UNIT II: PHYSICAL LAYER

Design issues of physical layer, Analog & Digital Data representation, encoding: Binary Stream, Manchester, NRZ, NRZ-I. Data transmission modes: Parallel, Serial, Multiplexing:FDM, WDM, TDM, switching: Circuit Switching and Packet Switching. Structure of a Switching.

Design Issues in Physical Layer

Physical layer co-ordinates the functions required to transmit a bit stream over a physical medium. It deals with the mechanical and electrical specifications of the interface and transmission medium.

The lowest layer of the OSI reference model is the physical layer. It is responsible for the actual physical connection between the devices. The physical layer contains information in the form of bits. It is responsible for transmitting individual bits from one node to the next.

Functions of physical layer:

1. **Data Rate –**
   This layer defines the rate of transmission which is the number of bits per second.
2. **Interface –**
   The physical layer defines the transmission interface between devices and transmission medium.
3. **Representation of Bits –**
   Data in this layer consist of stream of bits. The bits must be encoded into signals for transmission. It defines the type of encoding i.e. How 0's and 1's are changed to signals.
4. **Line configuration –**
   This layer connects devices with the medium either it can be point to point configuration or Multipoint configuration.
5. **Transmission Modes –**
   Physical layer defines the direction of transmission between two devices such as Simplex, Half duplex, Full duplex.
6. **Topologies –**
   Devices must be connected through any topologies either it can be Mesh, Star, Bus And Ring.

Design Issues in Physical Layer :

- The physical layer is basically concerned with transmitting raw bits over a communication channel.
- Mainly the design issues here deal with electrical, mechanical, timing interfaces, and the physical transmission medium, which lies below the physical layer.
- Design issue has to do with making sure that when 1 bit send from one side, it is received 1 bit by other side also not as a 0 bit.

CHAPTER 3

Data and Signals

One of the major functions of the physical layer is to move data in the form of electromagnetic signals across a transmission medium. Whether you are collecting numerical statistics from another computer, sending animated pictures from a design workstation, or causing a bell to ring at a distant control center, you are working with the transmission of **data** across network connections.

Generally, the data usable to a person or application are not in a form that can be transmitted over a network. For example, a photograph must first be changed to a form that transmission media can accept. Transmission media work by conducting energy along a physical path.

To be transmitted, data must be transformed to electromagnetic signals.

3.1 ANALOG AND DIGITAL

Both data and the signals that represent them can be either **analog or digital** in form.

Analog and Digital Data

Data can be analog or digital. The term **analog data** refers to information that is continuous; **digital data** refers to information that has discrete states. For example, an analog clock that has hour, minute, and second hands gives information in a continuous form; the movements of the hands are continuous. On the other hand, a digital clock that reports the hours and the minutes will change suddenly from 8:05 to 8:06.

Analog data, such as the sounds made by a human voice, take on continuous values. When someone speaks, an analog wave is created in the air. This can be captured by a microphone and converted to an analog signal or sampled and converted to a digital signal.

Digital data take on discrete values. For example, data are stored in computer memory in the form of 0s and 1s. They can be converted to a digital signal or modulated into an analog signal for transmission across a medium.

Data can be analog or digital. Analog data are continuous and take continuous values. Digital data have discrete states and take discrete values.

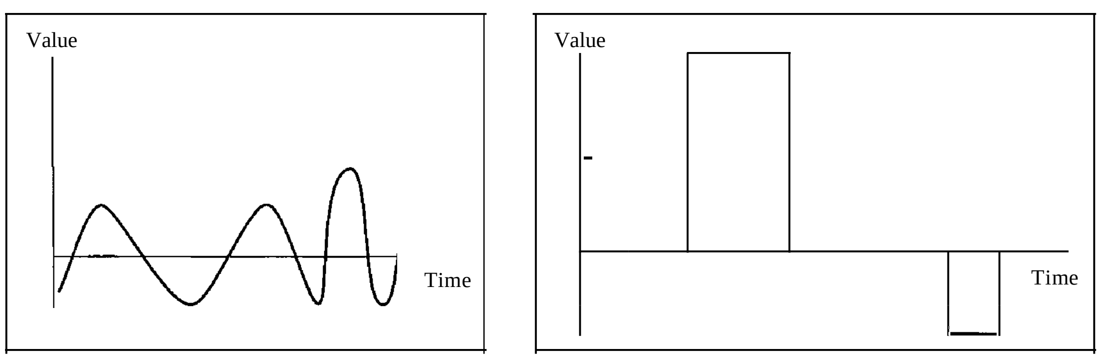
Analog and Digital Signals

Like the data they represent, signals can be either analog or digital. An analog signal has infinitely many levels of intensity over a period of time. As the wave moves from value *A* to value *B*, it passes through and includes an infinite number of values along its path. A digital signal, on the other hand, can have only a limited number of defined values. Although each value can be any number, it is often as simple as 1 and 0.

The simplest way to show signals is by plotting them on a pair of perpendicular axes. The vertical axis represents the value or strength of a signal. The horizontal axis represents time. Figure 3.1 illustrates an analog signal and a digital signal. The curve representing the analog signal passes through an infinite number of points. The vertical lines of the digital signal, however, demonstrate the sudden jump that the signal makes from value to value.

Signals can be analog or digital. Analog signals can have an infinite number of values in a range; digital signals can have only a limited number of values.

Figure 3.1 *Comparison of analog and digital signals*



a. Analog signal b. Digital signal

Periodic and Nonperiodic Signals

Both analog and digital signals can take one of two forms: *periodic* or *nonperiodic* (sometimes refer to as *aperiodic*, because the prefix *a* in Greek means "non").

A periodic signal completes a pattern within a measurable time frame, called a period, and repeats that pattern over subsequent identical periods. The completion of one full pattern is called a cycle. A nonperiodic signal changes without exhibiting a pattern or cycle that repeats over time.

Both analog and digital signals can be periodic or nonperiodic. In data communications, we commonly use periodic analog signals (because they need less bandwidth, as we will see in Chapter 5) and nonperiodic digital signals (because they can represent variation in data, as we will see in Chapter 6).

In data communications, we commonly use periodic analog signals and nonperiodic digital signals.

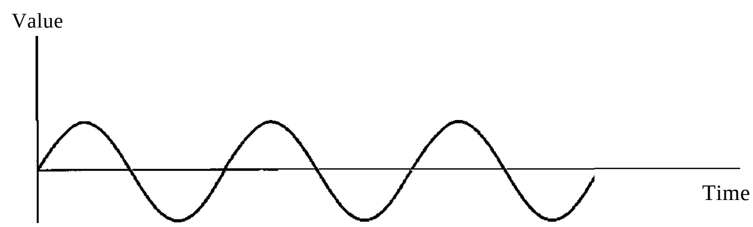
3.2 **PERIODIC** ANALOG SIGNALS

Periodic analog signals can be classified as simple or composite. A simple periodic analog signal, a sine wave, cannot be decomposed into simpler signals. A composite periodic analog signal is composed of multiple sine waves.

Sine Wave

The sine wave is the most fundamental form of a periodic analog signal. When we visualize it as a simple oscillating curve, its change over the course of a cycle is smooth and consistent, a continuous, rolling flow. Figure 3.2 shows a sine wave. Each cycle consists of a single arc above the time axis followed by a single arc below it.

Figure 3.2 *A sine wave*



We discuss a mathematical approach to sine waves in Appendix C.

A sine wave can be represented by three parameters: the *peak amplitude,* the *frequency,* and the *phase.* These three parameters fully describe a sine wave.

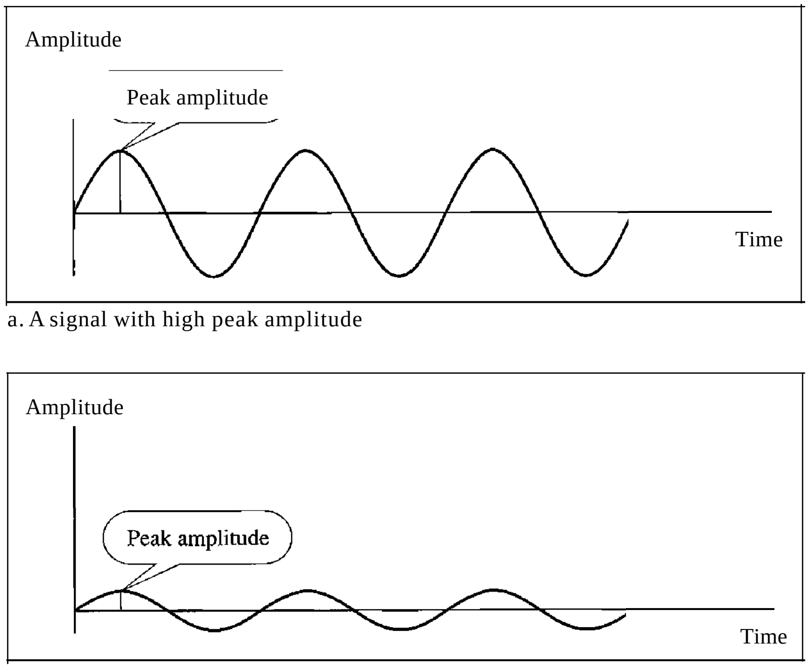
Peak Amplitude

The peak amplitude of a signal is the absolute value of its highest intensity, proportional to the energy it carries. For electric signals, peak amplitude is normally measured in *volts.* Figure 3.3 shows two signals and their peak amplitudes.

Example 3.1

The power in your house can be represented by a sine wave with a peak amplitude of 155 to 170 **V.** However, it is common knowledge that the voltage of the power in U.S. homes is 110 to 120 **V.**

Figure 3.3 *Two signals with the same phase and frequency, but different amplitudes*



a. A signal with high peak amplitude

b. A signal with low peak amplitude

This discrepancy is due to the fact that these are root mean square (rms) values. The signal is squared and then the average amplitude is calculated. The peak value is equal to 2^{112} x rms value.

Example 3.2

The voltage of battery is a constant; this constant value can be considered a sine wave, as we will see later. For example, the peak value of an AA battery is normally 1.5 V.

Period and Frequency

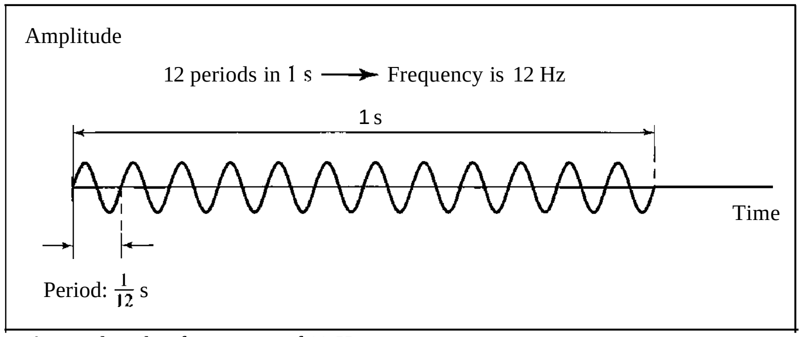
Period refers to the amount of time, in seconds, a signal needs to complete 1 cycle. Frequency refers to the number of periods in I s. Note that period and frequency are just one characteristic defined in two ways. Period is the inverse of frequency, and frequency is the inverse of period, as the following formulas show.

$$f = \frac{1}{T} \quad \text{and} \quad T = \frac{1}{f}$$

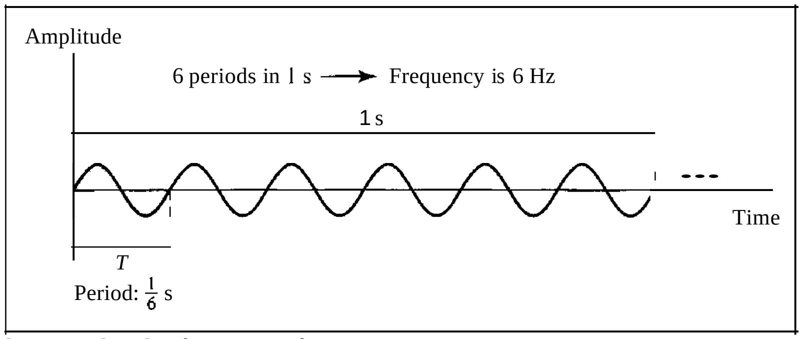
Frequency and period are the inverse of each other.

Figure 3.4 shows two signals and their frequencies.

Figure 3.4 *Two signals with the same amplitude and phase, but different frequencies*



a. A signal with a frequency of 12 Hz



b. A signal with a frequency of 6 Hz

Period is formally expressed in seconds. Frequency is formally expressed in Hertz (Hz), which is cycle per second. Units of period and frequency are shown in Table 3.1.

Table 3.1 *Units of period and frequency*

Unit	*Equivalent*	*Unit*	*Equivalent*
Seconds (s)	1 s	Hertz (Hz)	1 Hz
Milliseconds (ms)	10^{-3} s	Kilohertz (kHz)	10^{3} Hz
Microseconds (μs)	10^{-6} s	Megahertz (MHz)	10^{6} Hz
Nanoseconds (ns)	10^{-9} s	Gigahertz (GHz)	10^{9} Hz
Picoseconds (ps)	10^{-12} s	Terahertz (THz)	10^{12} Hz

Example 3.3

The power we use at home has a frequency of 60 Hz (50 Hz in Europe). The period of this sine wave can be determined as follows:

$$T = \frac{1}{f} = \frac{1}{60} = 0.0166 \text{ s} = 0.0166 \times 10^{3} \text{ ms} = 16.6 \text{ ms}$$

This means that the period of the power for our lights at home is 0.0116 s, or 16.6 ms. Our eyes are not sensitive enough to distinguish these rapid changes in amplitude.

