value of gold every day, results in discrete-time signals. Figure 2.4 shows a graph of the Wölfer sunspot numbers. Each sample of this discrete-time signal provides the number of sunspots observed during an interval of 1 year.

2.3 Continuous-Valued Versus Discrete-Valued Signals

The values of a continuous-time or discrete-time signal can be continuous or discrete. If a signal takes on all possible values on a finite or an infinite range, it is said to be a continuous-valued signal. Alternatively, if the signal takes on values from a finite set of possible values, it is said to be a discrete-valued signal. Usually, these values are equidistant and hence can be expressed as an integer multiple of the distance between two successive values. A discrete-time signal having a set of discrete values is called a *digital signal*. Figure 2.5 shows a digital signal that takes on one of four possible values.

In order for a signal to be processed digitally, it must be discrete in time and its values must be discrete (i.e., it must be a digital signal). If the signal to be processed is in analog form, it is converted to a digital signal by sampling the analog signal at discrete instants in time, obtaining a discrete-time signal, and then by *quantizing* its values to a set of discrete values, as described later in the chapter. The process

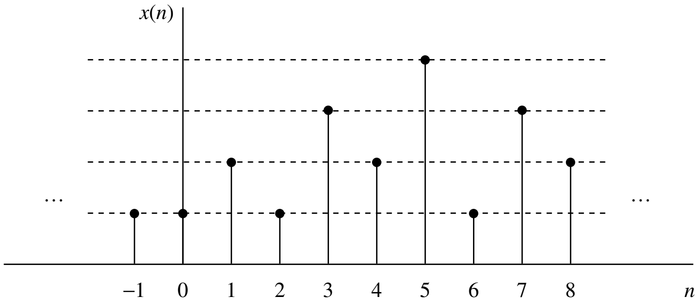


Figure 2.5 Digital signal with four different amplitude values.

of converting a continuous-valued signal into a discrete-valued signal, called *quantization*, is basically an approximation process. It may be accomplished simply by rounding or truncation. For example, if the allowable signal values in the digital signal are integers, say 0 through 15, the continuous-value signal is quantized into these integer values. Thus the signal value 8.58 will be approximated by the value 8 if the quantization process is performed by truncation or by 9 if the quantization process is performed by rounding to the nearest integer. An explanation of the analog-to-digital conversion process is given later in the chapter.

2.4 Deterministic Versus Random Signals

The mathematical analysis and processing of signals requires the availability of a mathematical description for the signal itself. This mathematical description, often referred to as the *signal model*, leads to another important classification of signals. Any signal that can be uniquely described by an explicit mathematical expression, a table of data, or a well-defined rule is called *deterministic*. This term is used to emphasize the fact that all past, present, and future values of the signal are known precisely, without any uncertainty.

In many practical applications, however, there are signals that either cannot be described to any reasonable degree of accuracy by explicit mathematical formulas, or such a description is too complicated to be of any practical use. The lack of such a relationship implies that such signals evolve in time in an unpredictable manner. We refer to these signals as *random*.The output of a noise generator, the seismic signal of Fig. 2.1, and the speech signal in Fig. 1.1 are examples of random signals.

The mathematical framework for the theoretical analysis of random signals is provided by the theory of probability and stochastic processes.

It should be emphasized at this point that the classification of a *real-world* signal as deterministic or random is not always clear. Sometimes, both approaches lead to meaningful results that provide more insight into signal behavior. At other times, the wrong classification may lead to erroneous results, since some mathematical tools may apply only to deterministic signals while others may apply only to random signals. This will become clearer as we examine specific mathematical tools.

3 The Concept of Frequency in Continuous-Time and Discrete-Time Signals

The concept of frequency is familiar to students in engineering and the sciences. This concept is basic in, for example, the design of a radio receiver, a high-fidelity system, or a spectral filter for color photography. From physics we know that frequency is closely related to a specific type of periodic motion called harmonic oscillation, which is described by sinusoidal functions. The concept of frequency is directly related to the concept of time. Actually, it has the dimension of inverse time. Thus we should expect that the nature of time (continuous or discrete) would affect the nature of the frequency accordingly.

3.1 Continuous-Time Sinusoidal Signals

A simple harmonic oscillation is mathematically described by the following continuous-time sinusoidal signal:

$$x_a(t) = A\cos(\Omega t + \theta), \quad -\infty < t < \infty \tag{3.1}$$

shown in Fig. 3.1. The subscript a used with $x(t)$ denotes an analog signal. This signal is completely characterized by three parameters: A is the *amplitude* of the sinusoid, Ω is the *frequency* in radians per second (rad/s), and θ is the *phase* in radians. Instead of Ω, we often use the frequency F in cycles per second or hertz (Hz), where

$$\Omega = 2\pi F \tag{3.2}$$

In terms of F, (3.1) can be written as

$$x_a(t) = A\cos(2\pi Ft + \theta), \quad -\infty < t < \infty \tag{3.3}$$

We will use both forms, (3.1) and (3.3), in representing sinusoidal signals.

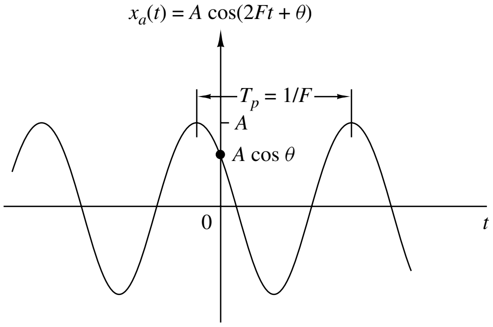


Figure 3.1 Example of an analog sinusoidal signal.

The analog sinusoidal signal in (3.3) is characterized by the following properties:

A1. For every fixed value of the frequency F, $x_a(t)$ is periodic. Indeed, it can easily be shown, using elementary trigonometry, that

$$x_a(t + T_p) = x_a(t)$$

where $T_p = 1/F$ is the fundamental period of the sinusoidal signal.

A2. Continuous-time sinusoidal signals with distinct (different) frequencies are themselves distinct.

A3. Increasing the frequency F results in an increase in the rate of oscillation of the signal, in the sense that more periods are included in a given time interval.

We observe that for $F = 0$, the value $T_p = \infty$ is consistent with the fundamental relation $F = 1/T_p$. Due to continuity of the time variable t, we can increase the frequency F, without limit, with a corresponding increase in the rate of oscillation.

The relationships we have described for sinusoidal signals carry over to the class of complex exponential signals

$$x_a(t) = Ae^{j(\Omega t+\theta)} \tag{3.4}$$

This can easily be seen by expressing these signals in terms of sinusoids using the Euler identity

$$e^{\pm j\phi} = \cos\phi \pm j\sin\phi \tag{3.5}$$

By definition, frequency is an inherently positive physical quantity. This is obvious if we interpret frequency as the number of cycles per unit time in a periodic signal. However, in many cases, only for mathematical convenience, we need to introduce negative frequencies. To see this we recall that the sinusoidal signal (3.1) may be expressed as

$$x_a(t) = A\cos(\Omega t + \theta) = \frac{A}{2}\, e^{j(\Omega t+\theta)} + \frac{A}{2}\, e^{-j(\Omega t+\theta)} \tag{3.6}$$

which follows from (3.5). Note that a sinusoidal signal can be obtained by adding two equal-amplitude complex-conjugate exponential signals, sometimes called phasors, illustrated in Fig. 3.2. As time progresses the phasors rotate in opposite directions with angular frequencies $\pm\Omega$ radians per second. Since a *positive frequency* corresponds to counterclockwise uniform angular motion, a *negative frequency* simply corresponds to clockwise angular motion.

For mathematical convenience, we use both negative and positive frequencies throughout this text. Hence the frequency range for analog sinusoids is $-\infty < F < \infty$.

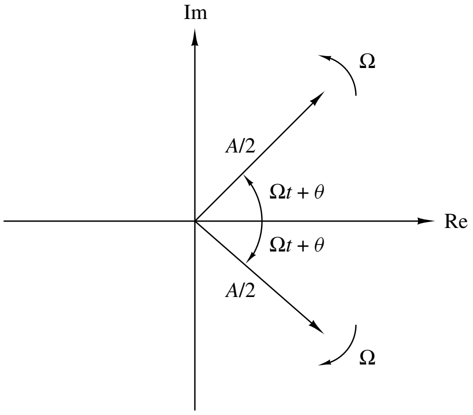


Figure 3.2
Representation of a cosine function by a pair of complex-conjugate exponentials (phasors).

3.2 Discrete-Time Sinusoidal Signals

A discrete-time sinusoidal signal may be expressed as

$$x(n) = A\cos(\omega n + \theta), \quad -\infty < n < \infty \tag{3.7}$$

where n is an integer variable, called the sample number, A is the *amplitude* of the sinusoid, ω is the *frequency* in radians per sample, and θ is the *phase* in radians.

