CHAPTER 5

Overview of Wireless Communication

LEARNING OBJECTIVES

After studying this chapter, you should be able to:

- Explain the importance of unlicensed frequencies.
- Compute path loss for free space and real-world environments using the path loss exponent.
- Characterize the multipath and Doppler spreading characteristics of channels.
- Describe the approaches used to correct channel impairments.
- Describe the three major ways digital data can be encoded onto an analog signal.
- Determine performance of modulation schemes from E_b/N_0 curves.
- Describe and compare error recovery processes for error detection, retransmission/ARQ, and error correction.
- Describe the capabilities and bandwidth efficiency of codes in terms of their coding rate, Hamming distance, and coding gain.
- Present an overview of OFDM and OFDMA.
- Explain the value of orthogonal carriers.
- Describe the operation of the two major forms of spread spectrum: frequency hopping and direct sequence.

This chapter is a condensed version of Chapters 6-10. It provides a complete coverage of the wireless physical medium but at less depth for quicker coverage. It covers the same material and uses many of the same figures as the subsequent five chapters. If more depth is desired for a particular topic, the corresponding later chapter can be consulted.

5.1 SPECTRUM CONSIDERATIONS

The proper choice of the range of wireless frequencies over which a technology is to operate (i.e., its **spectrum**) is vital to its success. Some frequencies travel better over long distances; others penetrate obstacles such as buildings and walls more effectively. Wireless frequencies need to be shared with multiple types of users.

Regulation

The wireless medium is shared by a myriad of different types of users, applications, and traffic types. These are controlled by regulatory bodies to provide fair use while also meeting the key demands of society. The following differentiates signals from each other.

- Carrier Frequency: Each signal is shifted from its base frequency up to a carrier frequency. For example, a 22 MHz IEEE 802.11 signal might be shifted up to be centered at a carrier frequency of 2.412 GHz, so that it would occupy 2.401 to 2.423 GHz.
- Signal Power: Signals are limited in their propagation range by the allowed transmission power. At sufficient distances from each other, multiple users and groups can reuse the same spectrum.
- Multiple Access Scheme: Multiple users within a same spectrum range can share the spectrum by each having their own small slice of time or frequency;

this is known as Time Division Multiple Access (TDMA) or Frequency Division Multiple Access (FDMA). They might also encode their signals in different ways while sharing the same time and frequency; this is known as **Code Division Multiple Access (CDMA)**.

In the United States, the Federal Communications Commission (FCC) regulates these issues for different types of groups to share the wireless spectrum. Similar bodies operate throughout the world. In most cases, a license is required by the FCC to operate. In some cases, auctions are conducted for the purchase of these licenses. The FCC regulates which frequencies are government exclusive, nongovernment exclusive, or government/nongovernment shared. They provide for a variety of services, including the following:

- Aeronautical
- Amateur
- Broadcasting
- Maritime
- Meteorological
- Mobile
- Satellite
- Space

An illustration of the spectrum allocations by the FCC can be seen on the book Web site at corybeardwireless.com. FCC licenses are allocated for different uses so that there are no conflicts. This frequently causes spectrum to be underutilized, so researchers are exploring a new concept known as **dynamic spectrum access**. Here users would share spectrum among primary and secondary users. If primary users are not active, secondary users could use the spectrum but release the spectrum as soon as primary users need it. A technology known as **cognitive radio** would be implemented in the devices to scan wide bands of frequency to sense when spectrum is being used.

Within a given spectrum band, it is possible for frequencies to be allocated among multiple services. FCC licenses are allocated for these different services so that there are no conflicts. Frequency bands used for the technologies covered in this textbook are relatively narrow compared to the overall wide spectrum. Several technologies (e.g., IEEE 802.11 and 802.15) use the industrial, scientific, and medical (ISM) bands because those frequencies can be used without a license as long the transmitters stay within power limitations and use a spread spectrum technique. Some of these ISM bands are 915 $\pm$ 13 MHz, 2450 $\pm$ 50 MHz, 5.8 $\pm$ 0.75 GHz, and 57–64 GHz.

Propagation Modes

A signal radiated from an antenna travels along one of three routes: ground wave, sky wave, or line of sight (LOS). Figure 5.1 illustrates each type.

- **Ground wave propagation** (Figure 5.1a) more or less follows the contour of the earth and can propagate considerable distances, well over the visual horizon. This effect is found in frequencies up to about 3 MHz. Electromagnetic