Example 3.4

Express a period of 100 ms in microseconds.

Solution

From Table 3.1 we find the equivalents of 1 ms (1 ms is 10^{-3} s) and 1 s (1 s is 10^6 μs). We make the following substitutions:

$$100 \text{ ms} = 100 \times 10^{-3} \text{ s} = 100 \times 10^{-3} \times 10^{6}\ \mu\text{s} = 10^{2} \times 10^{-3} \times 10^{6}\ \mu\text{s} = 10^{5}\ \mu\text{s}$$

Example 3.5

The period of a signal is 100 ms. What is its frequency in kilohertz?

Solution

First we change 100 ms to seconds, and then we calculate the frequency from the period (1 Hz = 10^{-3} kHz).

$$100 \text{ ms} = 100 \times 10^{-3} \text{ s} = 10^{-1} \text{ s}$$

$$f = \frac{1}{T} = \frac{1}{10^{-1}} \text{ Hz} = 10 \text{ Hz} = 10 \times 10^{-3} \text{ kHz} = 10^{-2} \text{ kHz}$$

More About Frequency

We already know that frequency is the relationship of a signal to time and that the frequency of a wave is the number of cycles it completes in 1 s. But another way to look at frequency is as a measurement of the rate of change. Electromagnetic signals are oscillating waveforms; that is, they fluctuate continuously and predictably above and below a mean energy level. A 40-Hz signal has one-half the frequency of an 80-Hz signal; it completes 1 cycle in twice the time of the 80-Hz signal, so each cycle also takes twice as long to change from its lowest to its highest voltage levels. Frequency, therefore, though described in cycles per second (hertz), is a general measurement of the rate of change of a signal with respect to time.

Frequency is the rate of change with respect to time. Change in a short span of time means high frequency. Change over a long span of time means low frequency.

If the value of a signal changes over a very short span of time, its frequency is high. If it changes over a long span of time, its frequency is low.

Two Extremes

What if a signal does not change at all? What if it maintains a constant voltage level for the entire time it is active? In such a case, its frequency is zero. Conceptually, this idea is a simple one. If a signal does not change at all, it never completes a cycle, so its frequency is 0 Hz.

But what if a signal changes instantaneously? What if it jumps from one level to another in no time? Then its frequency is infinite. In other words, when a signal changes instantaneously, its period is zero; since frequency is the inverse of period, in this case, the frequency is 1/0, or infinite (unbounded).

If a signal does not change at all, its frequency is zero.
If a signal changes instantaneously, its frequency is infinite.

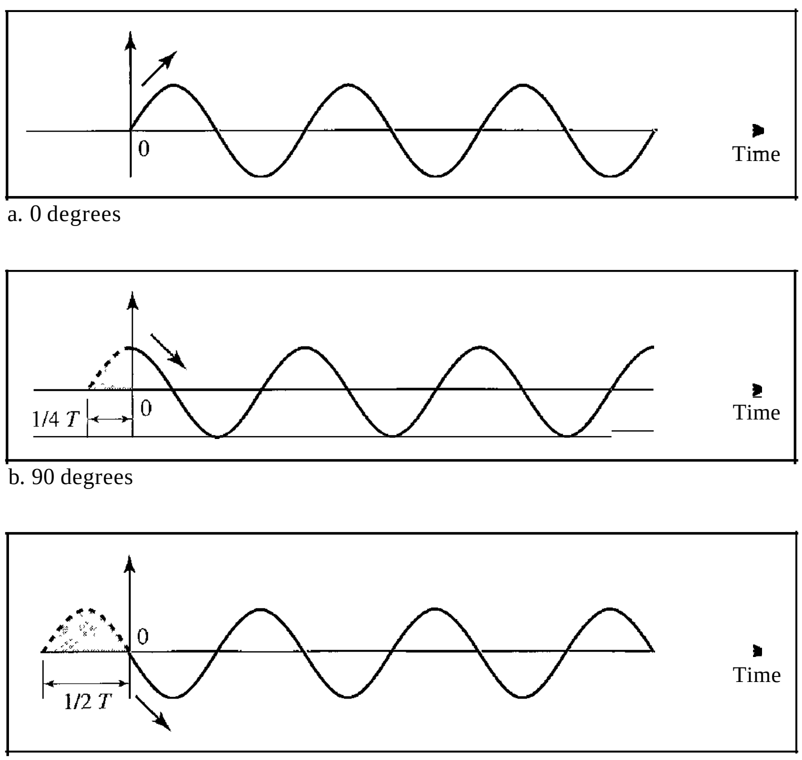
Phase

The term phase describes the position of the waveform relative to time 0. If we think of the wave as something that can be shifted backward or forward along the time axis, phase describes the amount of that shift. It indicates the status of the first cycle.

Phase describes the position of the waveform relative to time 0.

Phase is measured in degrees or radians [360° is *2n* rad; 1° is *2n/360* rad, and 1 rad is *360/(2n)]*. A phase shift of 360° corresponds to a shift of a complete period; a phase shift of 180° corresponds to a shift of one-half of a period; and a phase shift of 90° corresponds to a shift of one-quarter of a period (see Figure 3.5).

Figure 3.5 *Three sine waves with the same amplitude andfrequency, but different phases*



a. 0 degrees

b. 90 degrees

c. 180 degrees

Looking at Figure 3.5, we can say that

1. A sine wave with a phase of 0° starts at time 0 with a zero amplitude. The amplitude is increasing.
2. A sine wave with a phase of 90° starts at time 0 with a peak amplitude. The amplitude is decreasing.

3. A sine wave with a phase of 180° starts at time 0 with a zero amplitude. The amplitude is decreasing.

Another way to look at the phase is in terms of shift or offset. We can say that

1. A sine wave with a phase of 0° is not shifted.
2. A sine wave with a phase of 90° is shifted to the left by $\frac{1}{4}$ cycle. However, note that the signal does not really exist before time 0.
3. A sine wave with a phase of 180° is shifted to the left by $\frac{1}{2}$ cycle. However, note that the signal does not really exist before time 0.

Example 3.6

A sine wave is offset $\frac{1}{6}$ cycle with respect to time 0. What is its phase in degrees and radians?

Solution

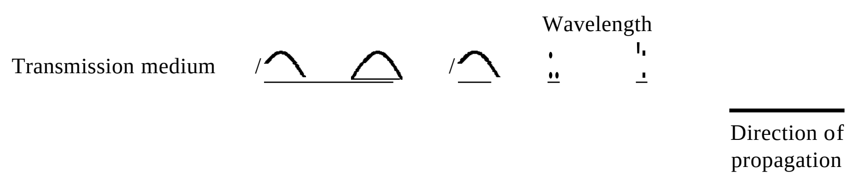
We know that 1 complete cycle is 360°. Therefore, $\frac{1}{6}$ cycle is

$$\frac{1}{6} \times 360 = 60° = 60 \times \frac{2\pi}{360} \text{ rad} = \frac{\pi}{3} \text{ rad} = 1.046 \text{ rad}$$

Wavelength

Wavelength is another characteristic of a signal traveling through a transmission medium. Wavelength binds the period or the frequency of a simple sine wave to the propagation speed of the medium (see Figure 3.6).

Figure 3.6 *Wavelength and period*



While the frequency of a signal is independent of the medium, the wavelength depends on both the frequency and the medium. Wavelength is a property of any type of signal. In data communications, we often use wavelength to describe the transmission of light in an optical fiber. The wavelength is the distance a simple signal can travel in one period.

Wavelength can be calculated if one is given the propagation speed (the speed of light) and the period of the signal. However, since period and frequency are related to each other, if we represent wavelength by λ, propagation speed by c (speed of light), and frequency by f, we get

$$\text{Wavelength} = \text{propagation speed} \times \text{period} = \frac{\text{propagation speed}}{\text{frequency}}$$

The propagation speed of electromagnetic signals depends on the medium and on the frequency of the signal. For example, in a vacuum, light is propagated with a speed of 3 x 10^8 mls. That speed is lower in air and even lower in cable.

The wavelength is normally measured in micrometers (microns) instead of meters. For example, the wavelength of red light (frequency $= 4 \times 10^{14}$) in air is

$$\lambda = \frac{c}{f} = \frac{3 \times 10^8}{4 \times 10^{14}} = 0.75 \times 10^{-6}\ \mathrm{m} = 0.75\ \mu\mathrm{m}$$

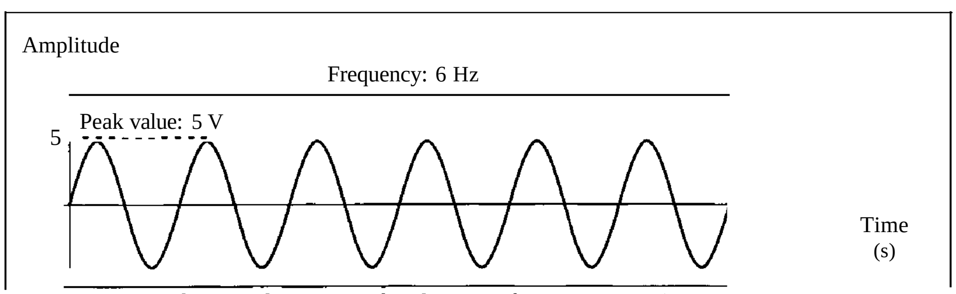
In a coaxial or fiber-optic cable, however, the wavelength is shorter (0.5 μm) because the propagation speed in the cable is decreased.

Time and Frequency Domains

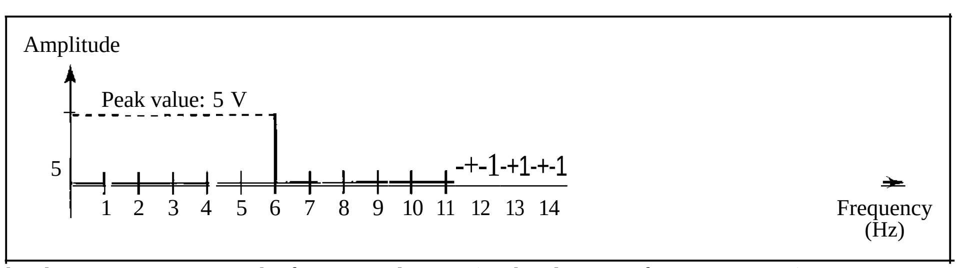
A sine wave is comprehensively defined by its amplitude, frequency, and phase. We have been showing a sine wave by using what is called a time-domain plot. The time-domain plot shows changes in signal amplitude with respect to time (it is an amplitude-versus-time plot). Phase is not explicitly shown on a time-domain plot.

To show the relationship between amplitude and frequency, we can use what is called a frequency-domain plot. A frequency-domain plot is concerned with only the peak value and the frequency. Changes of amplitude during one period are not shown. Figure 3.7 shows a signal in both the time and frequency domains.

Figure 3.7 *The time-domain andfrequency-domain plots ofa sine wave*



a. A sine wave in the time domain (**peak** value: 5 V, frequency: 6 Hz)



b. The same sine wave in the frequency domain (peak value: 5 V, frequency: 6 Hz)

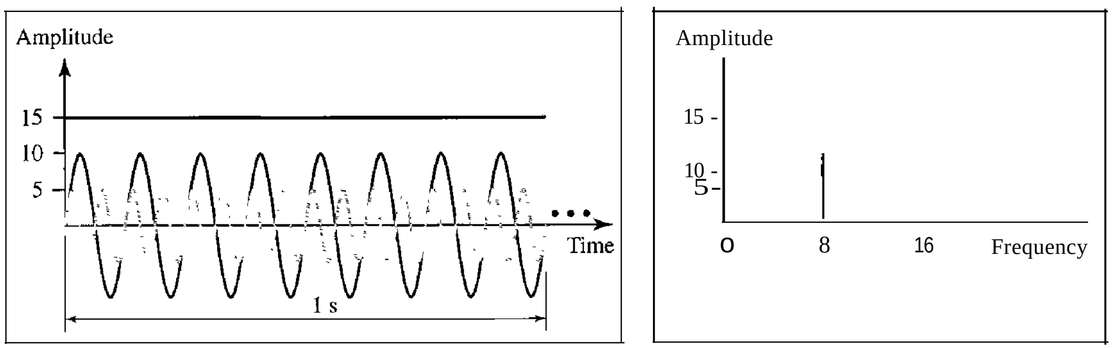
It is obvious that the frequency domain is easy to plot and conveys the information that one can find in a time domain plot. The advantage of the frequency domain is that we can immediately see the values of the frequency and peak amplitude. A complete sine wave is represented by one spike. The position of the spike shows the frequency; its height shows the peak amplitude.

A complete sine wave in the time domain can be represented by one single spike in the frequency domain.

Example 3.7

The frequency domain is more compact and useful when we are dealing with more than one sine wave. For example, Figure 3.8 shows three sine waves, each with different amplitude and frequency. All can be represented by three spikes in the frequency domain.

Figure 3.8 *The time domain and frequency domain of three sine waves*



a. Time-domain representation of three sine waves with frequencies 0, 8, and 16

b. Frequency-domain representation of the same three signals

Composite Signals

So far, we have focused on simple sine waves. Simple sine waves have many applications in daily life. We can send a single sine wave to carry electric energy from one place to another. For example, the power company sends a single sine wave with a frequency of 60 Hz to distribute electric energy to houses and businesses. As another example, we can use a single sine wave to send an alarm to a security center when a burglar opens a door or window in the house. In the first case, the sine wave is carrying energy; in the second, the sine wave is a signal of danger.

If we had only one single sine wave to convey a conversation over the phone, it would make no sense and carry no information. We would just hear a buzz. As we will see in Chapters 4 and 5, we need to send a composite signal to communicate data. A composite signal is made of many simple sine waves.