If instead of ω we use the frequency variable f defined by

$$\omega \equiv 2\pi f \tag{3.8}$$

the relation (3.7) becomes

$$x(n) = A\cos(2\pi f n + \theta), \quad -\infty < n < \infty \tag{3.9}$$

The frequency f has dimensions of cycles per sample. In Section 4, where we consider the sampling of analog sinusoids, we relate the frequency variable f of a discrete-time sinusoid to the frequency F in cycles per second for the analog sinusoid. For the moment we consider the discrete-time sinusoid in (3.7) independently of the continuous-time sinusoid given in (3.1). Figure 3.3 shows a sinusoid with frequency $\omega = \pi/6$ radians per sample ($f = \frac{1}{12}$ cycles per sample) and phase $\theta = \pi/3$.

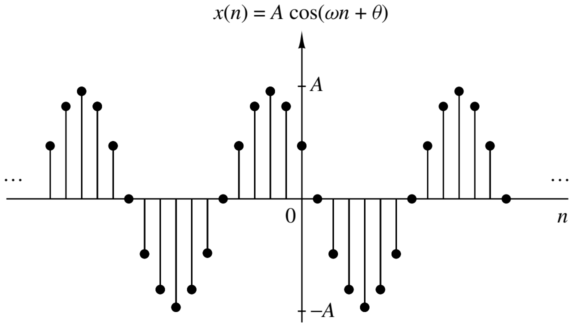


Figure 3.3
Example of a discrete-time sinusoidal signal ($\omega = \pi/6$ and $\theta = \pi/3$).

In contrast to continuous-time sinusoids, the discrete-time sinusoids are characterized by the following properties:

B1. *A discrete-time sinusoid is periodic only if its frequency f is a rational number.*

By definition, a discrete-time signal $x(n)$ is periodic with period $N(N > 0)$ if and only if

$$x(n+N) = x(n) \qquad \text{for all } n \tag{3.10}$$

The smallest value of N for which (3.10) is true is called the *fundamental period*.

The proof of the periodicity property is simple. For a sinusoid with frequency f_0 to be periodic, we should have

$$\cos[2\pi f_0(N+n)+\theta] = \cos(2\pi f_0 n + \theta)$$

This relation is true if and only if there exists an integer k such that

$$2\pi f_0 N = 2k\pi$$

or, equivalently,

$$f_0 = \frac{k}{N} \tag{3.11}$$

According to (3.11), a discrete-time sinusoidal signal is periodic only if its frequency f_0 can be expressed as the ratio of two integers (i.e., f_0 is rational).

To determine the fundamental period N of a periodic sinusoid, we express its frequency f_0 as in (3.11) and cancel common factors so that k and N are relatively prime. Then the fundamental period of the sinusoid is equal to N. Observe that a small change in frequency can result in a large change in the period. For example, note that $f_1 = 31/60$ implies that $N_1 = 60$, whereas $f_2 = 30/60$ results in $N_2 = 2$.

B2. *Discrete-time sinusoids whose frequencies are separated by an integer multiple of* 2π *are identical.*

To prove this assertion, let us consider the sinusoid $\cos(\omega_0 n + \theta)$. It easily follows that

$$\cos[(\omega_0 + 2\pi)n + \theta] = \cos(\omega_0 n + 2\pi n + \theta) = \cos(\omega_0 n + \theta) \tag{3.12}$$

As a result, all sinusoidal sequences

$$x_k(n) = A\cos(\omega_k n + \theta), \qquad k = 0, 1, 2, \ldots \tag{3.13}$$

where

$$\omega k = \omega_0 + 2k\pi, \qquad -\pi \le \omega_0 \le \pi$$

are *indistinguishable* (i.e., *identical*). Any sequence resulting from a sinusoid with a frequency $|\omega| > \pi$, or $|f| > \frac{1}{2}$, is identical to a sequence obtained from a sinusoidal signal with frequency $|\omega| < \pi$. Because of this similarity, we call the sinusoid having the frequency $|\omega| > \pi$ an *alias* of a corresponding sinusoid with frequency $|\omega| < \pi$. Thus we regard frequencies in the range $-\pi \le \omega \le \pi$, or $-\frac{1}{2} \le f \le \frac{1}{2}$, as unique and all frequencies $|\omega| > \pi$, or $|f| > \frac{1}{2}$, as aliases. The reader should notice the difference between discrete-time sinusoids and continuous-time sinusoids, where the latter result in distinct signals for Ω or F in the entire range $-\infty < \Omega < \infty$ or $-\infty < F < \infty$.

B3. *The highest rate of oscillation in a discrete-time sinusoid is attained when $\omega = \pi$ (or $\omega = -\pi$) or, equivalently, $f = \frac{1}{2}$ (or $f = -\frac{1}{2}$).*

To illustrate this property, let us investigate the characteristics of the sinusoidal signal sequence

$$x(n) = \cos \omega_0 n$$

when the frequency varies from 0 to π. To simplify the argument, we take values of $\omega_0 = 0, \pi/8, \pi/4, \pi/2, \pi$ corresponding to $f = 0, \frac{1}{16}, \frac{1}{8}, \frac{1}{4}, \frac{1}{2}$, which result in periodic sequences having periods $N = \infty$, 16, 8, 4, 2, as depicted in Fig. 3.4. We note that the period of the sinusoid decreases as the frequency increases. In fact, we can see that the rate of oscillation increases as the frequency increases.

To see what happens for $\pi \le \omega_0 \le 2\pi$, we consider the sinusoids with frequencies $\omega_1 = \omega_0$ and $\omega_2 = 2\pi - \omega_0$. Note that as ω_1 varies from π to 2π, ω_2 varies from π to 0. It can be easily seen that

$$\begin{aligned} x_1(n) &= A\cos\omega_1 n = A\cos\omega_0 n \\ x_2(n) &= A\cos\omega_2 n = A\cos(2\pi - \omega_0)n \\ &= A\cos(-\omega_0 n) = x_1(n) \end{aligned} \tag{3.14}$$

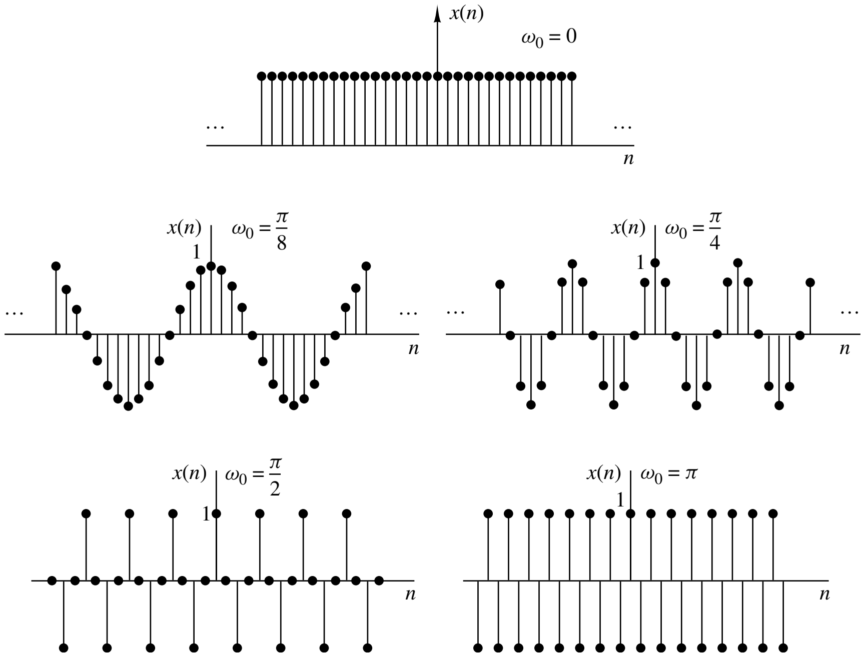


Figure 3.4 Signal $x(n) = \cos\omega_0 n$ for various values of the frequency ω_0.

Hence ω_2 is an alias of ω_1. If we had used a sine function instead of a cosine function, the result would basically be the same, except for a 180° phase difference between the sinusoids $x_1(n)$ and $x_2(n)$. In any case, as we increase the relative frequency ω_0 of a discrete-time sinusoid from π to 2π, its rate of oscillation decreases. For $\omega_0 = 2\pi$ the result is a constant signal, as in the case for $\omega_0 = 0$. Obviously, for $\omega_0 = \pi$ (or $f = \frac{1}{2}$) we have the highest rate of oscillation.

As for the case of continuous-time signals, negative frequencies can be introduced as well for discrete-time signals. For this purpose we use the identity

$$x(n) = A\cos(\omega n + \theta) = \frac{A}{2}\, e^{j(\omega n+\theta)} + \frac{A}{2}\, e^{-j(\omega n+\theta)} \tag{3.15}$$

Since discrete-time sinusoidal signals with frequencies that are separated by an integer multiple of 2π are identical, it follows that the frequencies in any interval $\omega_1 \leq \omega \leq \omega_1 + 2\pi$ constitute *all* the existing discrete-time sinusoids or complex exponentials. Hence the frequency range for discrete-time sinusoids is finite with duration 2π. Usually, we choose the range $0 \leq \omega \leq 2\pi$ or $-\pi \leq \omega \leq \pi$ ($0 \leq f \leq 1$, $-\frac{1}{2} \leq f \leq \frac{1}{2}$), which we call the *fundamental range*.