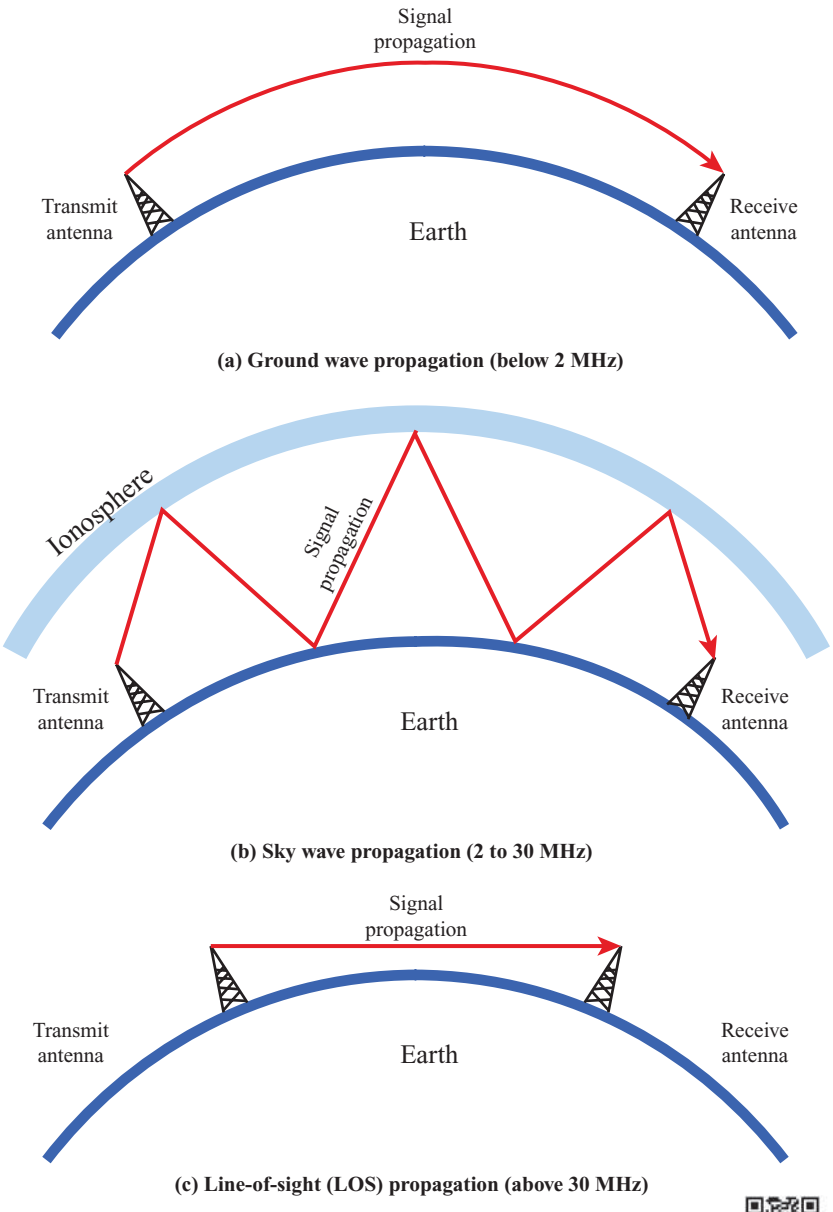


Figure 5.1 Wireless Propagation Modes

waves in this frequency range are scattered by the atmosphere in such a way that they do not penetrate the upper atmosphere. The best-known example of ground wave communication is AM radio.

- With **sky wave propagation** (Figure 5.1b), a signal from an earth-based antenna is ***refracted*** from the ionized layer of the upper atmosphere (ionosphere) back

down to earth. A sky-wave signal can travel through a number of hops, bouncing back and forth between the ionosphere and the earth's surface. With this propagation mode, a signal can be picked up thousands of kilometers from the transmitter. Sky waves generally operate between 3 and 30 MHz.

- **Line-of-sight propagation (LOS)** (Figure 5.1c) is necessary when neither ground wave nor sky wave propagation modes can operate. This generally occurs above 30 MHz. Most of the technologies we will discuss operate from 100s of MHz to a few GHz, so they operate in a line-of-sight mode. This does not mean that line of sight always requires complete free space between transmitters and receivers, however. Different frequencies will be attenuated by atmospheric effects or have capabilities for penetrating through surfaces (e.g., through walls, buildings, cars, etc.). For most materials, the ability to transmit through an object significantly degrades as frequency increases.

5.2 LINE-OF-SIGHT TRANSMISSION

With any communications system, the signal that is received will differ from the signal that is transmitted, due to various transmission impairments. For analog signals, these impairments introduce various random modifications that degrade the signal quality. For digital data, bit errors are introduced: A binary 1 is transformed into a binary 0, and vice versa. In this section, we examine the various impairments and comment on their effect on the information-carrying capacity of a communications link. Our concern in this book is mainly with LOS wireless transmission frequencies, and in this context, the most significant impairments are

- Attenuation and attenuation distortion
- Free space loss
- Noise
- Atmospheric absorption
- Multipath
- **Refraction**

Five Basic Propagation Mechanisms

There are five different mechanisms by which electromagnetic signals can transfer information from a transmitter to a receiver:

1. **Free-space propagation** transmits a wave when there are no obstructions. The signal strength decays as a function of distance.
2. **Transmission** propagates a signal as it penetrates in and through a medium. The signal is refracted at the surface of the medium to a different angle of transmission.
3. **Reflections** occur when electromagnetic waves impinge upon surfaces that are large relative to the wavelength of a signal.

4. **Diffraction** occurs when a signal is obstructed by an object with sharp edges. Secondary waves are then present behind the sharp edges to deliver the signal to a possibly shadowed receiver.
5. **Scattering** is involved when a signal interacts with large numbers of objects that are small relative to its wavelength. This can involve rough surfaces, foliage, street signs, etc. in a typical communication system.

The last four involve interacting objects. The dielectric and conducting properties of these objects affect the strength and angle of signal propagation when these interactions occur.

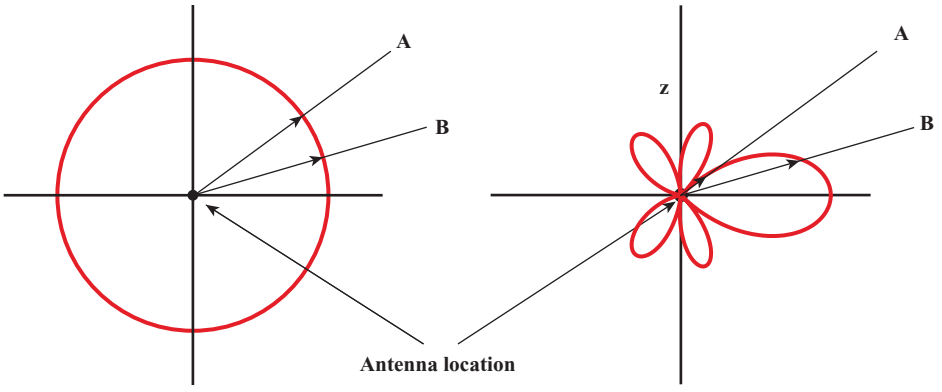
Antennas

Before examining free-space propagation, first it is important to have some understanding of antennas. An **antenna** can be defined as an electrical conductor or system of conductors used either for radiating electromagnetic energy or for collecting electromagnetic energy. For transmission of a signal, radio-frequency electrical energy from the transmitter is converted into electromagnetic energy by the antenna and radiated into the surrounding environment (atmosphere, space, water). For reception of a signal, electromagnetic energy impinging on the antenna is converted into radio-frequency electrical energy and fed into the receiver.

An antenna will radiate power in all directions but, typically, does not perform equally well in all directions. A common way to characterize the performance of an antenna is the **radiation pattern**, which is a graphical representation of the radiation properties of an antenna as a function of space coordinates. The simplest pattern is produced by an idealized antenna known as the isotropic antenna. An **isotropic antenna** is a point in space that radiates power in all directions equally. The actual radiation pattern for the isotropic antenna is a sphere with the antenna at the center. However, radiation patterns are almost always depicted as a two-dimensional cross section of the three-dimensional pattern. The pattern for the isotropic antenna is shown in Figure 5.2a. The distance from the antenna to each point on the radiation pattern is proportional to the power radiated from the antenna in that direction.