A **single-frequency** sine wave is not useful in data communications; we need to send a composite signal, a signal made of many simple sine waves.

In the early 1900s, the French mathematician Jean-Baptiste Fourier showed that any composite signal is actually a combination of simple sine waves with different frequencies, amplitudes, and phases. Fourier analysis is discussed in Appendix C; for our purposes, we just present the concept.

According to Fourier analysis, any composite signal is a combination of simple sine waves with different frequencies, amplitudes, and phases. Fourier analysis is discussed in Appendix C.

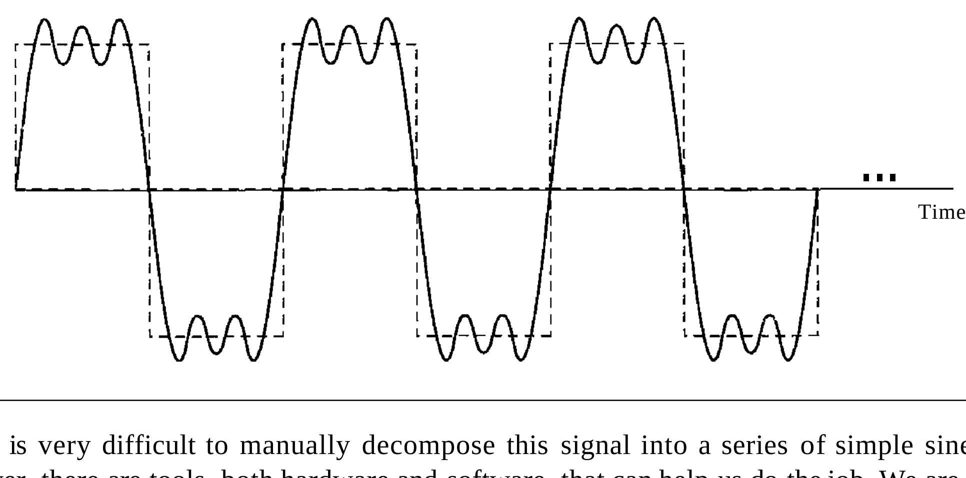
A composite signal can be periodic or nonperiodic. A periodic composite signal can be decomposed into a series of simple sine waves with discrete frequencies-frequencies that have integer values (1, 2, 3, and so on). A nonperiodic composite signal can be decomposed into a combination of an infinite number of simple sine waves with continuous frequencies, frequencies that have real values.

If the composite signal is periodic, the decomposition gives a series of signals with discrete frequencies; if the composite signal is nonperiodic, the decomposition gives a combination of sine waves with continuous frequencies.

Example 3.8

Figure 3.9 shows a periodic composite signal with frequency f This type of signal is not typical of those found in data communications.We can consider it to be three alarm systems, each with a different frequency. The analysis of this signal can give us a good understanding of how to decompose signals.

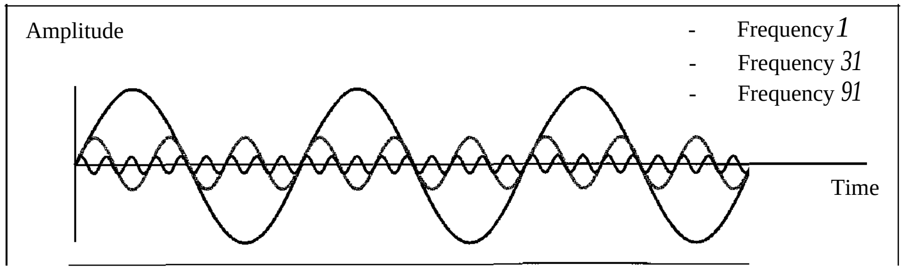
Figure 3.9 *A composite periodic signal*



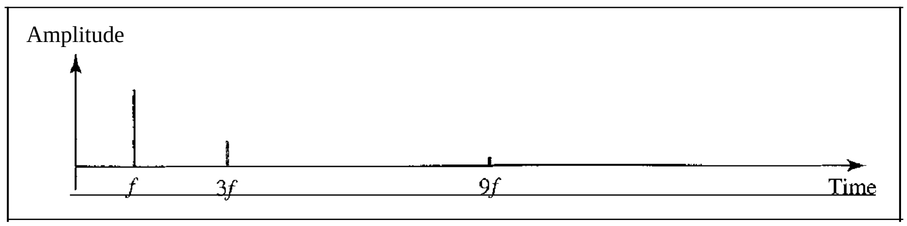
It is very difficult to manually decompose this signal into a series of simple sine waves. However, there are tools, both hardware and software, that can help us do the job. We are not concerned about how it is done; we are only interested in the result. Figure 3.10 shows the result of decomposing the above signal in both the time and frequency domains.

The amplitude of the sine wave with frequency f is almost the same as the peak amplitude of the composite signal. The amplitude of the sine wave with frequency $3f$ is one-third of that of the first, and the amplitude of the sine wave with frequency $9f$ is one-ninth of the first. The frequency

Figure 3.10 *Decomposition of a composite periodic signal in the time and frequency domains*



a. Time-domain decomposition of a composite signal



b. Frequency-domain decomposition of the composite signal

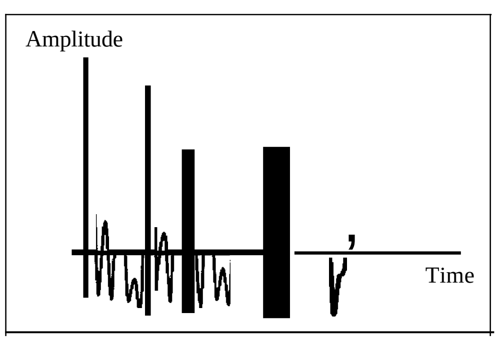
of the sine wave with frequency f is the same as the frequency of the composite signal; it is called the fundamental frequency, or first harmonic. The sine wave with frequency $3f$ has a frequency of 3 times the fundamental frequency; it is called the third harmonic. The third sine wave with frequency $9f$ has a frequency of 9 times the fundamental frequency; it is called the ninth harmonic.

Note that the frequency decomposition of the signal is discrete; it has frequencies f, $3f$, and $9f$ Because f is an integral number, $3f$ and $9f$ are also integral numbers. There are no frequencies such as $1.2f$ or $2.6f$ The frequency domain of a periodic composite signal is always made of discrete spikes.

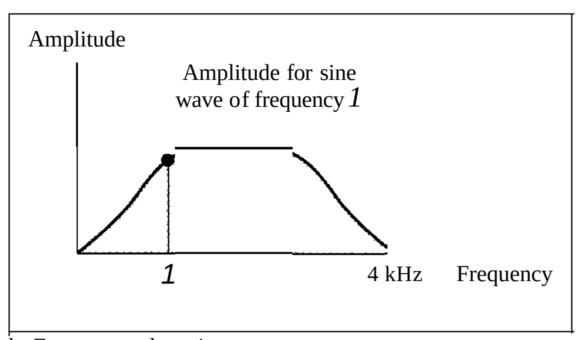
Example 3.9

Figure 3.11 shows a nonperiodic composite signal. It can be the signal created by a microphone or a telephone set when a word or two is pronounced. In this case, the composite signal cannot be periodic, because that implies that we are repeating the same word or words with exactly the same tone.

Figure 3.11 *The time and frequency domains of a nonperiodic signal*



a. Time domain



b. Frequency domain

In a time-domain representation of this composite signal, there are an infinite number of simple sine frequencies. Although the number of frequencies in a human voice is infinite, the range is limited. A normal human being can create a continuous range of frequencies between 0 and 4 kHz.

Note that the frequency decomposition of the signal yields a continuous curve. There are an infinite number of frequencies between 0.0 and 4000.0 (real values). To find the amplitude related to frequency f, we draw a vertical line at f to intersect the envelope curve. The height of the vertical line is the amplitude of the corresponding frequency.

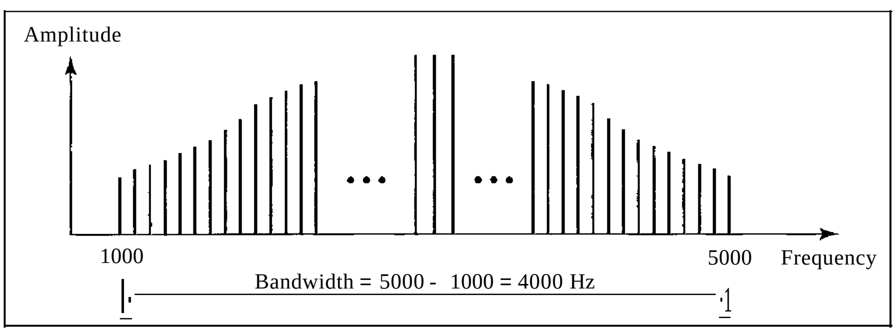
Bandwidth

The range of frequencies contained in a composite signal is its bandwidth. The bandwidth is normally a difference between two numbers. For example, if a composite signal contains frequencies between 1000 and 5000, its bandwidth is 5000 - 1000, or 4000.

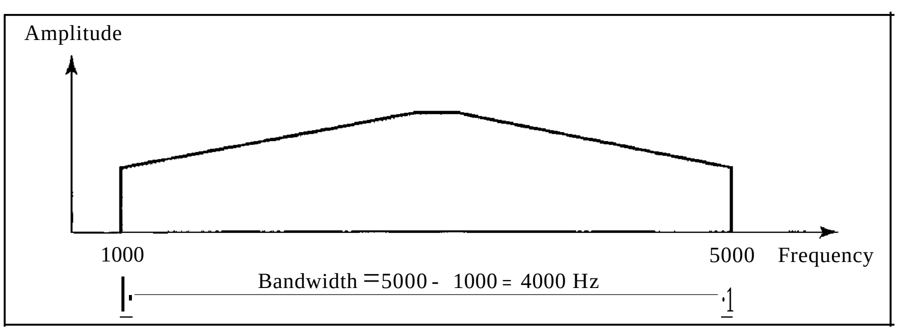
The bandwidth of a composite signal is the difference between the highest and the lowest frequencies contained in that signal.

Figure 3.12 shows the concept of bandwidth. The figure depicts two composite signals, one periodic and the other nonperiodic. The bandwidth of the periodic signal contains all integer frequencies between 1000 and 5000 (1000, 1001, 1002, ...). The bandwidth of the nonperiodic signals has the same range, but the frequencies are continuous.

Figure 3.12 *The bandwidth of periodic and nonperiodic composite signals*



a. Bandwidth of a periodic signal



b. Bandwidth of a nonperiodic signal

Example 3.10

If a periodic signal is decomposed into five sine waves with frequencies of 100, 300, 500, 700, and 900 Hz, what is its bandwidth? Draw the spectrum, assuming all components have a maximum amplitude of 10 V.

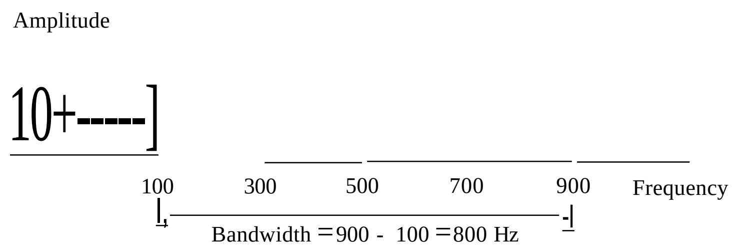
Solution

Let f_h be the highest frequency, f_l the lowest frequency, and B the bandwidth. Then

$$B = f_h - f_l = 900 - 100 = 800 \text{ Hz}$$

The spectrum has only five spikes, at 100, 300, 500, 700, and 900 Hz (see Figure 3.13).

Figure 3.13 *The bandwidth for Example 3.10*



Example 3.11

A periodic signal has a bandwidth of 20 Hz. The highest frequency is 60 Hz. What is the lowest frequency? Draw the spectrum if the signal contains all frequencies of the same amplitude.

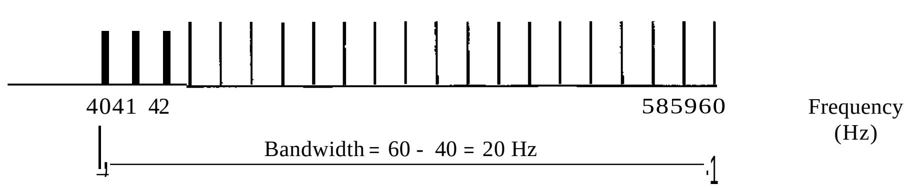
Solution

Let f_h be the highest frequency, f_l the lowest frequency, and B the bandwidth. Then

$$B = f_h - f_l \Rightarrow 20 = 60 - f_l \Rightarrow f_l = 60 - 20 = 40 \text{ Hz}$$

The spectrum contains all integer frequencies. We show this by a series of spikes (see Figure 3.14).

Figure 3.14 *The bandwidth for Example 3.11*



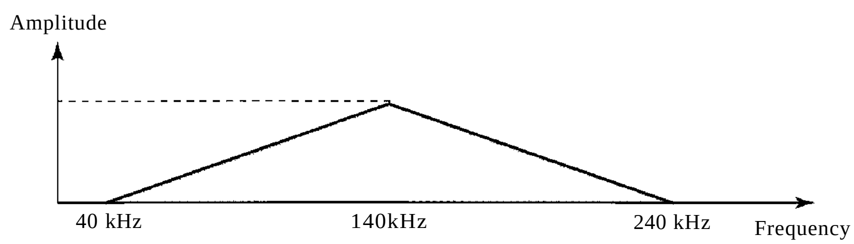
Example 3.12

A nonperiodic composite signal has a bandwidth of 200 kHz, with a middle frequency of 140 kHz and peak amplitude of 20 V. The two extreme frequencies have an amplitude of 0. Draw the frequency domain of the signal.