3.3 Harmonically Related Complex Exponentials

Sinusoidal signals and complex exponentials play a major role in the analysis of signals and systems. In some cases we deal with sets of *harmonically related* complex exponentials (or sinusoids). These are sets of periodic complex exponentials with fundamental frequencies that are multiples of a single positive frequency. Although we confine our discussion to complex exponentials, the same properties clearly hold for sinusoidal signals. We consider harmonically related complex exponentials in both continuous time and discrete time.

Continuous-time exponentials. The basic signals for continuous-time, harmonically related exponentials are

$$s_k(t) = e^{jk\Omega_0 t} = e^{j2\pi kF_0 t} \qquad k = 0, \pm 1, \pm 2, \ldots \tag{3.16}$$

We note that for each value of k, $s_k(t)$ is periodic with fundamental period $1/(kF_0) = T_p/k$ or fundamental frequency kF_0. Since a signal that is periodic with period T_p/k is also periodic with period $k(T_p/k) = T_p$ for any positive integer k, we see that all of the $s_k(t)$ have a common period of T_p. F_0 is allowed to take any value and all members of the set are distinct, in the sense that if $k_1 \neq k_2$, then $s_{k1}(t) \neq s_{k2}(t)$.

From the basic signals in (3.16) we can construct a linear combination of harmonically related complex exponentials of the form

$$x_a(t) = \sum_{k=-\infty}^{\infty} c_k s_k(t) = \sum_{k=-\infty}^{\infty} c_k e^{jk\Omega_0 t} \tag{3.17}$$

where c_k, $k = 0, \pm 1, \pm 2, \ldots$ are arbitrary complex constants. The signal $x_a(t)$ is periodic with fundamental period $T_p = 1/F_0$, and its representation in terms of

(3.17) is called the *Fourier series* expansion for $x_a(t)$. The complex-valued constants are the Fourier series coefficients and the signal $s_k(t)$ is called the kth harmonic of $x_a(t)$.

Discrete-time exponentials. Since a discrete-time complex exponential is periodic if its relative frequency is a rational number, we choose $f_0 = 1/N$ and we define the sets of harmonically related complex exponentials by

$$s_k(n) = e^{j2\pi k f_0 n}, \qquad k = 0, \pm 1, \pm 2, \ldots \tag{3.18}$$

In contrast to the continuous-time case, we note that

$$s_{k+N}(n) = e^{j2\pi n(k+N)/N} = e^{j2\pi n} s_k(n) = sk(n)$$

This means that, consistent with (3.10), there are only N distinct periodic complex exponentials in the set described by (3.18). Furthermore, all members of the set have a common period of N samples. Clearly, we can choose any consecutive N complex exponentials, say from $k = n_0$ to $k = n_0 + N - 1$, to form a harmonically related set with fundamental frequency $f_0 = 1/N$. Most often, for convenience, we choose the set that corresponds to $n_0 = 0$, that is, the set

$$s_k(n) = e^{j2\pi kn/N}, \qquad k = 0, 1, 2, \ldots, N-1 \tag{3.19}$$

As in the case of continuous-time signals, it is obvious that the linear combination

$$x(n) = \sum_{k=0}^{N-1} c_k s_k(n) = \sum_{k=0}^{N-1} c_k e^{j2\pi kn/N} \tag{3.20}$$

results in a periodic signal with fundamental period N. As we shall see later, this is the Fourier series representation for a periodic discrete-time sequence with Fourier coefficients $\{c_k\}$. The sequence $s_k(n)$ is called the kth harmonic of $x(n)$.

EXAMPLE 3.1

Stored in the memory of a digital signal processor is one cycle of the sinusoidal signal

$$x(n) = \sin\left(\frac{2\pi n}{N} + \theta\right)$$

where $\theta = 2\pi q/N$, where q and N are integers.

(a) Determine how this table of values can be used to obtain values of harmonically related sinusoids having the same phase.

(b) Determine how this table can be used to obtain sinusoids of the same frequency but different phase.

Solution.

(a) Let $x_k(n)$ denote the sinusoidal signal sequence

$$x_k(n) = \sin\left(\frac{2\pi nk}{N} + \theta\right)$$

This is a sinusoid with frequency $f_k = k/N$, which is harmonically related to $x(n)$. But $x_k(n)$ may be expressed as

$$\begin{aligned} x_k(n) &= \sin\left[\frac{2\pi(kn)}{N} + \theta\right] \\ &= x(kn) \end{aligned}$$

Thus we observe that $x_k(0) = x(0)$, $x_k(1) = x(k)$, $x_k(2) = x(2k)$, and so on. Hence the sinusoidal sequence $x_k(n)$ can be obtained from the table of values of $x(n)$ by taking every kth value of $x(n)$, beginning with $x(0)$. In this manner we can generate the values of all harmonically related sinusoids with frequencies $f_k = k/N$ for $k = 0, 1, \ldots, N-1$.

(b) We can control the phase θ of the sinusoid with frequency $f_k = k/N$ by taking the first value of the sequence from memory location $q = \theta N/2\pi$, where q is an integer. Thus the initial phase θ controls the starting location in the table and we wrap around the table each time the index (kn) exceeds N.

4 Analog-to-Digital and Digital-to-Analog Conversion

Most signals of practical interest, such as speech, biological signals, seismic signals, radar signals, sonar signals, and various communications signals such as audio and video signals, are analog. To process analog signals by digital means, it is first necessary to convert them into digital form, that is, to convert them to a sequence of numbers having finite precision. This procedure is called *analog-to-digital (A/D) conversion*, and the corresponding devices are called *A/D converters* (*ADCs*).

Conceptually, we view A/D conversion as a three-step process. This process is illustrated in Fig. 4.1.

1. *Sampling*. This is the conversion of a continuous-time signal into a discrete-time signal obtained by taking "samples" of the continuous-time signal at discrete-time instants. Thus, if $x_a(t)$ is the input to the sampler, the output is $x_a(nT) \equiv x(n)$, where T is called the *sampling interval*.

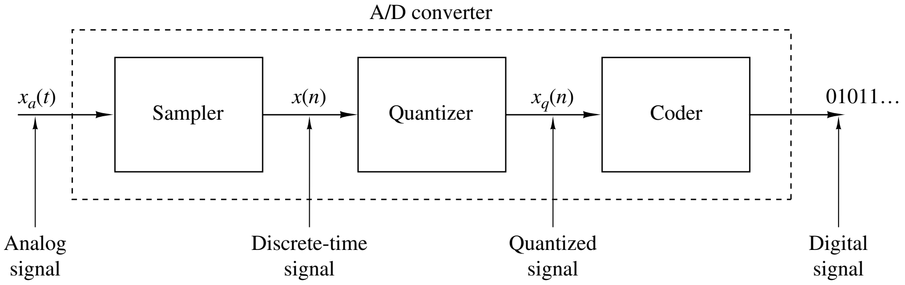


Figure 4.1 Basic parts of an analog-to-digital (A/D) converter.

2. *Quantization.* This is the conversion of a discrete-time continuous-valued signal into a discrete-time, discrete-valued (digital) signal. The value of each signal sample is represented by a value selected from a finite set of possible values. The difference between the unquantized sample $x(n)$ and the quantized output $x_q(n)$ is called the quantization error.

3. *Coding.* In the coding process, each discrete value $x_q(n)$ is represented by a b-bit binary sequence.

Although we model the A/D converter as a sampler followed by a quantizer and coder, in practice the A/D conversion is performed by a single device that takes $x_a(t)$ and produces a binary-coded number. The operations of sampling and quantization can be performed in either order but, in practice, sampling is always performed before quantization.

In many cases of practical interest (e.g., speech processing) it is desirable to convert the processed digital signals into analog form. (Obviously, we cannot listen to the sequence of samples representing a speech signal or see the numbers corresponding to a TV signal.) The process of converting a digital signal into an analog signal is known as *digital-to-analog (D/A) conversion.* All D/A converters "connect the dots" in a digital signal by performing some kind of interpolation, whose accuracy depends on the quality of the D/A conversion process. Figure 4.2 illustrates a simple form of D/A conversion, called a zero-order hold or a staircase approximation. Other approximations are possible, such as linearly connecting a pair of successive samples (linear interpolation), fitting a quadratic through three successive samples (quadratic interpolation), and so on. Is there an optimum (ideal) interpolator? For signals having a *limited frequency content* (finite bandwidth), the sampling theorem introduced in the following section specifies the optimum form of interpolation.