Figure 5.2b shows an actual directional antenna pattern produced from an array of antennas spaced apart by half of a wavelength and placed in a linear array. If weights are optimized, this pattern can be produced with a main lobe that is 60° wide. This requires four antennas. In this example, the main strength of the antenna is in the x direction. Notice that some energy is sent to the sides and back of the antenna in what are called the ***sidelobes***. There are also, however, ***nulls*** in the patterns where very little signal energy is sent in those directions.

The actual size of a radiation pattern is arbitrary. What is important is the *relative* distance from the antenna position in each direction. The relative distance determines the relative power. To determine the relative power in a given direction, a line is drawn from the antenna position at the appropriate angle, and the point of intercept with the radiation pattern is determined. Figure 5.2 shows a comparison of two transmission angles, A and B, drawn on the two radiation patterns. The



(a) Omnidirectional (b) Directional

Figure 5.2 Antenna Radiation Patterns

isotropic antenna produces an omnidirectional radiation pattern of equal strength in all directions, so the A and B vectors are of equal length. For the antenna pattern shown in Figure 5.2b, the B vector is longer than the A vector, indicating that more power is radiated in the B direction than in the A direction, and the relative lengths of the two vectors are proportional to the amount of power radiated in the two directions. Please note that this type of diagram shows relative ***antenna gain*** in each direction, not relative distance of coverage, although they are of course related.

Free Space Loss

For any type of wireless communication, the signal disperses with distance and causes **attenuation**. Energy dispersal can be viewed as radiating in a sphere with a receiver on the surface extracting energy on part of the surface area. A larger and larger sphere occurs as distance from the transmitter increases, so there is less energy per each unit of surface area. Therefore, an antenna with a fixed area will receive less signal power the farther it is from the transmitting antenna. For satellite communication, this is the primary mode of signal loss. Even if no other sources of attenuation or impairment are assumed, a transmitted signal attenuates over distance because the signal is being spread over a larger and larger area. This form of attenuation is known as **free space loss**, which can be expressed in terms of the ratio of the radiated power P_t to the power P_r received by the antenna or, in decibels, by taking 10 times the log of that ratio. For the ideal isotropic antenna, free space loss is

$$\frac{P_t}{P_r} = \frac{(4\pi d)^2}{\lambda^2} = \frac{(4\pi fd)^2}{c^2} \tag{5.1}$$

where

P_t = signal power at the transmitting antenna

P_r = signal power at the receiving antenna

λ = carrier wavelength

f = carrier frequency

d = propagation distance between antennas

c = speed of light (3×10^8 m/s)

and d and λ are in the same units (e.g., meters).

This can be recast in decibels as

$$L_{dB} = 10\log\frac{P_t}{P_r} = 20\log\left(\frac{4\pi d}{\lambda}\right) = -20\log(\lambda) + 20\log(d) + 21.98 \text{ dB}$$

$$= 20\log\left(\frac{4\pi fd}{\text{c}}\right) = 20\log(f) + 20\log(d) - 147.56 \text{ dB} \quad (5.2)$$

Figure 5.3 shows plots of curves from the free space loss equation.[1]

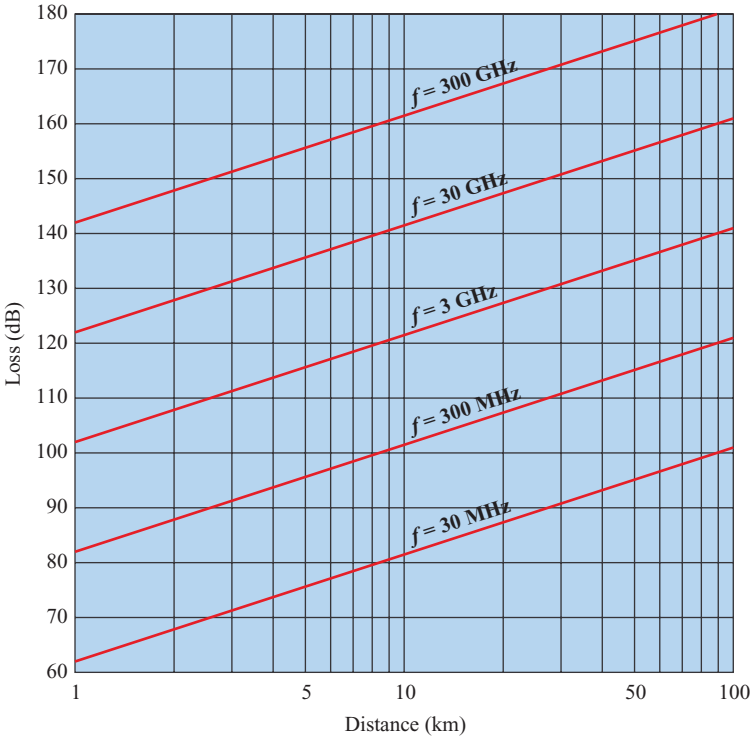


Figure 5.3 Free Space Loss

[1]As was mentioned in Appendix 2A, there is some inconsistency in the literature over the use of the terms *gain* and *loss*. Equation (5.2) follows the convention of Equation (2.2) in Section 2.4.

For other antennas, we must take into account the gain of the antenna, which yields the following free space loss equation:

$$\frac{P_t}{P_r} = \frac{(4\pi)^2(d)^2}{G_r G_t \lambda^2} = \frac{(\lambda d)^2}{A_r A_t} = \frac{(cd)^2}{f^2 A_r A_t}$$

where

G_t = gain of the transmitting antenna

G_r = gain of the receiving antenna

A_t = effective area of the transmitting antenna

A_r = effective area of the receiving antenna

The effective area of an antenna is related to the physical size of the antenna and to its shape.