Solution

The lowest frequency must be at 40 kHz and the highest at 240 kHz. Figure 3.15 shows the frequency domain and the bandwidth.

Figure 3.15 *The bandwidth for Example 3.12*



Example 3. J3

An example of a nonperiodic composite signal is the signal propagated by an AM radio station, In the United States, each AM radio station is assigned a lO-kHz bandwidth. The total bandwidth dedicated to AM radio ranges from 530 to 1700 kHz. We will show the rationale behind this lO-kHz bandwidth in Chapter 5.

Example 3. J4

Another example of a nonperiodic composite signal is the signal propagated by an FM radio station. In the United States, each FM radio station is assigned a 200-kHz bandwidth. The total bandwidth dedicated to FM radio ranges from 88 to 108 MHz. We will show the rationale behind this 200-kHz bandwidth in Chapter 5.

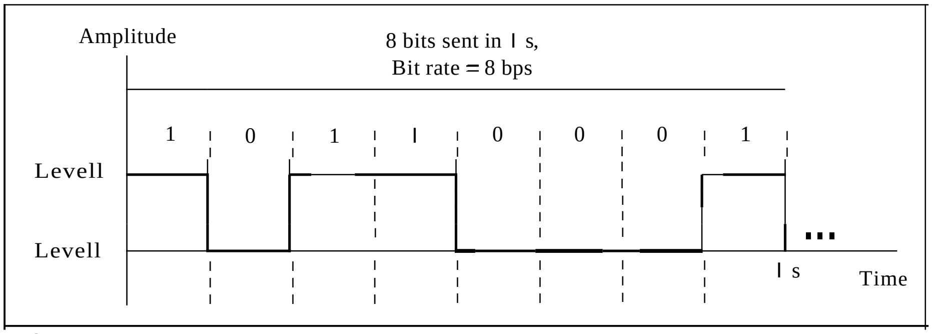
Example 3./5

Another example of a nonperiodic composite signal is the signal received by an old-fashioned analog black-and-white TV. A TV screen is made up of pixels (picture elements) with each pixel being either white or black. The screen is scanned 30 times per second. (Scanning is actually 60 times per second, but odd lines are scanned in one round and even lines in the next and then interleaved.) If we assume a resolution of 525 x 700 (525 vertical lines and 700 horizontal lines), which is a ratio of 3:4, we have 367,500 pixels per screen. If we scan the screen 30 times per second, this is 367,500 x 30 = 11,025,000 pixels per second. The worst-case scenario is alternating black and white pixels. In this case, we need to represent one color by the minimum amplitude and the other color by the maximum amplitude. We can send 2 pixels per cycle. Therefore, we need 11,025,000/2 =5,512,500 cycles per second, or Hz. The bandwidth needed is 5.5124 MHz. This worst-case scenario has such a low probability of occurrence that the assumption is that we need only 70 percent of this bandwidth, which is 3.85 MHz. Since audio and synchronization signals are also needed, a 4-MHz bandwidth has been set aside for each black and white TV channel. An analog color TV channel has a 6-MHz bandwidth.

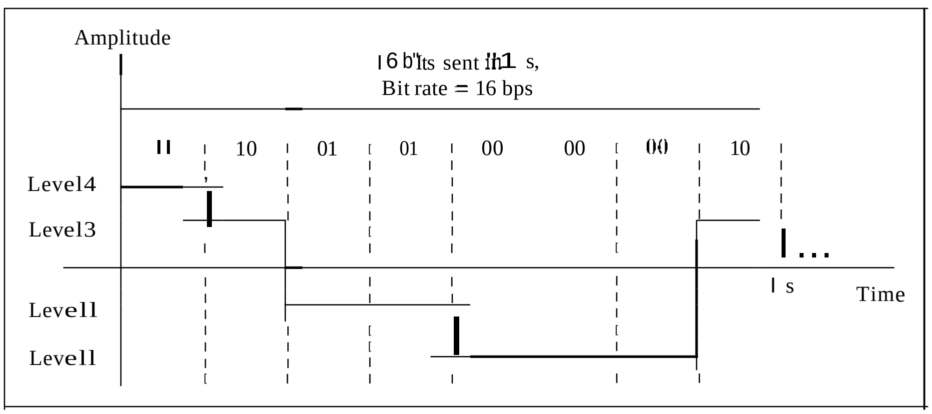
3.3 DIGITAL SIGNALS

In addition to being represented by an analog signal, information can also be represented by a digital signal. For example, a I can be encoded as a positive voltage and a 0 as zero voltage. A digital signal can have more than two levels. In this case, we can send more than 1 bit for each level. Figure 3.16 shows two signals, one with two levels and the other with four.

Figure 3.16 *Two digital signals: one with two signal levels and the other with four signal levels*



a. A digital signal with two levels



b. A digital signal with four levels

We send 1 bit per level in part a of the figure and 2 bits per level in part b of the figure. In general, if a signal has L levels, each level needs $\log_2 L$ bits.

Appendix C reviews information about exponential and logarithmic functions.

Example 3.16

A digital signal has eight levels. How many bits are needed per level? We calculate the number of bits from the formula

$$\text{Number of bits per level} = \log_2 8 = 3$$

Each signal level is represented by 3 bits.

Example 3.17

A digital signal has nine levels. How many bits are needed per level? We calculate the number of bits by using the formula. Each signal level is represented by 3.17 bits. However, this answer is not realistic. The number of bits sent per level needs to be an integer as well as a power of 2. For this example, 4 bits can represent one level.

Bit Rate

Most digital signals are nonperiodic, and thus period and frequency are not appropriate characteristics. Another *term-bit rate* (instead *ofjrequency)-is* used to describe digital signals. The bit rate is the number of bits sent in Is, expressed in bits per second (bps). Figure 3.16 shows the bit rate for two signals.

Example 3.18

Assume we need to download text documents at the rate of 100 pages per minute. What is the required bit rate of the channel?

Solution

A page is an average of 24 lines with 80 characters in each line. If we assume that one character requires 8 bits, the bit rate is

$$100 \times 24 \times 80 \times 8 = 1{,}636{,}000 \text{ bps} = 1.636 \text{ Mbps}$$

Example 3.19

A digitized voice channel, as we will see in Chapter 4, is made by digitizing a 4-kHz bandwidth analog voice signal. We need to sample the signal at twice the highest frequency (two samples per hertz). We assume that each sample requires 8 bits. What is the required bit rate?

Solution

The bit rate can be calculated as

$$2 \times 4000 \times 8 = 64{,}000 \text{ bps} = 64 \text{ kbps}$$

Example 3.20

What is the bit rate for high-definition TV (HDTV)?

Solution

HDTV uses digital signals to broadcast high quality video signals. The HDTV Screen is normally a ratio of 16 : 9 (in contrast to 4 : 3 for regular TV), which means the screen is wider. There are 1920 by 1080 pixels per screen, and the screen is renewed 30 times per second. Twenty-four bits represents one color pixel. We can calculate the bit rate as

$$1920 \times 1080 \times 30 \times 24 = 1{,}492{,}992{,}000 \text{ or } 1.5 \text{ Gbps}$$

The TV stations reduce this rate to 20 to 40 Mbps through compression.

Bit Length

We discussed the concept of the wavelength for an analog signal: the distance one cycle occupies on the transmission medium. We can define something similar for a digital signal: the bit length. The bit length is the distance one bit occupies on the transmission medium.

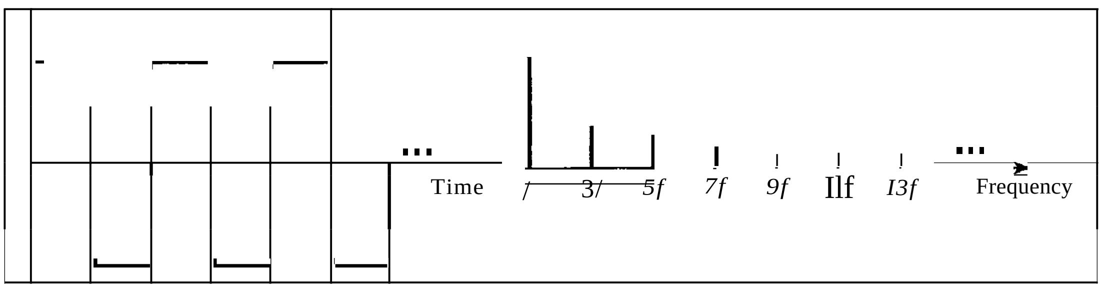
$$\text{Bit length} = \text{propagation speed} \times \text{bit duration}$$

Digital Signal as a Composite Analog Signal

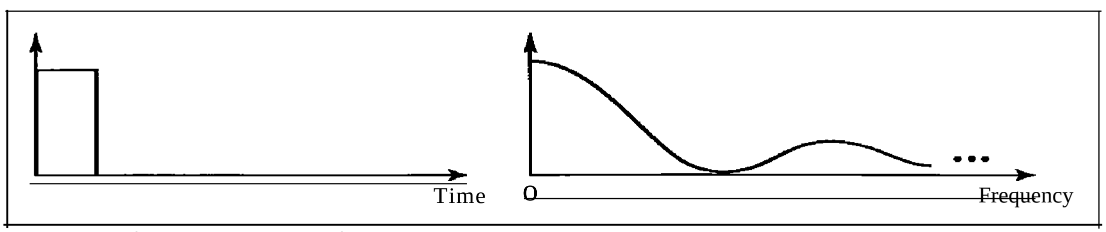
Based on Fourier analysis, a digital signal is a composite analog signal. The bandwidth is infinite, as you may have guessed. We can intuitively corne up with this concept when we consider a digital signal. A digital signal, in the time domain, comprises connected vertical and horizontal line segments. A vertical line in the time domain means a frequency of infinity (sudden change in time); a horizontal line in the time domain means a frequency of zero (no change in time). Going from a frequency of zero to a frequency of infinity (and vice versa) implies all frequencies in between are part of the domain.

Fourier analysis can be used to decompose a digital signal. If the digital signal is periodic, which is rare in data communications, the decomposed signal has a frequency-domain representation with an infinite bandwidth and discrete frequencies. If the digital signal is nonperiodic, the decomposed signal still has an infinite bandwidth, but the frequencies are continuous. Figure 3.17 shows a periodic and a nonperiodic digital signal and their bandwidths.

Figure 3.17 *The time andfrequency domains ofperiodic and nonperiodic digital signals*



a. Time and frequency domains of **periodic** digital signal



b. Time and frequency domains of nonperiodic digital signal

Note that both bandwidths are infinite, but the periodic signal has discrete frequencies while the nonperiodic signal has continuous frequencies.

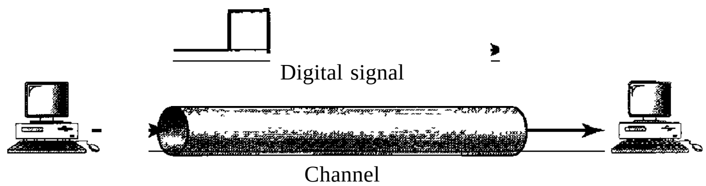
Transmission of Digital Signals

The previous discussion asserts that a digital signal, periodic or nonperiodic, is a composite analog signal with frequencies between zero and infinity. For the remainder of the discussion, let us consider the case of a nonperiodic digital signal, similar to the ones we encounter in data communications. The fundamental question is, How can we send a digital signal from point *A* to point *B*? We can transmit a digital signal by using one of two different approaches: baseband transmission or broadband transmission (using modulation).

Baseband Transmission

Baseband transmission means sending a digital signal over a channel without changing the digital signal to an analog signal. Figure 3.18 shows baseband transmission.

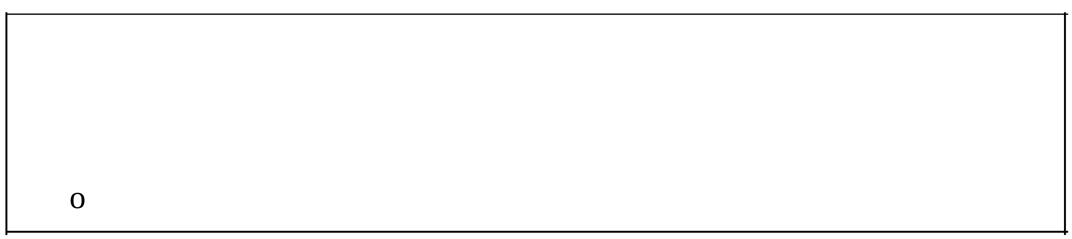
Figure 3.18 *Baseband transmission*



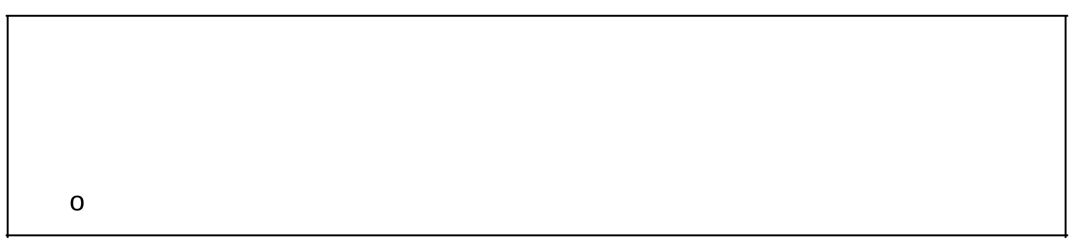
A digital signal is a composite analog signal with an infinite bandwidth.