Sampling and quantization are treated in this section. In particular, we demonstrate that sampling does not result in a loss of information, nor does it introduce distortion in the signal if the signal bandwidth is finite. In principle, the analog signal can be reconstructed from the samples, provided that the sampling rate is sufficiently high to avoid the problem commonly called *aliasing.* On the other hand, quantization

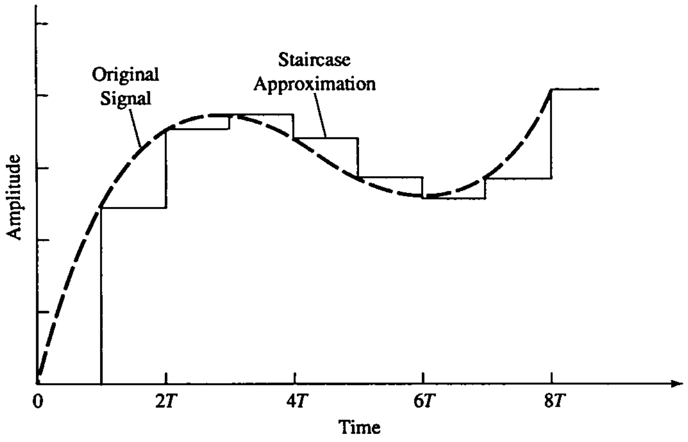


Figure 4.2 Zero-order hold digital-to-analog (D/A) conversion.

is a noninvertible or irreversible process that results in signal distortion. We shall show that the amount of distortion is dependent on the accuracy, as measured by the number of bits, in the A/D conversion process. The factors affecting the choice of the desired accuracy of the A/D converter are cost and sampling rate. In general, the cost increases with an increase in accuracy and/or sampling rate.

4.1 Sampling of Analog Signals

There are many ways to sample an analog signal. We limit our discussion to *periodic* or *uniform sampling*, which is the type of sampling used most often in practice. This is described by the relation

$$x(n) = x_a(nT), \qquad -\infty < n < \infty \tag{4.1}$$

where $x(n)$ is the discrete-time signal obtained by "taking samples" of the analog signal $x_a(t)$ every T seconds. This procedure is illustrated in Fig. 4.3. The time interval T between successive samples is called the *sampling period* or *sample interval* and its reciprocal $1/T = F_s$ is called the *sampling rate* (samples per second) or the *sampling frequency* (hertz).

Periodic sampling establishes a relationship between the time variables t and n of continuous-time and discrete-time signals, respectively. Indeed, these variables are linearly related through the sampling period T or, equivalently, through the sampling rate $F_s = 1/T$, as

$$t = nT = \frac{n}{F_s} \tag{4.2}$$

As a consequence of (4.2), there exists a relationship between the frequency variable F (or Ω) for analog signals and the frequency variable f (or ω) for discrete-time signals. To establish this relationship, consider an analog sinusoidal signal of the form

$$x_a(t) = A\cos(2\pi Ft + \theta) \tag{4.3}$$

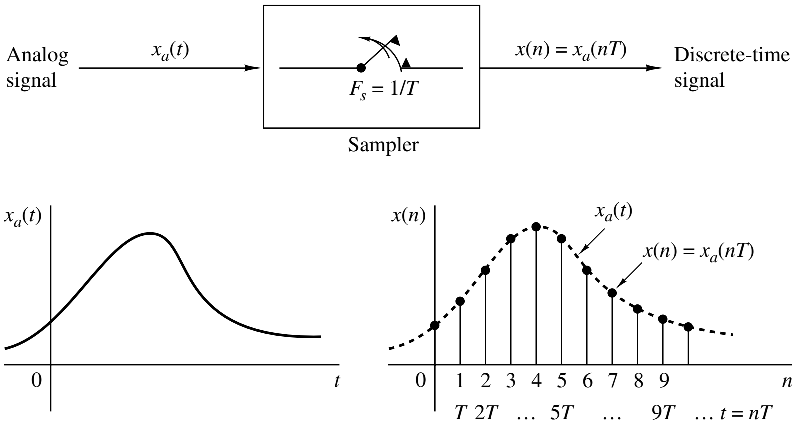


Figure 4.3 Periodic sampling of an analog signal.

which, when sampled periodically at a rate $F_s = 1/T$ samples per second, yields

$$\begin{aligned} x_a(nT) \equiv x(n) &= A\cos(2\pi FnT + \theta) \\ &= A\cos\left(\frac{2\pi nF}{F_s} + \theta\right) \end{aligned} \tag{4.4}$$

If we compare (4.4) with (3.9), we note that the frequency variables F and f are linearly related as

$$f = \frac{F}{F_s} \tag{4.5}$$

or, equivalently, as

$$\omega = \Omega T \tag{4.6}$$

The relation in (4.5) justifies the name *relative* or *normalized frequency*, which is sometimes used to describe the frequency variable f. As (4.5) implies, we can use f to determine the frequency F in hertz only if the sampling frequency F_s is known.

We recall from Section 3.1 that the ranges of the frequency variables F or Ω for continuous-time sinusoids are

$$\begin{aligned} -\infty < F < \infty \\ -\infty < \Omega < \infty \end{aligned} \tag{4.7}$$

However, the situation is different for discrete-time sinusoids. From Section 3.2 we recall that

$$\begin{aligned} -\frac{1}{2} < f < \frac{1}{2} \\ -\pi < \omega < \pi \end{aligned} \tag{4.8}$$

By substituting from (4.5) and (4.6) into (4.8), we find that the frequency of the continuous-time sinusoid when sampled at a rate $F_s = 1/T$ must fall in the range

$$-\frac{1}{2T} = -\frac{F_s}{2} \le F \le \frac{F_s}{2} = \frac{1}{2T} \tag{4.9}$$

or, equivalently,

$$-\frac{\pi}{T} = -\pi F_s \le \Omega \le \pi F_s = \frac{\pi}{T} \tag{4.10}$$

These relations are summarized in Table 1.

From these relations we observe that the fundamental difference between continuous-time and discrete-time signals is in their range of values of the frequency variables F and f, or Ω and ω. Periodic sampling of a continuous-time signal implies a mapping of the infinite frequency range for the variable F (or Ω) into a finite frequency range for the variable f (or ω). Since the highest frequency in a

TABLE 1 Relations Among Frequency Variables

Continuous-time signals		Discrete-time signals
$\Omega = 2\pi F$		$\omega = 2\pi f$
$\frac{\text{radians}}{\text{sec}}$ Hz		$\frac{\text{radians}}{\text{sample}}$ $\frac{\text{cycles}}{\text{sample}}$
	$\xrightarrow{\omega = \Omega T, f = F/F_s}$	$-\pi \le \omega \le \pi$
	$\xleftarrow{\Omega = \omega/T, F = f \cdot F_s}$	$-\frac{1}{2} \le f \le \frac{1}{2}$
$-\infty < \Omega < \infty$		$-\pi/T \le \Omega \le \pi/T$
$-\infty < F < \infty$		$-F_2/2 \le F \le F_s/2$

discrete-time signal is $\omega = \pi$ or $f = \frac{1}{2}$, it follows that, with a sampling rate F_s, the corresponding highest values of F and Ω are

$$\begin{aligned} F_{\text{max}} &= \frac{F_s}{2} = \frac{1}{2T} \\ \Omega_{\text{max}} &= \pi F_s = \frac{\pi}{T} \end{aligned} \tag{4.11}$$

Therefore, sampling introduces an ambiguity, since the highest frequency in a continuous-time signal that can be uniquely distinguished when such a signal is sampled at a rate $F_s = 1/T$ is $F_{\text{max}} = F_s/2$, or $\Omega_{\text{max}} = \pi F_s$. To see what happens to frequencies above $F_s/2$, let us consider the following example.

EXAMPLE 4.1

The implications of these frequency relations can be fully appreciated by considering the two analog sinusoidal signals

$$\begin{aligned} x_1(t) &= \cos 2\pi(10)t \\ x_2(t) &= \cos 2\pi(50)t \end{aligned} \tag{4.12}$$

which are sampled at a rate $F_s = 40$ Hz. The corresponding discrete-time signals or sequences are

$$\begin{aligned} x_1(n) &= \cos 2\pi \left(\frac{10}{40}\right) n = \cos \frac{\pi}{2} n \\ x_2(n) &= \cos 2\pi \left(\frac{50}{40}\right) n = \cos \frac{5\pi}{2} n \end{aligned} \tag{4.13}$$

However, $\cos 5\pi n/2 = \cos(2\pi n + \pi n/2) = \cos \pi n/2$. Hence $x_2(n) = x_1(n)$. Thus the sinusoidal signals are identical and, consequently, indistinguishable. If we are given the sampled values generated by $\cos(\pi/2)n$, there is some ambiguity as to whether these sampled values correspond to $x_1(t)$ or $x_2(t)$. Since $x_2(t)$ yields exactly the same values as $x_1(t)$ when the two are sampled at $F_s = 40$ samples per second, we say that the frequency $F_2 = 50$ Hz is an *alias* of the frequency $F_1 = 10$ Hz at the sampling rate of 40 samples per second.

It is important to note that F_2 is not the only alias of F_1. In fact at the sampling rate of 40 samples per second, the frequency $F_3 = 90$ Hz is also an alias of F_1, as is the frequency $F_4 = 130$ Hz, and so on. All of the sinusoids $\cos 2\pi(F_1 + 40k)t$, $k = 1, 2, 3, 4, \ldots$, sampled at 40 samples per second, yield identical values. Consequently, they are all aliases of $F_1 = 10$ Hz.