We can recast this equation as

$$\begin{aligned} L_{dB} &= 20\log(\lambda) + 20\log(d) - 10\log(A_t A_r) \\ &= -20\log(f) + 20\log(d) - 10\log(A_t A_r) + 169.54 \text{ dB} \end{aligned} \tag{5.3}$$

Thus, for the same antenna dimensions and separation, the longer the carrier wavelength (lower the carrier frequency f), the higher is the free space path loss. It is interesting to compare Equations (5.2) and (5.3). Equation (5.2) indicates that as the frequency increases, the free space loss also increases, which would suggest that at higher frequencies, losses become more burdensome. However, Equation (5.3) shows that we can easily compensate for this increased loss with antenna gains. Since Equation (5.1) says there is increased gain at higher frequencies, in fact there is a net gain at higher frequencies, other factors remaining constant. Equation (5.2) shows that at a fixed distance an increase in frequency results in an increased loss measured by 20 log (f). However, if we take into account antenna gain, and fix antenna area, then the change in loss is measured by $-20\log(f)$; that is, there is actually a decrease in loss at higher frequencies.

Example 5.1 Determine the isotropic free space loss at 4 GHz for the shortest path to a synchronous satellite from earth (35,863 km). At 4 GHz, the wavelength is $(3 \times 10^8)/(4 \times 10^9) = 0.075$ m. Then,

$$L_{\text{dB}} = -20\log(0.075) + 20\log(35.853 \times 10^6) + 21.98 = 195.6 \text{ dB}$$

Now consider the antenna gain of both the satellite- and ground-based antennas. Typical values are 44 dB and 48 dB, respectively. The free-space loss is:

$$L_{\text{dB}} = 195.6 - 44 - 48 = 103.6 \text{ dB}$$

Now assume a transmit power of 250 W at the earth station. What is the power received at the satellite antenna? A power of 250 W translates into 24 dBW, so the power at the receiving antenna is $24 - 103.6 = -79.6$ dBW, where dBW is the decibel-watt, defined in Appendix 2A. This signal is approximately 10^{-8} W, still useable by receiver circuitry.

Path Loss Exponent in Practical Systems

Practical systems involve many types of obstructions that cause reflections, scattering, etc. Both theoretical and measurement-based models have shown that beyond a certain distance the average received signal power decreases logarithmically with distance according to a $10n\log(d)$ relationship where n is known as the ***path loss exponent*** [RAPP02]. Such models have been used extensively. Both Equations (5.2) and (5.3) showed a $20\log(d)$ term which came from a d^2 distance relationship, hence a path-loss exponent of $n = 2$. These should be replaced with the more general $10n\log(d)$ term as follows:

$$\frac{P_t}{P_r} = \left(\frac{4\pi}{\lambda}\right)^2 d^n = \left(\frac{4\pi f}{\mathrm{c}}\right)^2 d^n$$

$$L_{dB} = 10\log\frac{P_t}{P_r} = 10\log\left(\left(\frac{4\pi}{\lambda}\right)^2 d^n\right) = -20\log(\lambda) + 10n\log(d) + 21.98\,\mathrm{dB}$$

$$= 10\log\left(\left(\frac{4\pi f}{\mathrm{c}}\right)^2 d^n\right) = 20\log(f) + 10n\log(d) - 147.56\ \mathrm{dB} \tag{5.4}$$

Using effective areas and the general path loss exponent, n,

$$L_{dB} = 20\log(\lambda) + 10n\log(d) - 10\log(A_t A_r)$$
$$= -20\log(f) + 10n\log(d) - 10\log(A_t A_r) + 169.54\ \mathrm{dB} \tag{5.5}$$

Table 5.1 shows typical path loss exponents obtained for various environments. Note that in a building, LOS can be better than $n = 2$ (e.g., in hallways) since reflections help keep the signal stronger than if it decayed with distance as in free space.

Example 5.2 Compare the path loss in dB for two possible cellular environments where there is (1) free space between mobiles and base stations, and (2) urban area cellular radio with $n = 3.1$. Use 1.9 GHz at a distance of 1.5 km and assume isotropic antennas.

For free space using $n = 2.0$

$$L_{\mathrm{dB}} = 20\log(1.9\times10^9) + 10\times2.0\log(1.5\times10^3) - 147.56 = 101.53\ \mathrm{dB}$$

For urban cellular radio using $n = 3.1$

$$L_{\mathrm{dB}} = 20\log(1.9\times10^9) + 10\times3.1\log(1.5\times10^3) - 147.56 = 136.47\ \mathrm{dB}$$

Table 5.1 Path Loss Exponents for Different Environments [RAPP02]

Environment	Path Loss Exponent, n
Free space	2
Urban area cellular radio	2.7 to 3.5
Shadowed cellular radio	3 to 5
In building line-of-sight	1.6 to 1.8
Obstructed in building	4 to 6
Obstructed in factories	2 to 3

Example 5.3 Compare the range of coverage for two possible cellular environments where there is (1) free space between mobiles and base stations, and (2) urban area cellular radio with $n = 3.1$. Use 1.9 GHz and assume isotropic antennas. Assume the transmit power is 2 W and the received power must be above -110 dBW.

P_t in dB $= 10 \log (2) = 3.0$

Requirement is, therefore, $L_{\text{dB}} < 113$ dB

For free space using $n = 2.0$

$$L_{\text{dB}} = 20 \log(1.9 \times 10^9) + 10 \times 2.0 \log(d) - 147.56 < 113 \text{ dB}$$
$$10 \times 2.0 \log(d) < 74.99 \text{ dB}$$
$$d < 5.61 \text{ km}$$

For free space using $n = 2.0$

$$L_{\text{dB}} = 20 \log(1.9 \times 10^9) + 10 \times 3.1 \log(d) - 147.56 < 113 \text{ dB}$$
$$10 \times 3.1 \log(d) < 74.99 \text{ dB}$$
$$d < 262 \text{ m}$$

Models Derived from Empirical Measurements

In designing a wireless system, the communications engineer must take account of various propagation effects, the desired maximum transmit power level at the base station and the mobile units, the typical height of the mobile unit antenna, and the available height of the BS antenna. These factors will determine the coverage area of a wireless system. Unfortunately, the propagation effects are dynamic and difficult to predict. The best that can be done is to come up with a model based on empirical data and to apply that model to a given environment to develop guidelines. One of the most widely used models was developed by Okumura et al. [OKUM68] and subsequently refined by Hata [HATA80], commonly called the Okumura-Hata model. The original was a detailed analysis of the Tokyo area and produced path loss information for an urban environment. The Okumura-Hata model is an empirical formulation that takes into account a variety of environments and conditions. For an urban environment, predicted path loss is

$$L_{\text{dB}} = 69.55 + 26.16 \log f_c - 13.82 \log h_t - \text{A}(h_r) + (44.9 - 6.55 \log h_t) \log d \quad (5.6)$$

where

f_c = carrier frequency in MHz from 150 to 1500 MHz

h_t = height of transmitting antenna (base station) in m, from 30 to 300 m

h_r = height of receiving antenna (mobile unit) in m, from 1 to 10 m

d = propagation distance between antennas in km, from 1 to 20 km

$\text{A}(h_r)$ = correction factor for mobile unit antenna height

For a small or medium-sized city, the correction factor is given by

$$\text{A}(h_r) = (1.1 \log f_c - 0.7)\, h_r - (1.56 \log f_c - 0.8) \text{ dB}$$