Baseband transmission requires that we have a low-pass channel, a channel with a bandwidth that starts from zero. This is the case if we have a dedicated medium with a bandwidth constituting only one channel. For example, the entire bandwidth of a cable connecting two computers is one single channel. As another example, we may connect several computers to a bus, but not allow more than two stations to communicate at a time. Again we have a low-pass channel, and we can use it for baseband communication. Figure 3.19 shows two low-pass channels: one with a narrow bandwidth and the other with a wide bandwidth. We need to remember that a low-pass channel with infinite bandwidth is ideal, but we cannot have such a channel in real life. However, we can get close.

Figure 3.19 *Bandwidths of two low-pass channels*



a. Low-pass channel, wide bandwidth



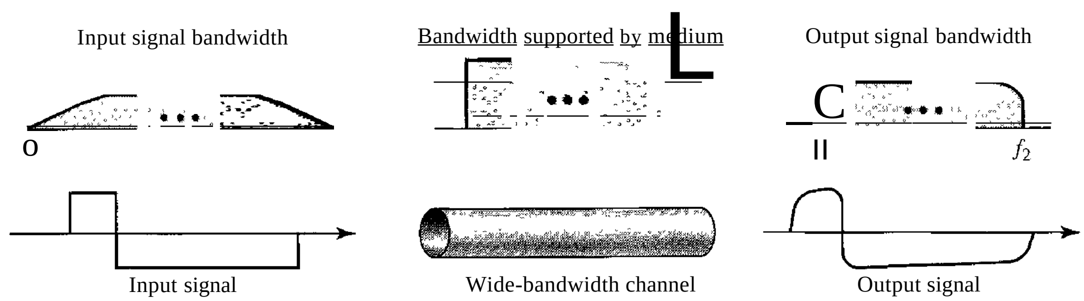
b. Low-pass channel, narrow bandwidth

Let us study two cases of a baseband communication: a low-pass channel with a wide bandwidth and one with a limited bandwidth.

Case 1: Low-Pass Channel with Wide Bandwidth

If we want to preserve the exact form of a nonperiodic digital signal with vertical segments vertical and horizontal segments horizontal, we need to send the entire spectrum, the continuous range of frequencies between zero and infinity. This is possible if we have a dedicated medium with an infinite bandwidth between the sender and receiver that preserves the exact amplitude of each component of the composite signal. Although this may be possible inside a computer (e.g., between CPU and memory), it is not possible between two devices. Fortunately, the amplitudes of the frequencies at the border of the bandwidth are so small that they can be ignored. This means that if we have a medium, such as a coaxial cable or fiber optic, with a very wide bandwidth, two stations can communicate by using digital signals with very good accuracy, as shown in Figure 3.20. Note that!i is close to zero, andh is very high.

Figure 3.20 *Baseband transmission using a dedicated medium*



Although the output signal is not an exact replica of the original signal, the data can still be deduced from the received signal. Note that although some of the frequencies are blocked by the medium, they are not critical.

> Baseband transmission of a digital signal that preserves the shape of the digital signal is possible only if we have a low-pass channel with an infinite or very wide bandwidth.

Example 3.21

An example of a dedicated channel where the entire bandwidth of the medium is used as one single channel is a LAN. Almost every wired LAN today uses a dedicated channel for two stations communicating with each other. In a bus topology LAN with multipoint connections, only two stations can communicate with each other at each moment in time (timesharing); the other stations need to refrain from sending data. In a star topology LAN, the entire channel between each station and the hub is used for communication between these two entities. We study LANs in Chapter 14.

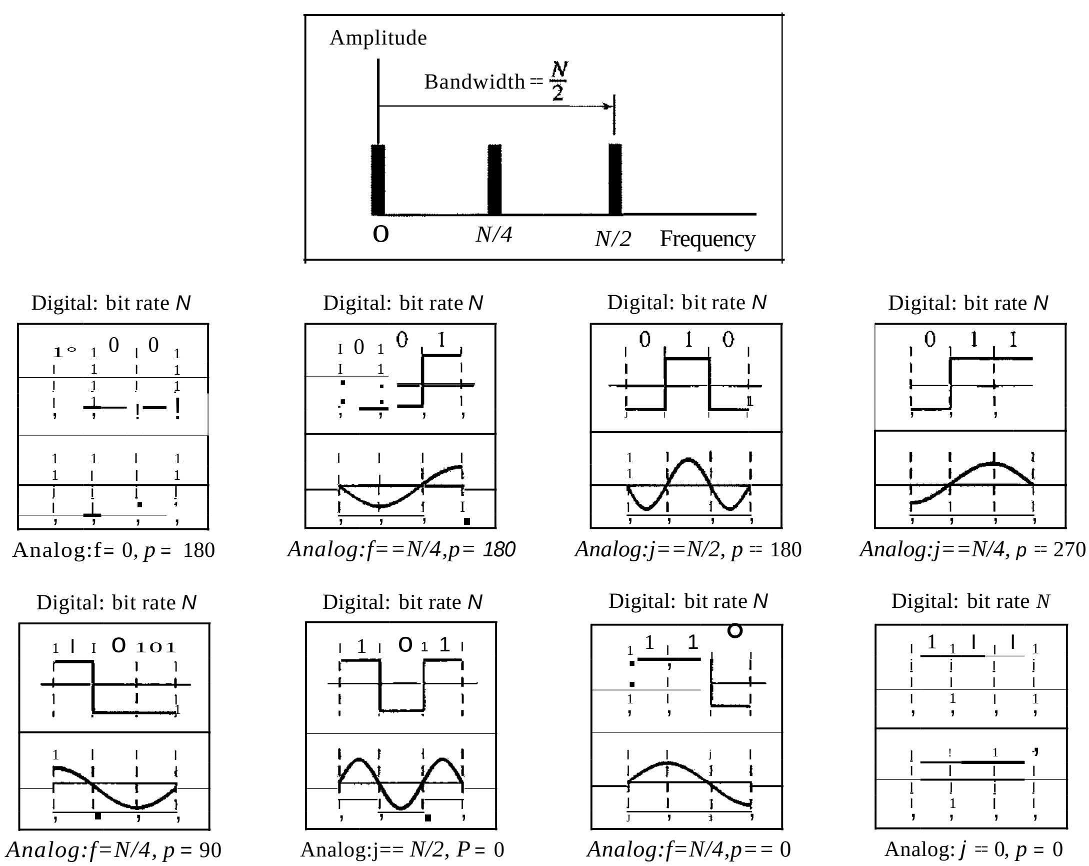
Case 2: Low-Pass Channel with Limited Bandwidth

In a low-pass channel with limited bandwidth, we approximate the digital signal with an analog signal. The level of approximation depends on the bandwidth available.

Rough Approximation Let us assume that we have a digital signal of bit rate *N*. If we want to send analog signals to roughly simulate this signal, we need to consider the worst case, a maximum number of changes in the digital signal. This happens when the signal carries the sequence 01010101 ... or the sequence 10101010· ... To simulate these two cases, we need an analog signal of frequency $f = N12$. Let 1 be the positive peak value and O be the negative peak value. We send 2 bits in each cycle; the frequency of the analog signal is one-half of the bit rate, or *N12.* However, just this one frequency cannot make all patterns; we need more components. The maximum frequency is *NI2.* As an example of this concept, let us see how a digital signal with a 3-bit pattern can be simulated by using analog signals. Figure 3.21 shows the idea. The two similar cases (000 and 111) are simulated with a signal with frequency $f = 0$ and a phase of 180° for 000 and a phase of 0° for 111. The two worst cases (010 and 101) are simulated with an analog signal with frequency $f = NI2$ and phases of 180° and 0°. The other four cases can only be simulated with an analog signal with $f = NI4$ and phases of 180°, 270°, 90°, and 0°. In other words, we need a channel that can handle frequencies 0, *N14,* and *NI2.* This rough approximation is referred to as using the first harmonic *(NI2)* frequency. The required bandwidth is

$$\text{Bandwidth} = \frac{N}{2} - 0 = \frac{N}{2}$$

Figure 3.21 *Rough approximation of a digital signal using the first harmonic for worst case*



Better Approximation To make the shape of the analog signal look more like that of a digital signal, we need to add more harmonics of the frequencies. We need to increase the bandwidth. We can increase the bandwidth to *3N12*, *5N12*, 7*NI2,* and so on. Figure 3.22 shows the effect of this increase for one of the worst cases, the pattern 010.

CHAPTER 4

Digital Transmission

A computer network is designed to send information from one point to another. This information needs to be converted to either a digital signal or an analog signal for transmission. In this chapter, we discuss the first choice, conversion to digital signals; in Chapter 5, we discuss the second choice, conversion to analog signals.

We discussed the advantages and disadvantages of digital transmission over analog transmission in Chapter 3. In this chapter, we show the schemes and techniques that we use to transmit data digitally. First, we discuss digital-to-digital conversion techniques, methods which convert digital data to digital signals. Second, we discuss analog-to-digital conversion techniques, methods which change an analog signal to a digital signaL Finally, we discuss transmission modes.

4.1 DIGITAL-TO-DIGITAL CONVERSION

In Chapter 3, we discussed data and signals. We said that data can be either digital or analog. We also said that signals that represent data can also be digital or analog. In this section, we see how we can represent digital data by using digital signals. The conversion involves three techniques: line coding, block coding, and scrambling. Line coding is always needed; block coding and scrambling mayor may not be needed.

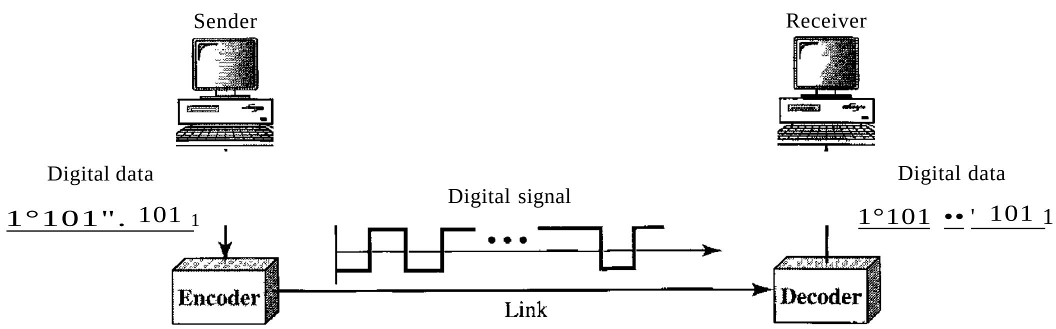
Line Coding

Line coding is the process of converting digital data to digital signals. We assume that data, in the form of text, numbers, graphical images, audio, or video, are stored in computer memory as sequences of bits (see Chapter 1). Line coding converts a sequence of bits to a digital signal. At the sender, digital data are encoded into a digital signal; at the receiver, the digital data are recreated by decoding the digital signal. Figure 4.1 shows the process.

Characteristics

Before discussing different line coding schemes, we address their common characteristics.

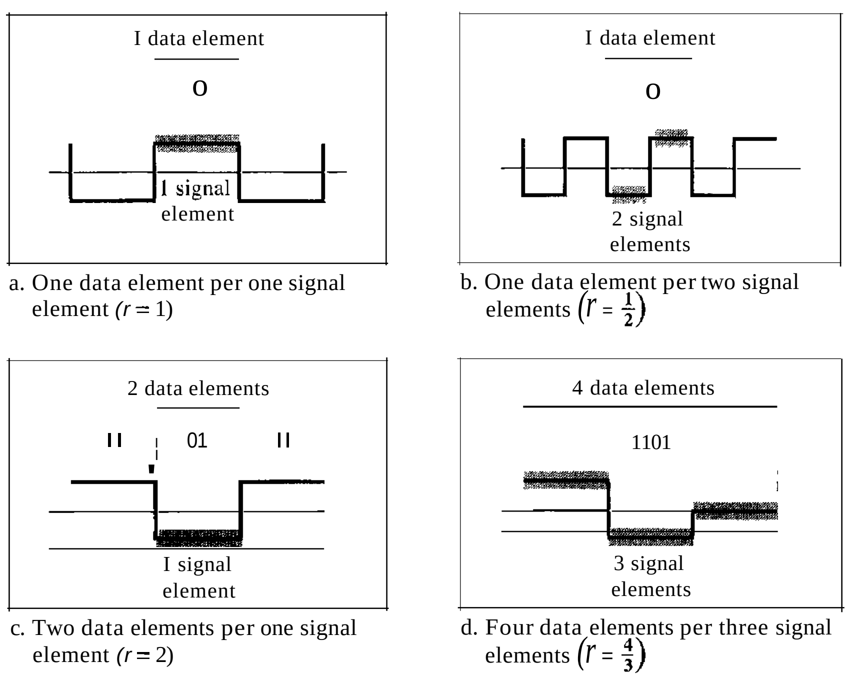
Figure 4.1 *Line coding and decoding*



Signal Element Versus Data Element Let us distinguish between a data element and a signal element. In data communications, our goal is to send data elements. A data element is the smallest entity that can represent a piece of information: this is the bit. In digital data communications, a signal element carries data elements. A signal element is the shortest unit (timewise) of a digital signal. In other words, data elements are what we need to send; signal elements are what we can send. Data elements are being carried; signal elements are the carriers.

We define a ratio r which is the number of data elements carried by each signal element. Figure 4.2 shows several situations with different values of r.

Figure 4.2 *Signal element versus data element*



a. One data element per one signal element ($r = 1$)

b. One data element per two signal elements $\left(r = \frac{1}{2}\right)$

c. Two data elements per one signal element ($r = 2$)

d. Four data elements per three signal elements $\left(r = \frac{4}{3}\right)$

In part a of the figure, one data element is carried by one signal element ($r = 1$). In part b of the figure, we need two signal elements (two transitions) to carry each data element $(r = \frac{1}{2})$. We will see later that the extra signal element is needed to guarantee synchronization. In part c of the figure, a signal element carries two data elements $(r = 2)$. Finally, in part d, a group of 4 bits is being carried by a group of three signal elements $(r = \frac{4}{3})$. For every line coding scheme we discuss, we will give the value of r.