In general, the sampling of a continuous-time sinusoidal signal

$$x_a(t) = A\cos(2\pi F_0 t + \theta) \tag{4.14}$$

with a sampling rate $F_s = 1/T$ results in a discrete-time signal

$$x(n) = A\cos(2\pi f_0 n + \theta) \tag{4.15}$$

where $f_0 = F_0/F_s$ is the relative frequency of the sinusoid. If we assume that $-F_s/2 \le F_0 \le F_s/2$, the frequency f_0 of $x(n)$ is in the range $-\frac{1}{2} \le f_0 \le \frac{1}{2}$, which is the frequency range for discrete-time signals. In this case, the relationship between F_0 and f_0 is one-to-one, and hence it is possible to identify (or reconstruct) the analog signal $x_a(t)$ from the samples $x(n)$.

On the other hand, if the sinusoids

$$x_a(t) = A\cos(2\pi F_k t + \theta) \tag{4.16}$$

where

$$F_k = F_0 + kF_s, \qquad k = \pm 1, \pm 2, \ldots \tag{4.17}$$

are sampled at a rate F_s, it is clear that the frequency F_k is outside the fundamental frequency range $-F_s/2 \le F \le F_s/2$. Consequently, the sampled signal is

$$\begin{aligned} x(n) \equiv x_a(nT) &= A\cos\left(2\pi \frac{F_0 + kF_s}{F_s} n + \theta\right) \\ &= A\cos(2\pi n F_0/F_s + \theta + 2\pi k n) \\ &= A\cos(2\pi f_0 n + \theta) \end{aligned}$$

which is identical to the discrete-time signal in (4.15) obtained by sampling (4.14). Thus an infinite number of continuous-time sinusoids is represented by sampling the *same* discrete-time signal (i.e., by the same set of samples). Consequently, if we are given the sequence $x(n)$, an ambiguity exists as to which continuous-time signal $x_a(t)$ these values represent. Equivalently, we can say that the frequencies $F_k = F_0 + kF_s$, $-\infty < k < \infty$ (k integer) are indistinguishable from the frequency F_0 after sampling and hence they are aliases of F_0. The relationship between the frequency variables of the continuous-time and discrete-time signals is illustrated in Fig. 4.4.

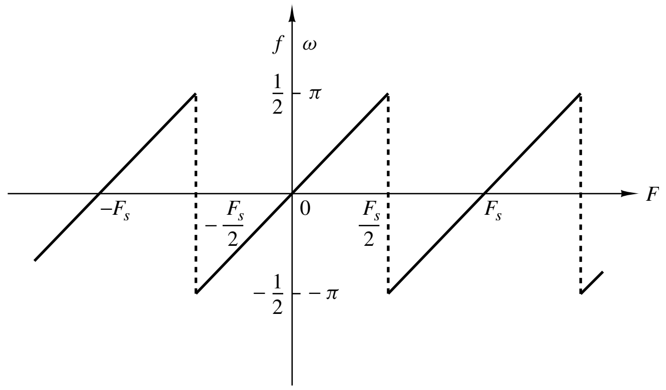


Figure 4.4 Relationship between the continuous-time and discrete-time frequency variables in the case of periodic sampling.

An example of aliasing is illustrated in Fig. 4.5, where two sinusoids with frequencies $F_0 = \frac{1}{8}$ Hz and $F_1 = -\frac{7}{8}$ Hz yield identical samples when a sampling rate of $F_s = 1$ Hz is used. From (4.17) it easily follows that for $k = -1$, $F_0 = F_1 + F_s = (-\frac{7}{8} + 1)$ Hz $= \frac{1}{8}$ Hz.

Since $F_s/2$, which corresponds to $\omega = \pi$, is the highest frequency that can be represented uniquely with a sampling rate F_s, it is a simple matter to determine the mapping of any (alias) frequency above $F_s/2$ ($\omega = \pi$) into the equivalent frequency below $F_s/2$. We can use $F_s/2$ or $\omega = \pi$ as the pivotal point and reflect or "fold" the alias frequency to the range $0 \leq \omega \leq \pi$. Since the point of reflection is $F_s/2$ ($\omega = \pi$), the frequency $F_s/2$ ($\omega = \pi$) is called the *folding frequency*.

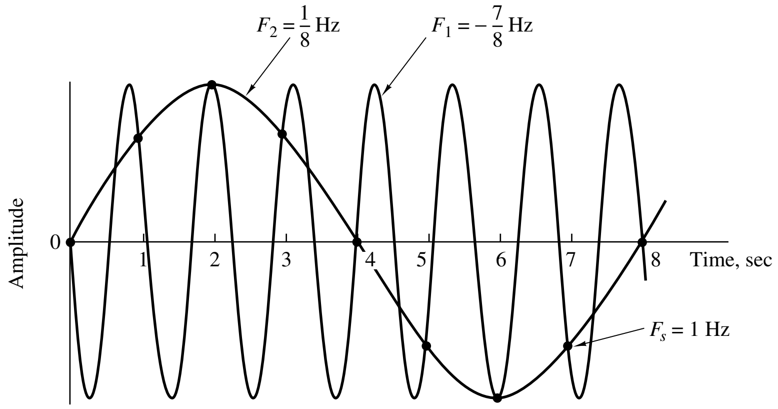


Figure 4.5 Illustration of aliasing.

EXAMPLE 4.2

Consider the analog signal

$$x_a(t) = 3 \cos 100\pi t$$

(a) Determine the minimum sampling rate required to avoid aliasing.

(b) Suppose that the signal is sampled at the rate $F_s = 200$ Hz. What is the discrete-time signal obtained after sampling?

(c) Suppose that the signal is sampled at the rate $F_s = 75$ Hz. What is the discrete-time signal obtained after sampling?

(d) What is the frequency $0 < F < F_s/2$ of a sinusoid that yields samples identical to those obtained in part (c)?

Solution.

(a) The frequency of the analog signal is $F = 50$ Hz. Hence the minimum sampling rate required to avoid aliasing is $F_s = 100$ Hz.

(b) If the signal is sampled at $F_s = 200$ Hz, the discrete-time signal is

$$x(n) = 3\cos\frac{100\pi}{200}n = 3\cos\frac{\pi}{2}n$$

(c) If the signal is sampled at $F_s = 75$ Hz, the discrete-time signal is

$$\begin{aligned} x(n) &= 3\cos\frac{100\pi}{75}n = 3\cos\frac{4\pi}{3}n \\ &= 3\cos\left(2\pi - \frac{2\pi}{3}\right)n \\ &= 3\cos\frac{2\pi}{3}n \end{aligned}$$

(d) For the sampling rate of $F_s = 75$ Hz, we have

$$F = fF_s = 75f$$

The frequency of the sinusoid in part (c) is $f = \frac{1}{3}$. Hence

$$F = 25 \text{ Hz}$$

Clearly, the sinusoidal signal

$$\begin{aligned} y_a(t) &= 3\cos 2\pi Ft \\ &= 3\cos 50\pi t \end{aligned}$$

sampled at $F_s = 75$ samples/s yields identical samples. Hence $F = 50$ Hz is an alias of $F = 25$ Hz for the sampling rate $F_s = 75$ Hz.

4.2 The Sampling Theorem

Given any analog signal, how should we select the sampling period T or, equivalently, the sampling rate F_s? To answer this question, we must have some information about the characteristics of the signal to be sampled. In particular, we must have some general information concerning the *frequency content* of the signal. Such information is generally available to us. For example, we know generally that the major frequency components of a speech signal fall below 3000 Hz. On the other hand, television signals, in general, contain important frequency components up to 5 MHz. The information content of such signals is contained in the amplitudes, frequencies, and phases of the various frequency components, but detailed knowledge of the characteristics of such signals is not available to us prior to obtaining the signals. In fact, the purpose of processing the signals is usually to extract this detailed information. However, if we know the maximum frequency content of the general class of signals (e.g., the class of speech signals, the class of video signals, etc.), we can specify the sampling rate necessary to convert the analog signals to digital signals.

Let us suppose that any analog signal can be represented as a sum of sinusoids of different amplitudes, frequencies, and phases, that is,

$$x_a(t) = \sum_{i=1}^{N} A_i \cos(2\pi F_i t + \theta_i) \tag{4.18}$$

where N denotes the number of frequency components. All signals, such as speech and video, lend themselves to such a representation over any short time segment. The amplitudes, frequencies, and phases usually change slowly with time from one time segment to another. However, suppose that the frequencies do not exceed some known frequency, say $F_{\max}$. For example, $F_{\max} = 3000$ Hz for the class of speech signals and $F_{\max} = 5$ MHz for television signals. Since the maximum frequency may vary slightly from different realizations among signals of any given class (e.g., it may vary slightly from speaker to speaker), we may wish to ensure that $F_{\max}$ does not exceed some predetermined value by passing the analog signal through a filter that severely attenuates frequency components above $F_{\max}$. Thus we are certain that no signal in the class contains frequency components (having significant amplitude or power) above $F_{\max}$. In practice, such filtering is commonly used prior to sampling.