And for a large city it is given by

$$A(h_r) = 8.29\,[\log(1.54\,h_r)]^2 - 1.1 \text{ dB} \qquad \text{for } f_c \leq 300 \text{ MHz}$$
$$A(h_r) = 3.2\,[\log(11.75\,h_r)]^2 - 4.97 \text{ dB} \qquad \text{for } f_c \geq 300 \text{ MHz}$$

To estimate the path loss in a suburban area, the formula for urban path loss in Equation (10.1) is modified as

$$L_{\text{dB}}(\text{suburban}) = L_{\text{dB}}(\text{urban small/medium city}) - 2[\log(f_c/28)]^2 - 5.4$$

And for the path loss in open or rural areas, the formula is modified as

$$L_{\text{dB}}(\text{open}) = L_{\text{dB}}(\text{urban small/medium city}) - 4.78\,(\log f_c)^2 - 18.733\,(\log f_c) - 40.98$$

The Okumura/Hata model is considered to be among the best in terms of accuracy in path loss prediction and provides a practical means of estimating path loss in a wide variety of situations [FREE07, RAPP02].

Example 5.4 Let $f_c = 900$ MHz, $h_t = 40$ m, $h_r = 5$ m, and $d = 10$ km. Estimate the path loss for a medium-sized city.

$$\begin{aligned} A(h_r) &= (1.1 \log 900 - 0.7)\,5 - (1.56 \log 900 - 0.8) \text{ dB} \\ &= 12.75 - 3.8 = 8.95 \text{ dB} \\ L_{\text{dB}} &= 69.55 + 26.16 \log 900 - 13.82 \log 40 - 8.95 + (44.9 - 6.55 \log 40) \log 10 \\ &= 69.55 + 77.28 - 22.14 - 8.95 + 34.4 = 150.14 \text{ dB} \end{aligned}$$

Noise

For any data transmission event, the received signal will consist of the transmitted signal, modified by the various distortions imposed by the transmission system, plus additional unwanted signals that are inserted somewhere between transmission and reception. These unwanted signals are referred to as **noise**. Noise is the major limiting factor in communications system performance.

Noise may be divided into four categories:

- **Thermal noise** is due to thermal agitation of electrons. It is present in all electronic devices and transmission media and is a function of temperature. Thermal noise is uniformly distributed across the frequency spectrum and hence is often referred to as ***white noise***. Thermal noise cannot be eliminated and therefore places an upper bound on communications system performance.
- When signals at different frequencies share the same transmission medium, the result may be **intermodulation noise**. Intermodulation noise produces signals at a frequency that is the sum or difference of the two original frequencies or multiples of those frequencies.
- **Crosstalk** has been experienced by anyone who, while using the telephone, has been able to hear another conversation; it is an unwanted coupling between signal paths.
- **Impulse noise**, however, is unpredictable and noncontinuous, consisting of irregular pulses or noise spikes of short duration and of relatively high

amplitude. It is generated from a variety of causes, including external electromagnetic disturbances, such as lightning, and faults and flaws in the communications system. Impulse noise is the primary source of error in digital data transmission. For example, a sharp spike of energy of 0.01 s duration would barely be noticed for voice conversation but would wash out about 10,000 bits of data being transmitted at 1 Mbps.

The Expression E_b/N_0

Chapter 2 introduced the **signal-to-noise ratio (SNR)**. There is a parameter related to SNR that is more convenient for determining digital data rates and error rates and that is the standard quality measure for digital communication system performance. The parameter is the ratio of signal *energy* per bit to noise power density per Hertz, E_b/N_0. The ratio E_b/N_0 is important because the bit error rate (BER) for digital data is a (decreasing) function of this ratio. Figure 5.4 illustrates the typical shape of a plot of BER versus E_b/N_0. Such plots are commonly found in the literature and several examples appear in this text. For any particular curve, as the signal strength relative to the noise increases (increasing E_b/N_0), the BER performance at the receiver decreases.

This makes intuitive sense. However, there is not a single unique curve that expresses the dependence of BER on E_b/N_0. Instead the performance of a transmission/reception system, in terms of BER versus E_b/N_0, also depends on the way in which the data is encoded onto the signal. Thus, Figure 5.4 shows two curves, one of which gives better performance than the other. A curve below and to the left of another curve defines superior performance. At the same BER for two signals, the

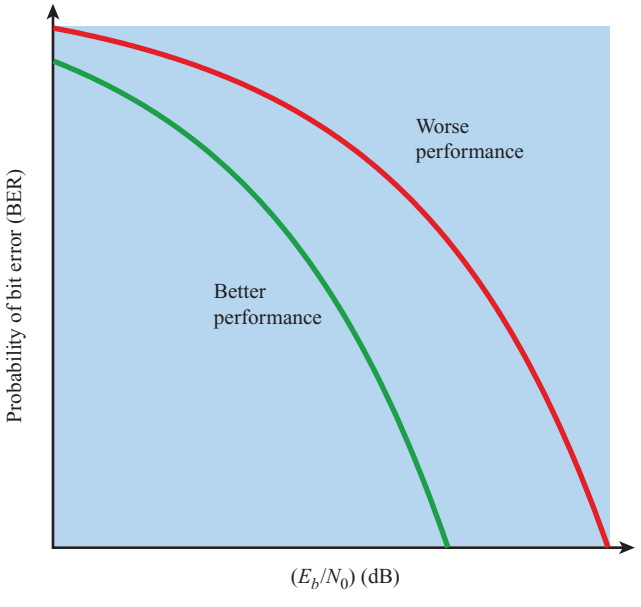


Figure 5.4 General Shape of BER vs E_b/N_0 Curves

curve to the left uses less E_b/N_0 to achieve that BER. For two signals using the same E_b/N_0, the curve below achieves a better BER. Chapter 7 explores the relationship of signal encoding with performance. A more detailed discussion of E_b/N_0 can be found in [SKLA01].