An analogy may help here. Suppose each data element is a person who needs to be carried from one place to another. We can think of a signal element as a vehicle that can carry people. When $r = 1$, it means each person is driving a vehicle. When $r > 1$, it means more than one person is travelling in a vehicle (a carpool, for example). We can also have the case where one person is driving a car and a trailer $(r = \frac{1}{2})$.

Data Rate Versus Signal Rate The **data rate** defines the number of data elements (bits) sent in 1s. The unit is bits per second (bps). The **signal rate** is the number of signal elements sent in 1s. The unit is the baud. There are several common terminologies used in the literature. The data rate is sometimes called the **bit rate**; the signal rate is sometimes called the **pulse rate**, the **modulation rate**, or the **baud rate**.

One goal in data communications is to increase the data rate while decreasing the signal rate. Increasing the data rate increases the speed of transmission; decreasing the signal rate decreases the bandwidth requirement. In our vehicle-people analogy, we need to carry more people in fewer vehicles to prevent traffic jams. We have a limited *bandwidth* in our transportation system.

We now need to consider the relationship between data rate and signal rate (bit rate and baud rate). This relationship, of course, depends on the value of r. It also depends on the data pattern. If we have a data pattern of all 1s or all 0s, the signal rate may be different from a data pattern of alternating 0s and 1s. To derive a formula for the relationship, we need to define three cases: the worst, best, and average. The worst case is when we need the maximum signal rate; the best case is when we need the minimum. In data communications, we are usually interested in the average case. We can formulate the relationship between data rate and signal rate as

$$S = c \times N \times \frac{1}{r} \quad \text{baud}$$

where N is the data rate (bps); c is the case factor, which varies for each case; S is the number of signal elements; and r is the previously defined factor.

Example 4.1

A signal is carrying data in which one data element is encoded as one signal element $(r = 1)$. If the bit rate is 100 kbps, what is the average value of the baud rate if c is between 0 and 1?

Solution

We assume that the average value of c is $\frac{1}{2}$. The baud rate is then

$$S = c \times N \times \frac{1}{r} = \frac{1}{2} \times 100{,}000 \times \frac{1}{1} = 50{,}000 = 50 \text{ kbaud}$$

Bandwidth We discussed in Chapter 3 that a digital signal that carries infonnation is nonperiodic. We also showed that the bandwidth of a nonperiodic signal is continuous with an infinite range. However, most digital signals we encounter in real life have a bandwidth with finite values. In other words, the bandwidth is theoretically infinite, but many of the components have such a small amplitude that they can be ignored. The effective bandwidth is finite. From now on, when we talk about the bandwidth of a digital signal, we need to remember that we are talking about this effective bandwidth.

Although the actual bandwidth of a digital signal is infinite, the effective bandwidth is finite.

We can say that the baud rate, not the bit rate, determines the required bandwidth for a digital signal. If we use the transpOltation analogy, the number of vehicles affects the traffic, not the number of people being carried. More changes in the signal mean injecting more frequencies into the signal. (Recall that frequency means change and change means frequency.) The bandwidth reflects the range of frequencies we need. There is a relationship between the baud rate (signal rate) and the bandwidth. Bandwidth is a complex idea. When we talk about the bandwidth, we normally define a range of frequencies. We need to know where this range is located as well as the values of the lowest and the highest frequencies. In addition, the amplitude (if not the phase) of each component is an impOltant issue. In other words, we need more information about the bandwidth than just its value; we need a diagram of the bandwidth. We will show the bandwidth for most schemes we discuss in the chapter. For the moment, we can say that the bandwidth (range of frequencies) is proportional to the signal rate (baud rate). The minimum bandwidth can be given as

$$B_{min} = c \times N \times \frac{1}{r}$$

We can solve for the maximum data rate if the bandwidth of the channel is given.

$$N_{max} = \frac{1}{c} \times B \times r$$

Example 4.2

The maximum data rate of a channel (see Chapter 3) is $N_{max} = 2 \times B \times \log_2 L$ (defined by the Nyquist formula). Does this agree with the previous formula for N_{max}?

Solution

A signal with L levels actually can carry $\log_2 L$ bits per level. If each level corresponds to one signal element and we assume the average case ($c = \frac{1}{2}$), then we have

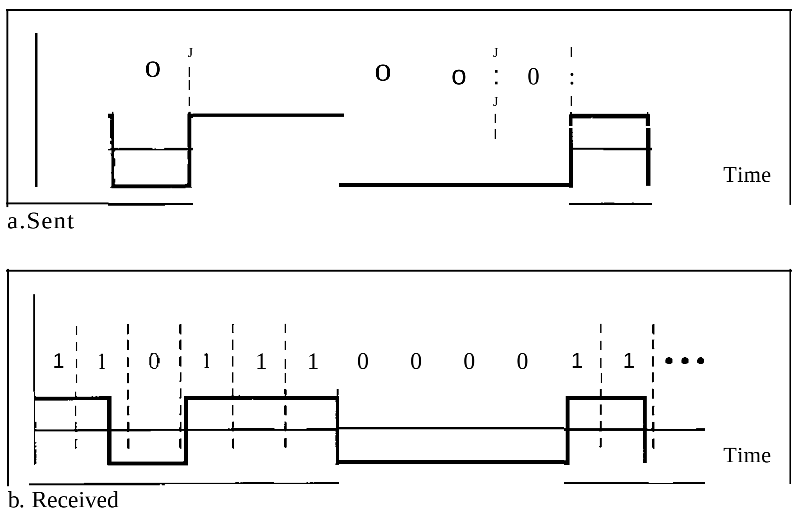
$$N_{max} = \frac{1}{c} \times B \times r = 2 \times B \times \log_2 L$$

Baseline Wandering In decoding a digital signal, the receiver calculates a running average of the received signal power. This average is called the *baseline*. The incoming signal power is evaluated against this baseline to determine the value of the data element. A long string of 0s or 1s can cause a drift in the baseline (baseline wandering) and make it difficult for the receiver to decode correctly. A good line coding scheme needs to prevent baseline wandering.

DC Components When the voltage level in a digital signal is constant for a while, the spectrum creates very low frequencies (results of Fourier analysis). These frequencies around zero, called DC (direct-current) *components,* present problems for a system that cannot pass low frequencies or a system that uses electrical coupling (via a transformer). For example, a telephone line cannot pass frequencies below 200 Hz. Also a long-distance link may use one or more transformers to isolate different parts of the line electrically. For these systems, we need a scheme with no DC component.

Self-synchronization To correctly interpret the signals received from the sender, the receiver's bit intervals must correspond exactly to the sender's bit intervals. If the receiver clock is faster or slower, the bit intervals are not matched and the receiver might misinterpret the signals. Figure 4.3 shows a situation in which the receiver has a shorter bit duration. The sender sends 10110001, while the receiver receives 110111000011.

Figure 4.3 *Effect of lack of synchronization*



A self-synchronizing digital signal includes timing information in the data being transmitted. This can be achieved if there are transitions in the signal that alert the receiver to the beginning, middle, or end of the pulse. If the receiver's clock is out of synchronization, these points can reset the clock.

Example 4.3

In a digital transmission, the receiver clock is 0.1 percent faster than the sender clock. How many extra bits per second does the receiver receive if the data rate is 1 kbps? How many if the data rate is 1 Mbps?

Solution

At 1 kbps, the receiver receives 1001 bps instead of 1000 bps.

1000 bits sent 1001 bits received 1 extra bps

At 1 Mbps, the receiver receives 1,001,000 bps instead of 1,000,000 bps.

1,000,000 bits sent 1,001,000 bits received 1000 extra bps

Built-in Error Detection It is desirable to have a built-in error-detecting capability in the generated code to detect some of or all the errors that occurred during transmission. Some encoding schemes that we will discuss have this capability to some extent.

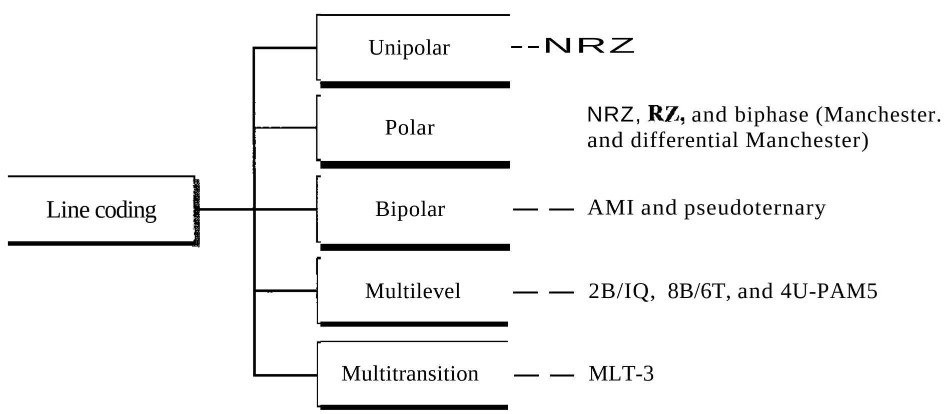
Immunity to Noise and Interference Another desirable code characteristic is a code that is immune to noise and other interferences. Some encoding schemes that we will discuss have this capability.

Complexity A complex scheme is more costly to implement than a simple one. For example, a scheme that uses four signal levels is more difficult to interpret than one that uses only two levels.

Line Coding Schemes

We can roughly divide line coding schemes into five broad categories, as shown in Figure 4.4.

Figure 4.4 *Line coding schemes*



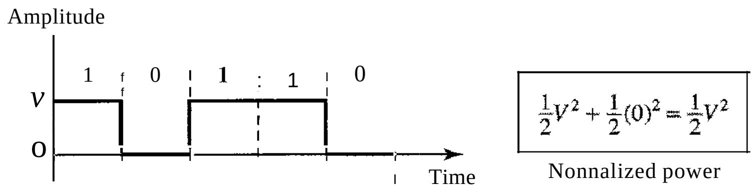
There are several schemes in each category. We need to be familiar with all schemes discussed in this section to understand the rest of the book. This section can be used as a reference for schemes encountered later.

Unipolar Scheme

In a unipolar scheme, all the signal levels are on one side of the time axis, either above or below.

NRZ (Non-Return-to-Zero) Traditionally, a unipolar scheme was designed as a non-return-to-zero (NRZ) scheme in which the positive voltage defines bit I and the zero voltage defines bit 0. It is called NRZ because the signal does not return to zero at the middle of the bit. Figure 4.5 show a unipolar NRZ scheme.

Figure 4.5 *Unipolar NRZ scheme*



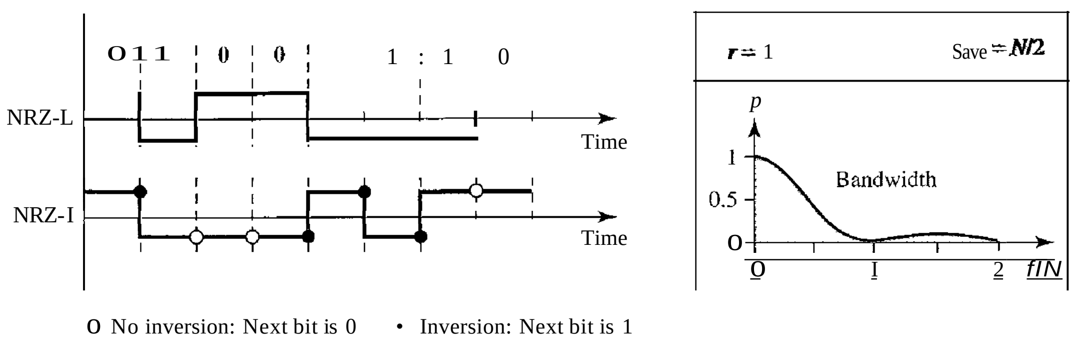
Compared with its polar counterpart (see the next section), this scheme is very costly. As we will see shortly, the normalized power (power needed to send 1 bit per unit line resistance) is double that for polar NRZ. For this reason, this scheme is normally not used in data communications today.

Polar Schemes

In polar schemes, the voltages are on the both sides of the time axis. For example, the voltage level for 0 can be positive and the voltage level for I can be negative.

Non-Return-to-Zero (NRZ) In polar NRZ encoding, we use two levels of voltage amplitude. We can have two versions of polar NRZ: NRZ-Land NRZ-I, as shown in Figure 4.6. The figure also shows the value of *r*, the average baud rate, and the bandwidth. In the first variation, NRZ-L (NRZ-Level), the level of the voltage determines the value of the bit. In the second variation, NRZ-I (NRZ-Invert), the change or lack of change in the level of the voltage determines the value of the bit. If there is no change, the bit is 0; if there is a change, the bit is 1.

Figure 4.6 *Polar NRZ-L and NRZ-I schemes*



In NRZ-L the level of the voltage determines the value of the bit. **In** NRZ-I the inversion or the lack of inversion determines the value of the bit.

Let us compare these two schemes based on the criteria we previously defined. Although baseline wandering is a problem for both variations, it is twice as severe in NRZ-L. If there is a long sequence of 0s or Is in NRZ-L, the average signal power becomes skewed. The receiver might have difficulty discerning the bit value. In NRZ-I this problem occurs only for a long sequence of as. If somehow we can eliminate the long sequence of as, we can avoid baseline wandering. We will see shortly how this can be done.

The synchronization problem (sender and receiver clocks are not synchronized) also exists in both schemes. Again, this problem is more serious in NRZ-L than in NRZ-I. While a long sequence of as can cause a problem in both schemes, a long sequence of 1s affects only NRZ-L.