From our knowledge of $F_{\max}$, we can select the appropriate sampling rate. We know that the highest frequency in an analog signal that can be unambiguously reconstructed when the signal is sampled at a rate $F_s = 1/T$ is $F_s/2$. Any frequency above $F_s/2$ or below $-F_s/2$ results in samples that are identical with a corresponding frequency in the range $-F_s/2 \le F \le F_s/2$. To avoid the ambiguities resulting from aliasing, we must select the sampling rate to be sufficiently high. That is, we must select $F_s/2$ to be greater than $F_{\max}$. Thus to avoid the problem of aliasing, F_s is selected so that

$$F_s > 2F_{\max} \tag{4.19}$$

where $F_{\max}$ is the largest frequency component in the analog signal. With the sampling rate selected in this manner, any frequency component, say $|F_i| < F_{\max}$, in the analog signal is mapped into a discrete-time sinusoid with a frequency

$$-\frac{1}{2} \le f_i = \frac{F_i}{F_s} \le \frac{1}{2} \tag{4.20}$$

or, equivalently,

$$-\pi \le \omega_i = 2\pi f_i \le \pi \tag{4.21}$$

Since, $|f| = \frac{1}{2}$ or $|\omega| = \pi$ is the highest (unique) frequency in a discrete-time signal, the choice of sampling rate according to (4.19) avoids the problem of aliasing.

In other words, the condition $F_s > 2F_{\max}$ ensures that all the sinusoidal components in the analog signal are mapped into corresponding discrete-time frequency components with frequencies in the fundamental interval. Thus all the frequency components of the analog signal are represented in sampled form without ambiguity, and hence the analog signal can be reconstructed without distortion from the sample values using an "appropriate" interpolation (digital-to-analog conversion) method. The "appropriate" or ideal interpolation formula is specified by the *sampling theorem*.

Sampling Theorem. If the highest frequency contained in an analog signal $x_a(t)$ is $F_{\max} = B$ and the signal is sampled at a rate $F_s > 2F_{\max} \equiv 2B$, then $x_a(t)$ can be exactly recovered from its sample values using the interpolation function

$$g(t) = \frac{\sin 2\pi Bt}{2\pi Bt} \tag{4.22}$$

Thus $x_a(t)$ may be expressed as

$$x_a(t) = \sum_{n=-\infty}^{\infty} x_a\left(\frac{n}{F_s}\right) g\left(t - \frac{n}{F_s}\right) \tag{4.23}$$

where $x_a(n/F_s) = x_a(nT) \equiv x(n)$ are the samples of $x_a(t)$.

When the sampling of $x_a(t)$ is performed at the minimum sampling rate $F_s = 2B$, the reconstruction formula in (4.23) becomes

$$x_a(t) = \sum_{n=-\infty}^{\infty} x_a\left(\frac{n}{2B}\right) \frac{\sin 2\pi B(t - n/2B)}{2\pi B(t - n/2B)} \tag{4.24}$$

The sampling rate $F_N = 2B = 2F_{\max}$ is called the *Nyquist rate*. Figure 4.6 illustrates the ideal D/A conversion process using the interpolation function in (4.22).

As can be observed from either (4.23) or (4.24), the reconstruction of $x_a(t)$ from the sequence $x(n)$ is a complicated process, involving a weighted sum of the interpolation function $g(t)$ and its time-shifted versions $g(t - nT)$ for $-\infty < n < \infty$, where the weighting factors are the samples $x(n)$. Because of the complexity and the infinite number of samples required in (4.23) or (4.24), these reconstruction formulas are primarily of theoretical interest.

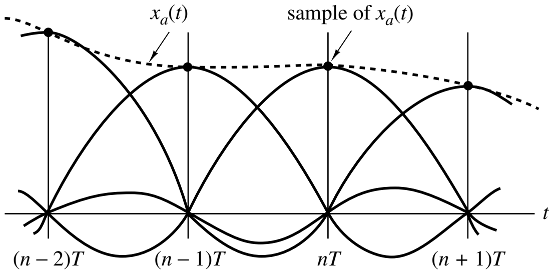


Figure 4.6
Ideal D/A conversion (interpolation).

EXAMPLE 4.3

Consider the analog signal

$$x_a(t) = 3\cos 50\pi t + 10\sin 300\pi t - \cos 100\pi t$$

What is the Nyquist rate for this signal?

Solution. The frequencies present in the signal above are

$$F_1 = 25 \text{ Hz}, \qquad F_2 = 150 \text{ Hz}, \qquad F_3 = 50 \text{ Hz}$$

Thus $F_{\max} = 150$ Hz and according to (4.19),

$$F_s > 2F_{\max} = 300 \text{ Hz}$$

The Nyquist rate is $F_N = 2F_{\max}$. Hence

$$F_N = 300 \text{ Hz}$$

Discussion. It should be observed that the signal component $10\sin 300\pi t$, sampled at the Nyquist rate $F_N = 300$, results in the samples $10\sin \pi n$, which are identically zero. In other words, we are sampling the analog sinusoid at its zero-crossing points, and hence we miss this signal component completely. This situation does not occur if the sinusoid is offset in phase by some amount θ. In such a case we have $10\sin(300\pi t + \theta)$ sampled at the Nyquist rate $F_N = 300$ samples per second, which yields the samples

$$\begin{aligned} 10\sin(\pi n + \theta) &= 10(\sin \pi n \cos\theta + \cos \pi n \sin\theta) \\ &= 10\sin\theta \cos \pi n \\ &= (-1)^n 10 \sin\theta \end{aligned}$$

Thus if $\theta \neq 0$ or π, the samples of the sinusoid taken at the Nyquist rate are not all zero. However, we still cannot obtain the correct amplitude from the samples when the phase θ is unknown. A simple remedy that avoids this potentially troublesome situation is to sample the analog signal at a rate higher than the Nyquist rate.

EXAMPLE 4.4

Consider the analog signal

$$x_a(t) = 3\cos 2000\pi t + 5\sin 6000\pi t + 10\cos 12{,}000\pi t$$

(a) What is the Nyquist rate for this signal?

(b) Assume now that we sample this signal using a sampling rate $F_s = 5000$ samples/s. What is the discrete-time signal obtained after sampling?

(c) What is the analog signal $y_a(t)$ that we can reconstruct from the samples if we use ideal interpolation?

Solution.

(a) The frequencies existing in the analog signal are

$$F_1 = 1 \text{ kHz}, \qquad F_2 = 3 \text{ kHz}, \qquad F_3 = 6 \text{ kHz}$$

Thus $F_{\max} = 6$ kHz, and according to the sampling theorem,

$$F_s > 2F_{\max} = 12 \text{ kHz}$$

The Nyquist rate is

$$F_N = 12 \text{ kHz}$$

(b) Since we have chosen $F_s = 5$ kHz, the folding frequency is

$$\frac{F_s}{2} = 2.5 \text{ kHz}$$

and this is the maximum frequency that can be represented uniquely by the sampled signal. By making use of (4.2) we obtain

$$\begin{aligned} x(n) = x_a(nT) &= x_a\left(\frac{n}{F_s}\right) \\ &= 3\cos 2\pi\left(\tfrac{1}{5}\right)n + 5\sin 2\pi\left(\tfrac{3}{5}\right)n + 10\cos 2\pi\left(\tfrac{6}{5}\right)n \\ &= 3\cos 2\pi\left(\tfrac{1}{5}\right)n + 5\sin 2\pi\left(1-\tfrac{2}{5}\right)n + 10\cos 2\pi\left(1+\tfrac{1}{5}\right)n \\ &= 3\cos 2\pi\left(\tfrac{1}{5}\right)n + 5\sin 2\pi\left(-\tfrac{2}{5}\right)n + 10\cos 2\pi\left(\tfrac{1}{5}\right)n \end{aligned}$$

Finally, we obtain

$$x(n) = 13\cos 2\pi\left(\tfrac{1}{5}\right)n - 5\sin 2\pi\left(\tfrac{2}{5}\right)n$$

The same result can be obtained using Fig. 4.4. Indeed, since $F_s = 5$ kHz, the folding frequency is $F_s/2 = 2.5$ kHz. This is the maximum frequency that can be represented uniquely by the sampled signal. From (4.17) we have $F_0 = F_k - kF_s$. Thus F_0 can be obtained by subtracting from F_k an integer multiple of F_s such that $-F_s/2 \le F_0 \le F_s/2$. The frequency F_1 is less than $F_s/2$ and thus it is not affected by aliasing. However, the other two frequencies are above the folding frequency and they will be changed by the aliasing effect. Indeed,

$$F_2' = F_2 - F_s = -2 \text{ kHz}$$

$$F_3' = F_3 - F_s = 1 \text{ kHz}$$

From (4.5) it follows that $f_1 = \frac{1}{5}$, $f_2 = -\frac{2}{5}$, and $f_3 = \frac{1}{5}$, which are in agreement with the result above.

(c) Since the frequency components at only 1 kHz and 2 kHz are present in the sampled signal, the analog signal we can recover is

$$ya(t) = 13\cos 2000\pi t - 5\sin 4000\pi t$$

which is obviously different from the original signal $x_a(t)$. This distortion of the original analog signal was caused by the aliasing effect, due to the low sampling rate used.