5.3 FADING IN THE MOBILE ENVIRONMENT

Perhaps the most challenging technical problem facing communications systems engineers is **fading** in a mobile environment. The term *fading* refers to the time variation of received signal power caused by changes in the transmission medium or path(s). In a fixed environment, fading is affected by changes in atmospheric conditions, such as rainfall. But in a mobile environment, where one of the two antennas is moving relative to the other, the relative location of various obstacles changes over time, creating complex transmission effects. Sometimes these variations can be quite rapid.

Multipath Propagation

One of the key effects causing fading is multipath propagation. For wireless facilities where there is a relatively free choice of where antennas are to be located, they can be placed so that if there are no nearby interfering obstacles, there is a direct line-of-sight path from transmitter to receiver. This is generally the case for many satellite facilities and for point-to-point microwave. In other cases, such as mobile telephony, there are obstacles in abundance. The signal can be reflected by such obstacles so that multiple copies of the signal with varying delays can be received. In fact, in extreme cases, the receiver might capture only reflected signals and not the direct signal. Depending on the differences in the path lengths of the direct and reflected waves, the composite signal can be either larger or smaller than the direct signal. Reinforcement and cancellation of the signal can occur, resulting from copies of the signal added together following multiple paths.

Three propagation mechanisms, illustrated in Figure 5.5, play a role. **Reflection** occurs when an electromagnetic signal encounters a surface that is large relative to the wavelength of the signal. For example, suppose a ground-reflected wave near the mobile unit is received. The ground-reflected wave and the line-of-sight (LOS) wave may tend to cancel, resulting in high signal loss. Further, because the mobile antenna is lower than most human-made structures in the area, multipath interference occurs. These reflected waves may interfere constructively or destructively at the receiver.

Diffraction occurs at the edge of an impenetrable body that is large compared to the wavelength of the radio wave. When a radio wave encounters such an edge, waves propagate in different directions with the edge as the source. Thus, signals can be received even when there is no unobstructed LOS from the transmitter.

If the size of an obstacle is on the order of the wavelength of the signal or less, **scattering** occurs. An incoming signal is scattered into several weaker outgoing signals. At typical cellular microwave frequencies, there are numerous objects, such as lamp posts and traffic signs, that can cause scattering. Thus, scattering effects are difficult to predict.

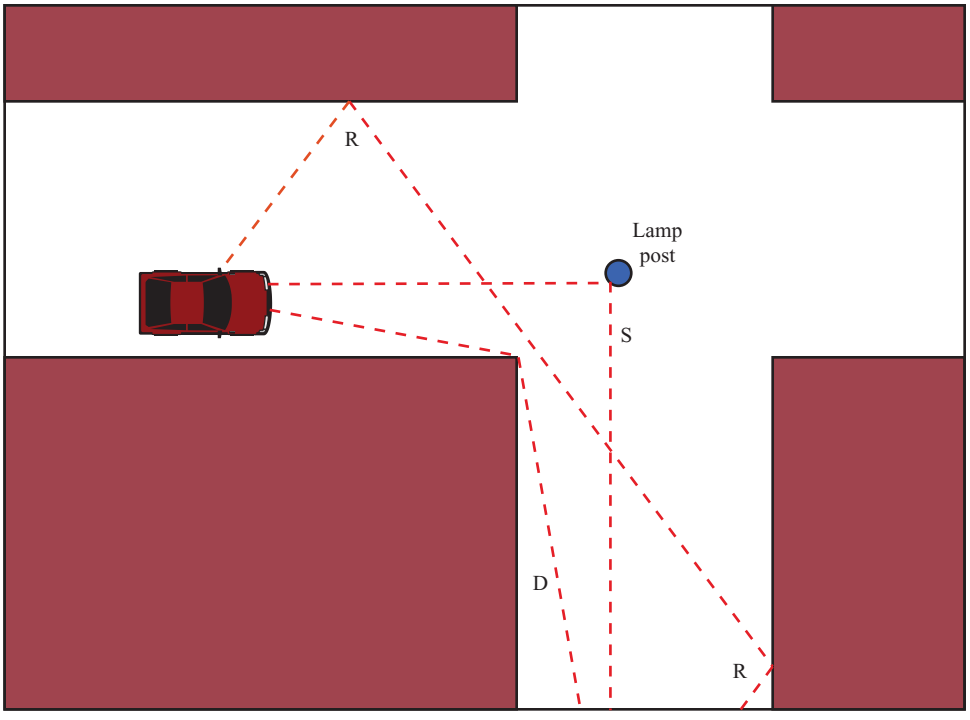


Figure 5.5 Sketch of Three Important Propagation Mechanisms: Reflection (R), Scattering (S), and Diffraction (D)

These three propagation effects influence system performance in various ways depending on local conditions and as the mobile unit moves within a cell. If a mobile unit has a clear LOS to the transmitter, then diffraction and scattering are generally minor effects, although reflection may have a significant impact. If there is no clear LOS, such as in an urban area at street level, then diffraction and scattering are the primary means of signal reception.

The Effects of Multipath Propagation As multipath signals add together, the resulting signal power can be stronger, but can also be lower by a factor of 100 or 1000 (20 or 30 dB). The signal level relative to noise declines, making signal detection at the receiver more difficult.

A second phenomenon, of particular importance for digital transmission, is intersymbol interference (ISI). Consider that we are sending a narrow pulse at a given frequency across a link between a fixed antenna and a mobile unit. Figure 5.6 shows what the channel may deliver to the receiver if the impulse is sent at two different times. The upper line shows two pulses at the time of transmission. The lower line shows the resulting pulses at the receiver. In each case the first received pulse is the desired LOS signal. The magnitude of that pulse may change because of changes in atmospheric attenuation. Further, as the mobile unit moves farther away from the fixed antenna, the amount of LOS attenuation increases. But in addition to this