Another problem with NRZ-L occurs when there is a sudden change of polarity in the system. For example, if twisted-pair cable is the medium, a change in the polarity of the wire results in all as interpreted as Is and all Is interpreted as as. NRZ-I does not have this problem. Both schemes have an average signal rate of *NI2* Bd.

NRZ-L and NRZ-J both have an average signal rate of *NI2* Bd.

Let us discuss the bandwidth. Figure 4.6 also shows the normalized bandwidth for both variations. The vertical axis shows the power density (the power for each I Hz of bandwidth); the horizontal axis shows the frequency. The bandwidth reveals a very serious problem for this type of encoding. The value of the power density is velY high around frequencies close to zero. This means that there are DC components that carry a high level of energy. As a matter of fact, most of the energy is concentrated in frequencies between a and *NII.* This means that although the average of the signal rate is *N12,* the energy is not distributed evenly between the two halves.

NRZ-L and NRZ-J both have a DC component problem.

Example 4.4

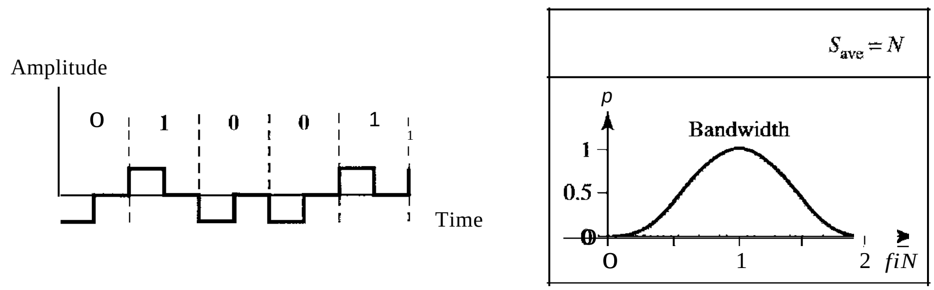
A system is using NRZ-I to transfer 10-Mbps data. What are the average signal rate and minimum bandwidth?

Solution

The average signal rate is S = *NI2* = 500 kbaud. The minimum bandwidth for this average baud rate is B_{min} = S = 500 kHz.

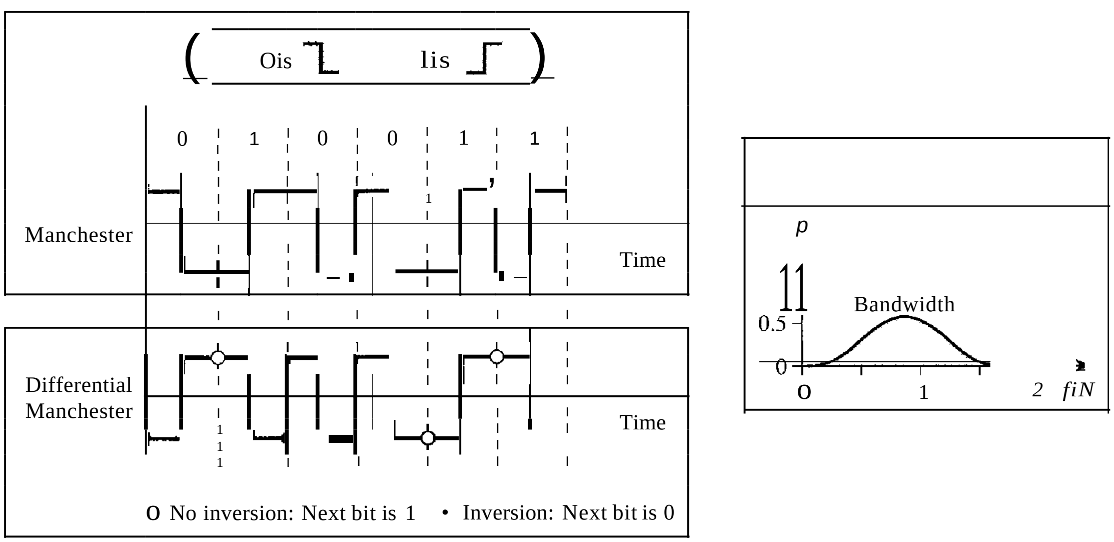
Return to Zero (RZ) The main problem with NRZ encoding occurs when the sender and receiver clocks are not synchronized. The receiver does not know when one bit has ended and the next bit is starting. One solution is the return-to-zero (RZ) scheme, which uses three values: positive, negative, and zero. In RZ, the signal changes not between bits but during the bit. In Figure 4.7 we see that the signal goes to 0 in the middle of each bit. It remains there until the beginning of the next bit. The main disadvantage of RZ encoding is that it requires two signal changes to encode a bit and therefore occupies greater bandwidth. The same problem we mentioned, a sudden change of polarity resulting in all as interpreted as 1s and all 1s interpreted as as, still exist here, but there is no DC component problem. Another problem is the complexity: RZ uses three levels of voltage, which is more complex to create and discern. As a result of all these deficiencies, the scheme is not used today. Instead, it has been replaced by the better-performing Manchester and differential Manchester schemes (discussed next).

Figure 4.7 *Polar RZ scheme*



Biphase: Manchester and Differential Manchester The idea of RZ (transition at the middle of the bit) and the idea of NRZ-L are combined into the Manchester scheme. In Manchester encoding, the duration of the bit is divided into two halves. The voltage remains at one level during the first half and moves to the other level in the second half. The transition at the middle of the bit provides synchronization. Differential Manchester, on the other hand, combines the ideas of RZ and NRZ-I. There is always a transition at the middle of the bit, but the bit values are determined at the beginning of the bit. If the next bit is 0, there is a transition; if the next bit is 1, there is none. Figure 4.8 shows both Manchester and differential Manchester encoding.

Figure 4.8 *Polar biphase: Manchester and differential Manchester schemes*



In Manchester and differential Manchester encoding, the transition at the middle of the bit is used for synchronization.

The Manchester scheme overcomes several problems associated with NRZ-L, and differential Manchester overcomes several problems associated with NRZ-I. First, there is no baseline wandering. There is no DC component because each bit has a positive and negative voltage contribution. The only drawback is the signal rate. The signal rate for Manchester and differential Manchester is double that for NRZ. The reason is that there is always one transition at the middle of the bit and maybe one transition at the end of each bit. Figure 4.8 shows both Manchester and differential Manchester encoding schemes. Note that Manchester and differential Manchester schemes are also called biphase schemes.

The minimum bandwidth of Manchester and differential Manchester is 2 times that of NRZ.

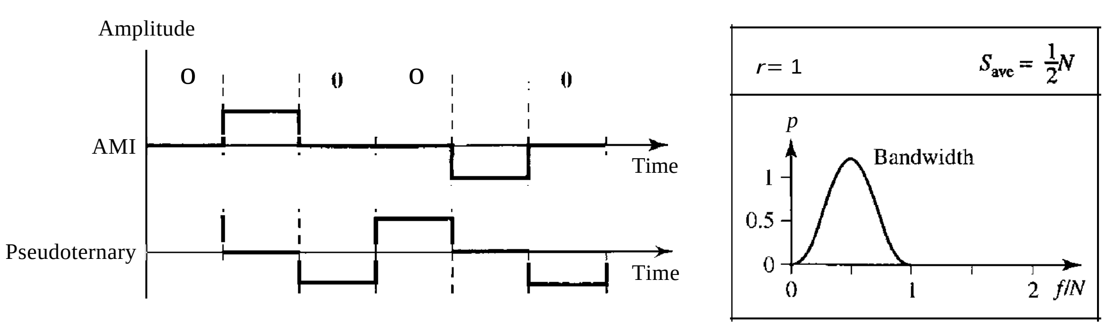
Bipolar Schemes

In bipolar encoding (sometimes called *multilevel binary)*, there are three voltage levels: positive, negative, and zero. The voltage level for one data element is at zero, while the voltage level for the other element alternates between positive and negative.

In bipolar encoding, we use three levels: positive, zero, and negative.

AMI and Pseudoternary Figure 4.9 shows two variations of bipolar encoding: AMI and pseudoternary. A common bipolar encoding scheme is called bipolar alternate mark inversion (AMI). In the term *alternate mark inversion,* the word *mark* comes from telegraphy and means 1. So AMI means alternate I inversion. A neutral zero voltage represents binary 0. Binary Is are represented by alternating positive and negative voltages. A variation of AMI encoding is called pseudoternary in which the 1 bit is encoded as a zero voltage and the 0 bit is encoded as alternating positive and negative voltages.

Figure 4.9 *Bipolar schemes: AMI and pseudoternary*



The bipolar scheme was developed as an alternative to NRZ. The bipolar scheme has the same signal rate as NRZ, but there is no DC component. The NRZ scheme has most of its energy concentrated near zero frequency, which makes it unsuitable for transmission over channels with poor performance around this frequency. The concentration of the energy in bipolar encoding is around frequency *N12.* Figure 4.9 shows the typical energy concentration for a bipolar scheme.

One may ask why we do not have DC component in bipolar encoding. We can answer this question by using the Fourier transform, but we can also think about it intuitively. If we have a long sequence of 1s, the voltage level alternates between positive and negative; it is not constant. Therefore, there is no DC component. For a long sequence of 0s, the voltage remains constant, but its amplitude is zero, which is the same as having no DC component. In other words, a sequence that creates a constant zero voltage does not have a DC component.

AMI is commonly used for long-distance communication, but it has a synchronization problem when a long sequence of 0s is present in the data. Later in the chapter, we will see how a scrambling technique can solve this problem.

Multilevel Schemes

The desire to increase the data speed or decrease the required bandwidth has resulted in the creation of many schemes. The goal is to increase the number of bits per baud by encoding a pattern of m data elements into a pattern of n signal elements. We only have two types of data elements (0s and 1s), which means that a group of m data elements can produce a combination of 2^m data patterns. We can have different types of signal elements by allowing different signal levels. If we have L different levels, then we can produce L^n combinations of signal patterns. If $2^m = L^n$, then each data pattern is encoded into one signal pattern. If $2^m < L^n$, data patterns occupy only a subset of signal patterns. The subset can be carefully designed to prevent baseline wandering, to provide synchronization, and to detect errors that occurred during data transmission. Data encoding is not possible if $2^m > L^n$ because some of the data patterns cannot be encoded.

The code designers have classified these types of coding as *mBnL*, where m is the length of the binary pattern, B means binary data, n is the length of the signal pattern, and L is the number of levels in the signaling. A letter is often used in place of L: B (binary) for $L = 2$, T (ternary) for $L = 3$, and Q (quaternary) for $L = 4$. Note that the first two letters define the data pattern, and the second two define the signal pattern.

In *mBnL* schemes, a pattern of m data elements is encoded as a pattern of n signal elements in which $2^m \leq L^n$.

2B1Q The first *mBnL* scheme we discuss, two binary, one quaternary (2B1Q), uses data patterns of size 2 and encodes the 2-bit patterns as one signal element belonging to a four-level signal. In this type of encoding $m = 2$, $n = 1$, and $L = 4$ (quatemary). Figure 4.10 shows an example of a 2B1Q signal.

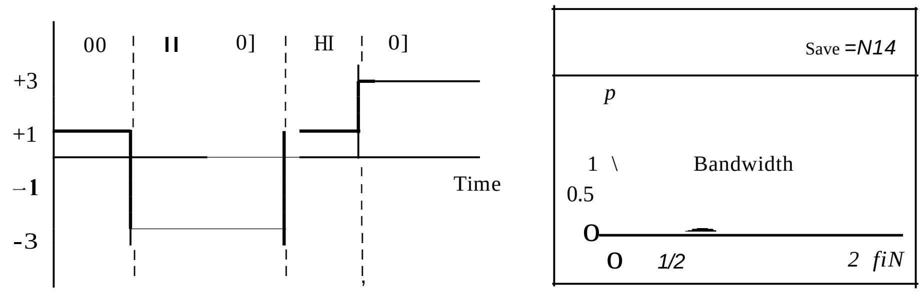
The average signal rate of 2B1Q is $S = N/4$. This means that using 2B1Q, we can send data 2 times faster than by using NRZ-L. However, 2B1Q uses four different signal levels, which means the receiver has to discern four different thresholds. The reduced bandwidth comes with a price. There are no redundant signal patterns in this scheme because $2^2 = 4^1$.

As we will see in Chapter 9, 2B1Q is used in DSL (Digital Subscriber Line) technology to provide a high-speed connection to the Internet by using subscriber telephone lines.

Figure 4.10 *Multilevel: 2B1Q scheme*

Next bits	Previous level: positive Next level	Previous level: negative Next level
00	+1	-1
01	+3	-3
10	-1	+1
11	-3	+3

Transition table

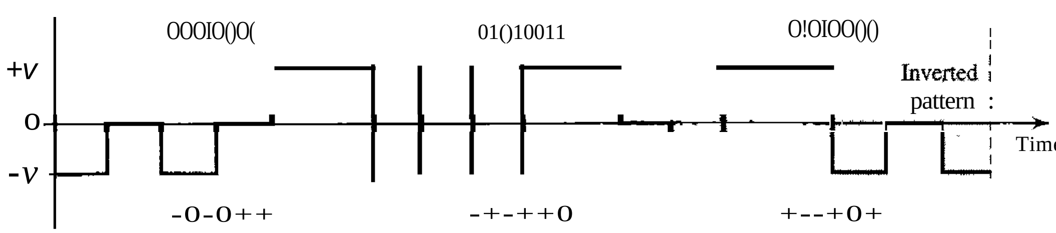


Assuming positive original level

8B6T A very interesting scheme is eight binary, six ternary (8B6T). This code is used with 100BASE-4T cable, as we will see in Chapter 13. The idea is to encode a pattern of 8 bits as a pattern of 6 signal elements, where the signal has three levels (ternary). In this type of scheme, we can have $2^8 = 256$ different data patterns and $3^6 = 478$ different signal patterns. The mapping table is shown in Appendix D. There are 478 - 256 = 222 redundant signal elements that provide synchronization and error detection. Part of the redundancy is also used to provide DC balance. Each signal pattern has a weight of 0 or +1 DC values. This means that there is no pattern with the weight -1. To make the whole stream Dc-balanced, the sender keeps track of the weight. If two groups of weight 1 are encountered one after another, the first one is sent as is, while the next one is totally inverted to give a weight of -1.