Although aliasing is a pitfall to be avoided, there are two useful practical applications based on the exploitation of the aliasing effect. These applications are the stroboscope and the sampling oscilloscope. Both instruments are designed to operate as aliasing devices in order to represent high frequencies as low frequencies.

To elaborate, consider a signal with high-frequency components confined to a given frequency band $B_1 < F < B_2$, where $B_2 - B_1 \equiv B$ is defined as the bandwidth of the signal. We assume that $B << B_1 < B_2$. This condition means that the frequency components in the signal are much larger than the bandwidth B of the signal. Such signals are usually called bandpass or narrowband signals. Now, if this signal is sampled at a rate $F_s \geq 2B$, but $F_s << B_1$, then all the frequency components contained in the signal will be aliases of frequencies in the range $0 < F < F_s/2$. Consequently, if we observe the frequency content of the signal in the fundamental range $0 < F < F_s/2$, we know precisely the frequency content of the analog signal since we know the frequency band $B_1 < F < B_2$ under consideration. Consequently, if the signal is a narrowband (bandpass) signal, we can reconstruct the original signal from the samples, provided that the signal is sampled at a rate $F_s > 2B$, where B is the bandwidth. This statement constitutes another form of the sampling theorem, which we call the *bandpass form* in order to distinguish it from the previous form of the sampling theorem, which applies in general to all types of signals. The latter is sometimes called the *baseband form*.

4.3 Quantization of Continuous-Amplitude Signals

As we have seen, a digital signal is a sequence of numbers (samples) in which each number is represented by a finite number of digits (finite precision).

The process of converting a discrete-time continuous-amplitude signal into a digital signal by expressing each sample value as a finite (instead of an infinite) number of digits is called *quantization*. The error introduced in representing the continuous-valued signal by a finite set of discrete value levels is called *quantization error* or *quantization noise*.

We denote the quantizer operation on the samples $x(n)$ as $Q[x(n)]$ and let $x_q(n)$ denote the sequence of quantized samples at the output of the quantizer. Hence

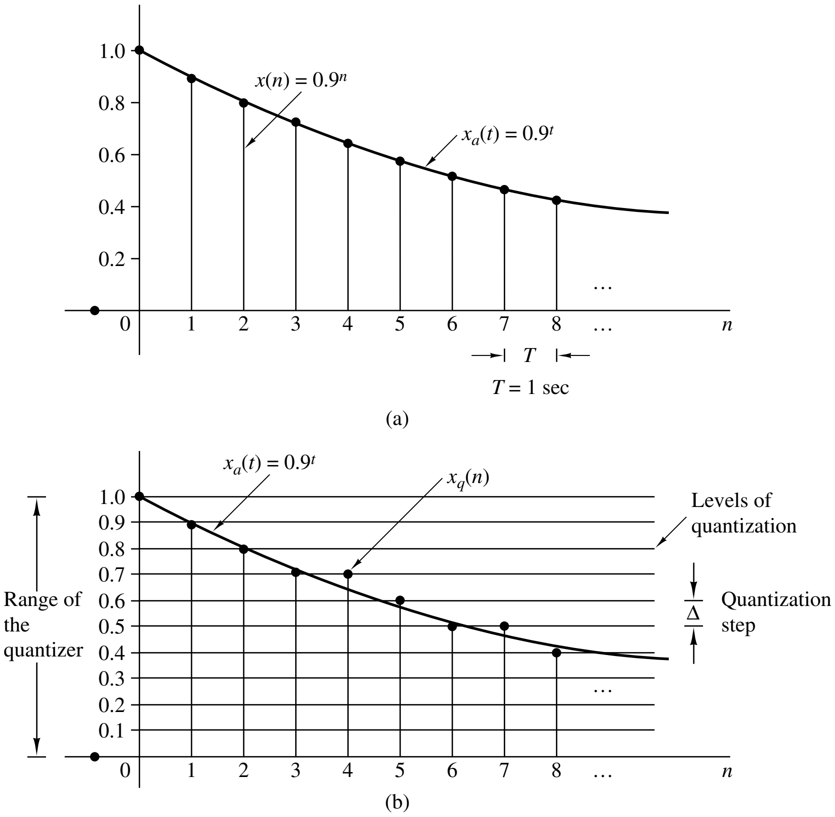
$$x_q(n) = Q[x(n)]$$

Then the quantization error is a sequence $e_q(n)$ defined as the difference between the quantized value and the actual sample value. Thus

$$e_q(n) = x_q(n) - x(n) \tag{4.25}$$

We illustrate the quantization process with an example. Let us consider the discrete-time signal

$$x(n) = \begin{cases} 0.9^n, & n \geq 0 \\ 0, & n < 0 \end{cases}$$



(a)

(b)

Figure 4.7 Illustration of quantization.

obtained by sampling the analog exponential signal $x_a(t) = 0.9^t$, $t \geq 0$ with a sampling frequency $F_s = 1$ Hz (see Fig. 4.7(a)). Observation of Table 2, which shows the values of the first 10 samples of $x(n)$, reveals that the description of the sample value $x(n)$ requires n significant digits. It is obvious that this signal cannot be processed by using a calculator or a digital computer since only the first few samples can be stored and manipulated. For example, most calculators process numbers with only eight significant digits.

However, let us assume that we want to use only one significant digit. To eliminate the excess digits, we can either simply discard them (*truncation*) or discard them by rounding the resulting number (*rounding*). The resulting quantized signals $x_q(n)$ are shown in Table 2. We discuss only quantization by rounding, although it is just as easy to treat truncation. The rounding process is graphically illustrated in Fig. 4.7(b). The values allowed in the digital signal are called the *quantization levels*, whereas the distance Δ between two successive quantization levels is called the *quantization step size* or *resolution*. The rounding quantizer assigns each sample of $x(n)$ to the nearest quantization level. In contrast, a quantizer that performs truncation would have assigned each sample of $x(n)$ to the quantization level below

TABLE 2 Numerical Illustration of Quantization with One Significant Digit Using Truncation or Rounding

n	$x(n)$ Discrete-time signal	$x_q(n)$ (Truncation)	$x_q(n)$ (Rounding)	$e_q(n) = x_q(n) - x(n)$ (Rounding)
0	1	1.0	1.0	0.0
1	0.9	0.9	0.9	0.0
2	0.81	0.8	0.8	−0.01
3	0.729	0.7	0.7	−0.029
4	0.6561	0.6	0.7	0.0439
5	0.59049	0.5	0.6	0.00951
6	0.531441	0.5	0.5	−0.031441
7	0.4782969	0.4	0.5	0.0217031
8	0.43046721	0.4	0.4	−0.03046721
9	0.387420489	0.3	0.4	0.012579511

it. The quantization error $e_q(n)$ in rounding is limited to the range of $-\Delta/2$ to $\Delta/2$, that is,

$$-\frac{\Delta}{2} \leq e_q(n) \leq \frac{\Delta}{2} \tag{4.26}$$

In other words, the instantaneous quantization error cannot exceed half of the quantization step (see Table 2).

If $x_{\min}$ and $x_{\max}$ represent the minimum and maximum values of $x(n)$ and L is the number of quantization levels, then

$$\Delta = \frac{x_{\max} - x_{\min}}{L - 1} \tag{4.27}$$

We define the *dynamic range* of the signal as $x_{\max} - x_{\min}$. In our example we have $x_{\max} = 1$, $x_{\min} = 0$, and $L = 11$, which leads to $\Delta = 0.1$. Note that if the dynamic range is fixed, increasing the number of quantization levels L results in a decrease of the quantization step size. Thus the quantization error decreases and the accuracy of the quantizer increases. In practice we can reduce the quantization error to an insignificant amount by choosing a sufficient number of quantization levels.

Theoretically, quantization of analog signals always results in a loss of information. This is a result of the ambiguity introduced by quantization. Indeed, quantization is an irreversible or noninvertible process (i.e., a many-to-one mapping) since all samples in a distance $\Delta/2$ about a certain quantization level are assigned the same value. This ambiguity makes the exact quantitative analysis of quantization extremely difficult.

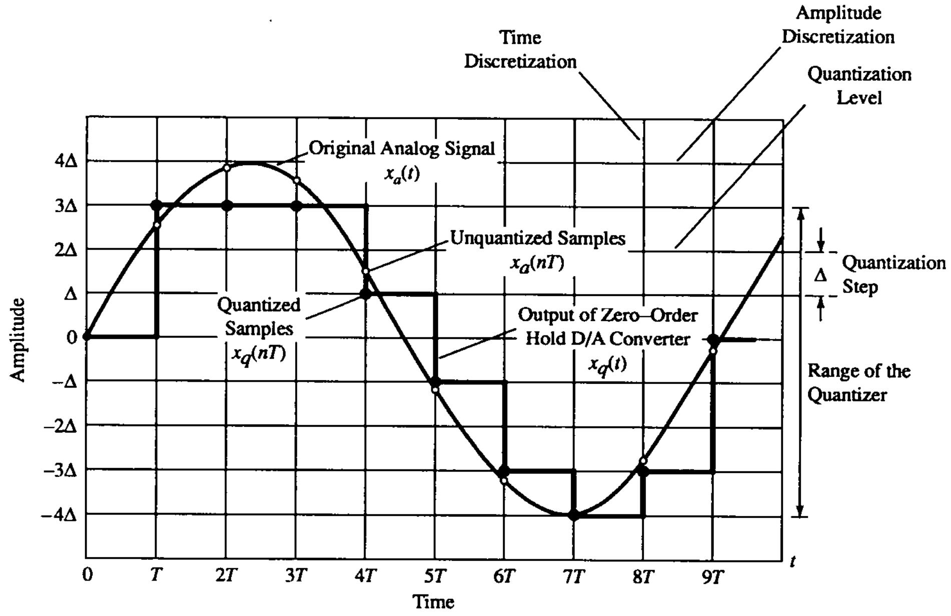


Figure 4.8 Sampling and quantization of a sinusoidal signal.