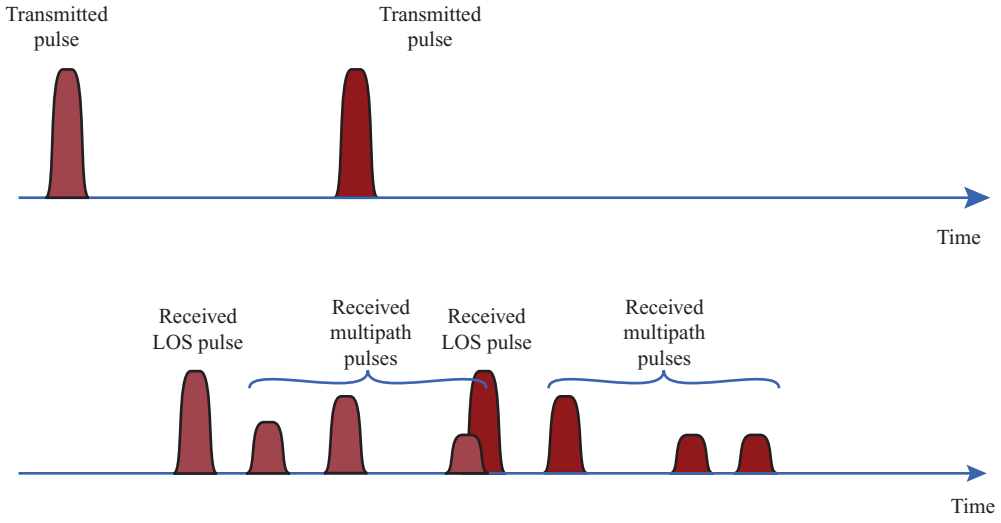


Figure 5.6 Two Pulses in Time Variant Multipath

primary pulse, there may be multiple secondary pulses due to reflection, diffraction, and scattering. Now suppose that this pulse encodes one or more bits of data. In that case, one or more delayed copies of a pulse may arrive at the same time as the primary pulse for a subsequent bit. These delayed pulses act as a form of noise to the subsequent primary pulse, making recovery of the bit information more difficult.

As the mobile antenna moves, the location of various obstacles changes; hence the number, magnitude, and timing of the secondary pulses change. This makes it difficult to design signal processing techniques that will filter out multipath effects so that the intended signal is recovered with fidelity.

Types of Fading Fading effects in a mobile environment can be classified as either small-scale or large-scale. Referring to Figure 5.5, as the mobile unit moves down a street in an urban environment, as the mobile user covers distances well in excess of a wavelength, the urban environment changes as the user passes buildings of different heights, vacant lots, intersections, and so forth. Over these longer distances, there is a change in the average received power. This change is mainly caused by shadowing and differences in distance from the transmitter. This is indicated by the slowly changing waveform in Figure 5.7 and is referred to as **large-scale fading**.

However, rapid variations in signal strength also occur over distances of about one-half a wavelength. At a frequency of 900 MHz, which is typical for mobile cellular applications, a wavelength is 0.33 m. The rapidly changing waveform in Figure 5.7 shows an example of the spatial variation of received signal amplitude at 900 MHz in an urban setting. Note that changes of amplitude can be as much as 20 or 30 dB over a short distance. This type of rapidly changing fading phenomenon, known as **small-scale fading**, affects not only mobile phones in automobiles, but even a mobile phone user walking down an urban street.

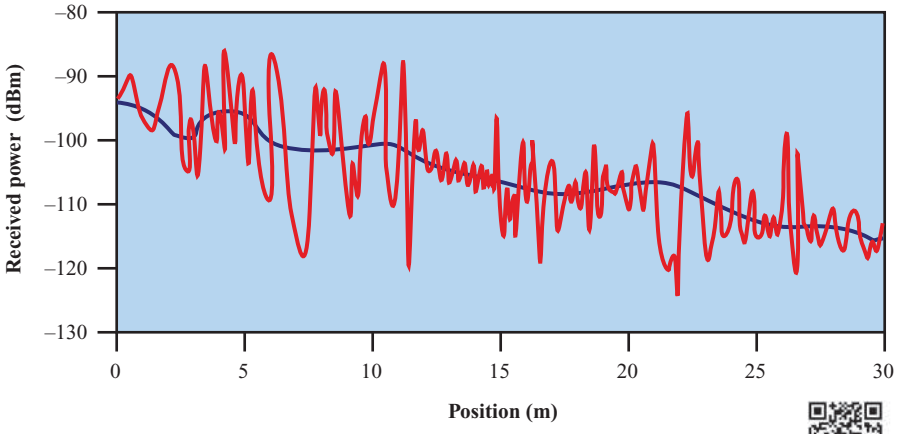


Figure 5.7 Typical Slow and Fast Fading in an Urban Mobile Environment

There are two distinct types of small-scale fading effects.

- **Doppler spread** causes signal performance to change with time due to the movement of mobiles and obstacles.
- **Multipath fading** causes the signal to vary with location due to the combination of delayed multipath signal arrivals.

Regarding Doppler spread, a channel may change over a very short time span. This is characterized by the channel's **coherence time**, T_c, which is the time over which the channel stays relatively constant. Coherence times for a pedestrian might be 70 ms, whereas times might be 5 ms for a vehicle moving at highway speeds.[2] This might have a significant effect on a signal, depending on its bit rate, r_b bits/s. This signal would have a bit time $T_b = 1/r_b$ s/bit. If the coherence time T_c is much, much longer than the bit time T_b, then the channel could be called **slow fading**. The channel changes very slowly during the time to transmit each bit. If, however, this is not true, the channel is undergoing **fast fading**. Therefore, for our purposes in this book we consider a channel to be fast fading if the coherence time T_c is less than, approximately equal, or even slightly greater than the bit time T_b, since in all cases the coherence time is not much, much greater than the bit time.

The other small-scale effect, multipath fading, can cause distortion and intersymbol interference. **Flat fading** is that type of fading in which all frequency components of the received signal fluctuate in the same proportions simultaneously. Multipath fading can be characterized by a **coherence bandwidth**, B_C, which is the range of frequencies over which the channel response is relatively constant. Therefore, if the coherence bandwidth is much, much greater than the signal bandwidth, then flat fading occurs. If a signal bandwidth can be approximated as $B_S \approx r_b$, then B_C must be much, much greater than B_S. In contrast, **frequency selective fading** occurs when flat fading is not present. It affects unequally the different spectral

[2]A common formula is $T_c = 0.423c/vf$, where c is the speed of light, v is the velocity of movement, and f is the frequency [RAPP02].