Figure 4.11 shows an example of three data patterns encoded as three signal patterns. The three possible signal levels are represented as -, 0, and +. The first 8-bit pattern 00010001 is encoded as the signal pattern -0-0++ with weight 0; the second 8-bit pattern 01010011 is encoded as - + - + + 0 with weight +1. The third bit pattern should be encoded as + - - + 0 + with weight +1. To create DC balance, the sender inverts the actual signal. The receiver can easily recognize that this is an inverted pattern because the weight is -1. The pattern is inverted before decoding.

Figure 4.11 *Multilevel: 8B6T scheme*



Adaptive DM

A better performance can be achieved if the value of δ is not fixed. In adaptive delta modulation, the value of δ changes according to the amplitude of the analog signal.

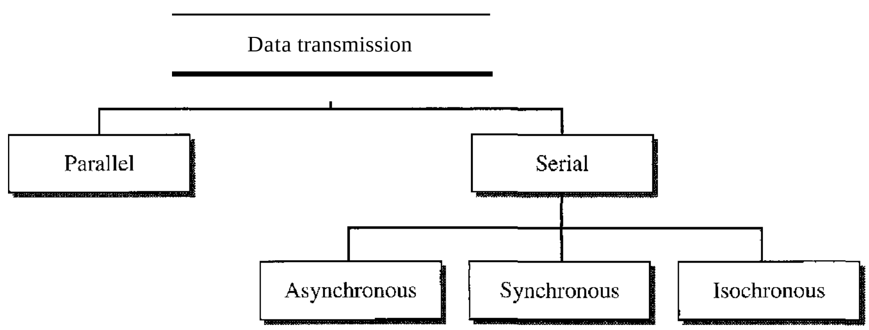
Quantization Error

It is obvious that DM is not perfect. Quantization error is always introduced in the process. The quantization error of DM, however, is much less than that for PCM.

4.3 TRANSMISSION MODES

Of primary concern when we are considering the transmission of data from one device to another is the wiring, and of primary concern when we are considering the wiring is the data stream. Do we send 1 bit at a time; or do we group bits into larger groups and, if so, how? The transmission of binary data across a link can be accomplished in either parallel or serial mode. In parallel mode, multiple bits are sent with each clock tick. In serial mode, 1 bit is sent with each clock tick. While there is only one way to send parallel data, there are three subclasses of serial transmission: asynchronous, synchronous, and isochronous (see Figure 4.31).

Figure 4.31 *Data transmission and modes*



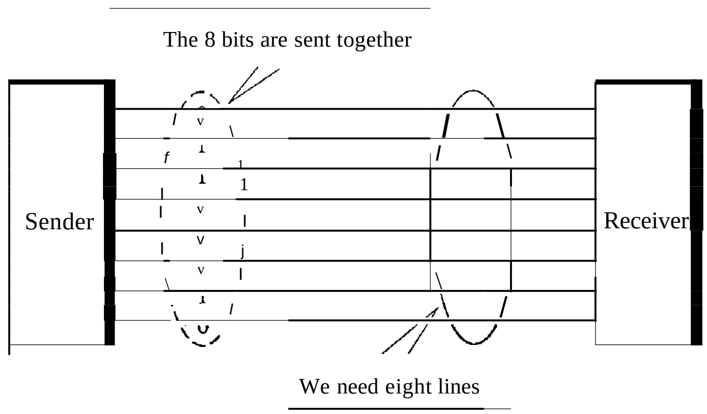
Parallel Transmission

Binary data, consisting of Is and Os, may be organized into groups of n bits each. Computers produce and consume data in groups of bits much as we conceive of and use spoken language in the form of words rather than letters. By grouping, we can send data n bits at a time instead of 1. This is called parallel transmission.

The mechanism for parallel transmission is a conceptually simple one: Use n wires to send n bits at one time. That way each bit has its own wire, and all n bits of one group can be transmitted with each clock tick from one device to another. Figure 4.32 shows how parallel transmission works for $n = 8$. Typically, the eight wires are bundled in a cable with a connector at each end.

The advantage of parallel transmission is speed. All else being equal, parallel transmission can increase the transfer speed by a factor of n over serial transmission.

Figure 4.32 *Parallel transmission*

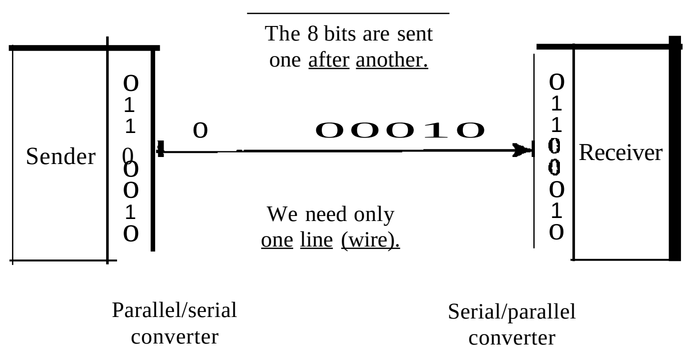


But there is a significant disadvantage: cost. Parallel transmission requires *n* communication lines (wires in the example) just to transmit the data stream. Because this is expensive, parallel transmission is usually limited to short distances.

Serial Transmission

In serial transmission one bit follows another, so we need only one communication channel rather than *n* to transmit data between two communicating devices (see Figure 4.33).

Figure 4.33 *Serial transmission*



The advantage of serial over parallel transmission is that with only one communication channel, serial transmission reduces the cost of transmission over parallel by roughly a factor of *n*.

Since communication within devices is parallel, conversion devices are required at the interface between the sender and the line (parallel-to-serial) and between the line and the receiver (serial-to-parallel).

Serial transmission occurs in one of three ways: asynchronous, synchronous, and isochronous.

Asynchronous Transmission

Asynchronous transmission is so named because the timing of a signal is unimportant. Instead, information is received and translated by agreed upon patterns. As long as those patterns are followed, the receiving device can retrieve the information without regard to the rhythm in which it is sent. Patterns are based on grouping the bit stream into bytes. Each group, usually 8 bits, is sent along the link as a unit. The sending system handles each group independently, relaying it to the link whenever ready, without regard to a timer.

Without synchronization, the receiver cannot use timing to predict when the next group will arrive. To alert the receiver to the arrival of a new group, therefore, an extra bit is added to the beginning of each byte. This bit, usually a 0, is called the start bit. To let the receiver know that the byte is finished, 1 or more additional bits are appended to the end of the byte. These bits, usually Is, are called stop bits. By this method, each byte is increased in size to at least 10 bits, of which 8 bits is information and 2 bits or more are signals to the receiver. In addition, the transmission of each byte may then be followed by a gap of varying duration. This gap can be represented either by an idle channel or by a stream of additional stop bits.

In asynchronous transmission, we send 1 start bit (0) at the beginning and 1 or more stop bits (Is) at the end of each byte. There may be a gap between each byte.

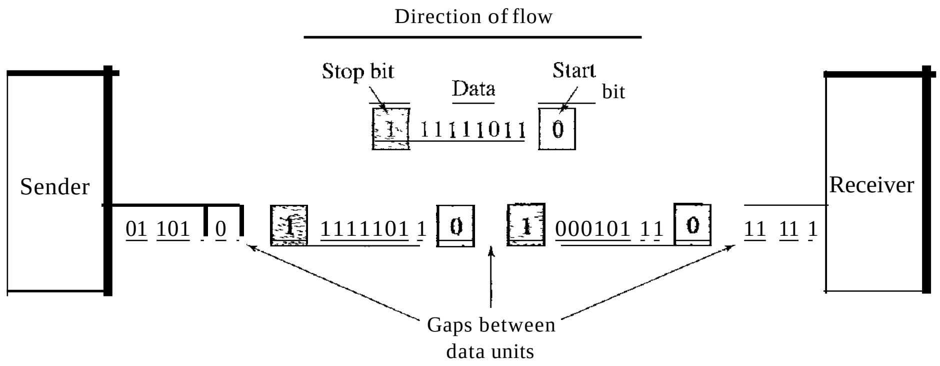
The start and stop bits and the gap alert the receiver to the beginning and end of each byte and allow it to synchronize with the data stream. This mechanism is called *asynchronous* because, at the byte level, the sender and receiver do not have to be synchronized. But within each byte, the receiver must still be synchronized with the incoming bit stream. That is, some synchronization is required, but only for the duration of a single byte. The receiving device resynchronizes at the onset of each new byte. When the receiver detects a start bit, it sets a timer and begins counting bits as they come in. After *n* bits, the receiver looks for a stop bit. As soon as it detects the stop bit, it waits until it detects the next start bit.

Asynchronous here means "asynchronous at the byte **level,"** but the bits are still synchronized; their durations are the same.

Figure 4.34 is a schematic illustration of asynchronous transmission. In this example, the start bits are as, the stop bits are 1s, and the gap is represented by an idle line rather than by additional stop bits.

The addition of stop and start bits and the insertion of gaps into the bit stream make asynchronous transmission slower than forms of transmission that can operate without the addition of control information. But it is cheap and effective, two advantages that make it an attractive choice for situations such as low-speed communication. For example, the connection of a keyboard to a computer is a natural application for asynchronous transmission. A user types only one character at a time, types extremely slowly in data processing terms, and leaves unpredictable gaps of time between each character.

Figure 4.34 *Asynchronous transmission*



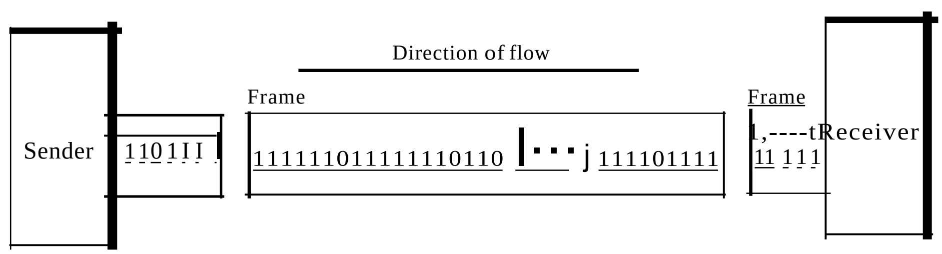
Synchronous Transmission

In synchronous transmission, the bit stream is combined into longer "frames," which may contain multiple bytes. Each byte, however, is introduced onto the transmission link without a gap between it and the next one. It is left to the receiver to separate the bit stream into bytes for decoding purposes. In other words, data are transmitted as an unbroken string of 1s and 0s, and the receiver separates that string into the bytes, or characters, it needs to reconstruct the information.

In synchronous transmission, we send bits one after another without start or stop bits or gaps. It is the responsibility of the receiver to group the bits.

Figure 4.35 gives a schematic illustration of synchronous transmission. We have drawn in the divisions between bytes. In reality, those divisions do not exist; the sender puts its data onto the line as one long string. If the sender wishes to send data in separate bursts, the gaps between bursts must be filled with a special sequence of 0s and 1s that means *idle.* The receiver counts the bits as they arrive and groups them in 8-bit units.

Figure 4.35 *Synchronous transmission*



Without gaps and start and stop bits, there is no built-in mechanism to help the receiving device adjust its bit synchronization midstream. Timing becomes very important, therefore, because the accuracy of the received information is completely dependent on the ability of the receiving device to keep an accurate count of the bits as they come in.

The advantage of synchronous transmission is speed. With no extra bits or gaps to introduce at the sending end and remove at the receiving end, and, by extension, with fewer bits to move across the link, synchronous transmission is faster than asynchronous transmission. For this reason, it is more useful for high-speed applications such as the transmission of data from one computer to another. Byte synchronization is accomplished in the data link layer.

We need to emphasize one point here. Although there is no gap between characters in synchronous serial transmission, there may be uneven gaps between frames.

Isochronous

In real-time audio and video, in which uneven delays between frames are not acceptable, synchronous transmission fails. For example, TV images are broadcast at the rate of 30 images per second; they must be viewed at the same rate. If each image is sent by using one or more frames, there should be no delays between frames. For this type of application, synchronization between characters is not enough; the entire stream of bits must be synchronized. The isochronous transmission guarantees that the data arrive at a fixed rate.

4.4 RECOMMENDED READING

For more details about subjects discussed in this chapter, we recommend the following books. The items in brackets [...] refer to the reference list at the end of the text.

Books

Digital to digital conversion is discussed in Chapter 7 of [Pea92], Chapter 3 of [CouOl], and Section 5.1 of [Sta04]. Sampling is discussed in Chapters 15, 16, 17, and 18 of [Pea92], Chapter 3 of [CouO!], and Section 5.3 of [Sta04]. [Hsu03] gives a good mathematical approach to modulation and sampling. More advanced materials can be found in [Ber96].

4.5 KEY TERMS

adaptive delta modulation
alternate mark inversion (AMI)
analog-to-digital conversion
asynchronous transmission
baseline
baseline wandering
baud rate
biphase
bipolar
bipolar with 8-zero substitution (B8ZS)
bit rate
block coding
companding and expanding
data element
data rate
DC component
delta modulation (DM)
differential Manchester
digital-to-digital conversion
digitization