4.4 Quantization of Sinusoidal Signals

Figure 4.8 illustrates the sampling and quantization of an analog sinusoidal signal $x_a(t) = A\cos\Omega_0 t$ using a rectangular grid. Horizontal lines within the range of the quantizer indicate the allowed levels of quantization. Vertical lines indicate the sampling times. Thus, from the original analog signal $x_a(t)$ we obtain a discrete-time signal $x(n) = x_a(nT)$ by sampling and a discrete-time, discrete-amplitude signal $x_q(nT)$ after quantization. In practice, the staircase signal $x_q(t)$ can be obtained by using a zero-order hold. This analysis is useful because sinusoids are used as test signals in A/D converters.

If the sampling rate F_s satisfies the sampling theorem, quantization is the only error in the A/D conversion process.

Thus we can evaluate the quantization error by quantizing the analog signal $x_a(t)$ instead of the discrete-time signal $x(n) = x_a(nT)$. Inspection of Fig. 4.8 indicates that the signal $x_a(t)$ is almost linear between quantization levels (see Fig. 4.9). The

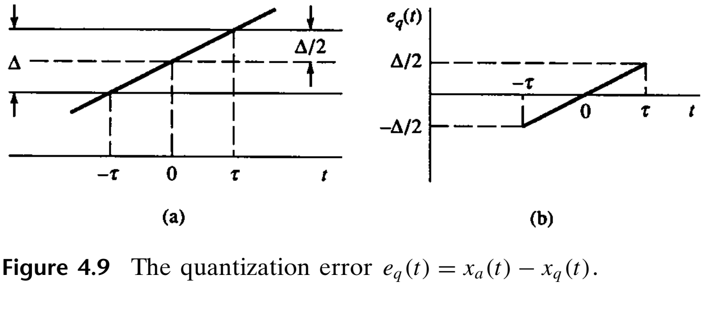


Figure 4.9 The quantization error $e_q(t) = x_a(t) - x_q(t)$.

corresponding quantization error $e_q(t) = x_a(t) - x_q(t)$ is shown in Fig. 4.9. In Fig. 4.9, τ denotes the time that $x_a(t)$ stays within the quantization levels. The mean-square error power P_q is

$$P_q = \frac{1}{2\tau}\int_{-\tau}^{\tau} e_q^2(t)\,dt = \frac{1}{\tau}\int_{0}^{\tau} e_q^2(t)\,dt \tag{4.28}$$

Since $e_q(t) = (\Delta/2\tau)t$, $-\tau \le t \le \tau$, we have

$$P_q = \frac{1}{\tau}\int_0^{\tau}\left(\frac{\Delta}{2\tau}\right)^2 t^2\,dt = \frac{\Delta^2}{12} \tag{4.29}$$

If the quantizer has b bits of accuracy and the quantizer covers the entire range $2A$, the quantization step is $\Delta = 2A/2^b$. Hence

$$P_q = \frac{A^2/3}{2^{2b}} \tag{4.30}$$

The average power of the signal $x_a(t)$ is

$$P_x = \frac{1}{T_p}\int_0^{T_p}(A\cos\Omega_0 t)^2\,dt = \frac{A^2}{2} \tag{4.31}$$

The quality of the output of the A/D converter is usually measured by the *signal-to-quantization noise ratio (SQNR)*, which provides the ratio of the signal power to the noise power:

$$\text{SQNR} = \frac{P_x}{P_q} = \frac{3}{2}\cdot 2^{2b}$$

Expressed in decibels (dB), the SQNR is

$$\text{SQNR(dB)} = 10\log_{10}\text{SQNR} = 1.76 + 6.02b \tag{4.32}$$

This implies that the SQNR increases approximately 6 dB for every bit added to the word length, that is, for each doubling of the quantization levels.

Although formula (4.32) was derived for sinusoidal signals, we shall a similar result holds for every signal whose dynamic range spans the range of the quantizer. This relationship is extremely important because it dictates the number of bits required by a specific application to assure a given signal-to-noise ratio. For example, most compact disc players use a sampling frequency of 44.1 kHz and 16-bit sample resolution, which implies a SQNR of more than 96 dB.

4.5 Coding of Quantized Samples

The coding process in an A/D converter assigns a unique binary number to each quantization level. If we have L levels we need at least L different binary numbers. With a word length of b bits we can create 2^b different binary numbers. Hence we have $2^b \ge L$, or equivalently, $b \ge \log_2 L$. Thus the number of bits required in the coder is the smallest integer greater than or equal to $\log_2 L$. In our example (Table 2) it can easily be seen that we need a coder with $b = 4$ bits. Commercially available A/D converters may be obtained with finite precision of $b = 16$ or less. Generally, the higher the sampling speed and the finer the quantization, the more expensive the device becomes.

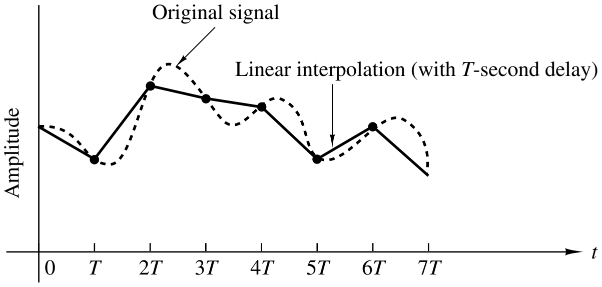


Figure 4.10
Linear point connector (with T-second delay).

4.6 Digital-to-Analog Conversion

To convert a digital signal into an analog signal we can use a digital-to-analog (D/A) converter. As stated previously, the task of a D/A converter is to interpolate between samples.

The sampling theorem specifies the optimum interpolation for a bandlimited signal. However, this type of interpolation is too complicated and, hence, impractical, as indicated previously. From a practical viewpoint, the simplest D/A converter is the zero-order hold shown in Fig. 4.2, which simply holds constant the value of one sample until the next one is received. Additional improvement can be obtained by using linear

interpolation as shown in Fig. 4.10 to connect successive samples with straight-line segments. Better interpolation can be achieved by using more sophisticated higher-order interpolation techniques.

In general, suboptimum interpolation techniques result in passing frequencies above the folding frequency. Such frequency components are undesirable and are usually removed by passing the output of the interpolator through a proper analog filter, which is called a *postfilter* or *smoothing filter*.

Thus D/A conversion usually involves a suboptimum interpolator followed by a postfilter.

4.7 Analysis of Digital Signals and Systems Versus Discrete-Time Signals and Systems

We have seen that a digital signal is defined as a function of an integer independent variable and its values are taken from a finite set of possible values. The usefulness of such signals is a consequence of the possibilities offered by digital computers. Computers operate on numbers, which are represented by a stringof 0's and 1's. The length of this string (*word length*) is fixed and finite and usually is 8, 12, 16, or 32 bits. The effects of finite word length in computations cause complications in the analysis of digital signal processing systems. To avoid these complications, we neglect the quantized nature of digital signals and systems in much of our analysis and consider them as discrete-time signals and systems.

Although not covered in this chapter, investigating the consequences of using a finite word length is an important topic, since many digital signal processing problems are solved with small computers or microprocessors that employ fixed-point arithmetic. Consequently, one must look carefully at the problem of finite-precision arithmetic and account for it in the design of software and hardware that performs the desired signal processing tasks.

5 Summary and References

In this introductory chapter we have attempted to provide the motivation for digital signal processing as an alternative to analog signal processing. We presented the basic elements of a digital signal processing system and defined the operations needed to convert an analog signal into a digital signal ready for processing. Of particular importance is the sampling theorem, which was introduced by Nyquist (1928) and later popularized in the classic paper by Shannon (1949). Sinusoidal signals were introduced primarily for the purpose of illustrating the aliasing phenomenon and for the subsequent development of the sampling theorem.

Quantization effects that are inherent in the A/D conversion of a signal were also introduced in this chapter. Signal quantization is best treated in statistical terms.

Finally, the topic of signal reconstruction, or D/A conversion, was described briefly.

There are numerous practical applications of digital signal processing. The book edited by Oppenheim (1978) treats applications to speech processing, image processing, radar signal processing, sonar signal processing, and geophysical signal processing.