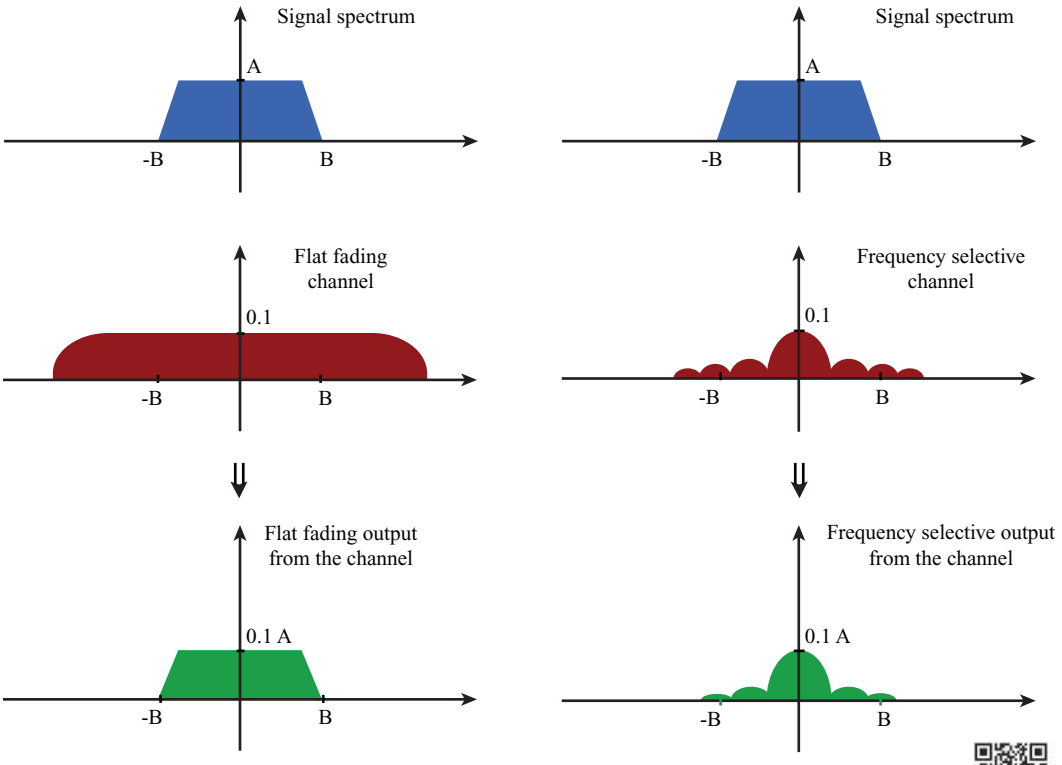


Figure 5.8 Flat and Frequency Selective Fading

components of a radio signal. If attenuation occurs over only a portion of the bandwidth of the signal the fading is considered to be frequency selective. Figure 5.8 illustrates a flat fading channel versus a frequency selective channel relative to the bandwidth of a signal.

These characterizations for Doppler spread and multipath fading do not depend on each other. Therefore, four combinations can occur: fast-flat, slow-flat, fast-frequency selective, and slow-frequency selective fading.

Example 5.5 Suppose that a pedestrian is moving through an urban environment that has a wireless channel with a coherence time of 70 ms and a coherence bandwidth of 150 kHz. The bit rate of the signal being used is 100 kbps.

a. How would the channel be characterized regarding Doppler spread and multipath fading?

To check for slow fading, test the following, using a factor of 10 for much, much greater.

$$T_b = 1/r_S = 10\ \mu s$$

$$T_C \gg T_b?$$

$$T_C > 10T_b?$$

$$\textit{Test condition}: 70\ ms > 100\ \mu s?$$

This is true, so *slow fading*.
To check for flat fading, test the following.

$$\text{Assume } B_S \approx r_S = 100\,kHz$$
$$B_C \gg B_S?$$
$$B_C > 10B_S?$$
$$\textit{Test condition}: 150\,kHz > 1\,Mbps?$$

This is not true, so *frequency selective fading*.
This channel is slow and frequency selective.

b. What range of bit rates can be supported to have flat fading?

This is the requirement

$$B_C \gg B_S$$
$$B_C > 10B_S$$
$$150\,kHz > 10B_S$$
$$B_S < 15\,kHz$$
$$r_b < 15 \text{ kbps}$$

5.4 CHANNEL CORRECTION MECHANISMS

The efforts to compensate for the errors and distortions introduced by multipath fading fall into four general categories: forward error correction, adaptive equalization, adaptive modulation and coding, and diversity techniques with multiple-input multiple-output (MIMO). In the typical mobile wireless environment, techniques from all three categories are combined to combat the error rates encountered.

Forward Error Correction

Forward error correction is applicable in digital transmission applications: those in which the transmitted signal carries digital data or digitized voice or video data. The term *forward* refers to procedures whereby a receiver, using only information contained in the incoming digital transmission, corrects bit errors in the data. This is in contrast to backward error correction, in which the receiver merely detects the presence of errors and then sends a request back to the transmitter to retransmit the data in error. Backward error correction is not practical in many wireless applications. For example, in satellite communications, the amount of delay involved makes retransmission undesirable. In mobile communications, the error rates are often so high that there is a high probability that the retransmitted block of bits will also contain errors. In these applications, forward error correction is required. In essence, forward error correction is achieved as follows:

1. Using a coding algorithm, the transmitter adds a number of additional, redundant bits to each transmitted block of data. These bits form an ***error-correcting code*** and are calculated as a function of the data bits.

2. For each incoming block of bits (data plus error-correcting code), the receiver calculates a new error-correcting code from the incoming data bits. If the calculated code matches the incoming code, then the receiver assumes that no error has occurred in this block of bits.
3. If the incoming and calculated codes do not match, then one or more bits are in error. If the number of bit errors is below a threshold that depends on the length of the code and the nature of the algorithm, it is possible for the receiver to determine the bit positions in error and correct all errors.

Typically in mobile wireless applications, the ratio of total bits sent to data bits sent is between 2 and 3. This may seem an extravagant amount of overhead, in that the capacity of the system is cut to one-half or one-third of its potential, but the mobile wireless environment is such a challenging medium that such levels of redundancy are necessary.

Section 5.6 and Chapter 10 examine forward error correction techniques in more detail.

Adaptive Equalization

Adaptive equalization can be applied to transmissions that carry analog information (e.g., analog voice or video) or digital information (e.g., digital data, digitized voice or video) and is used to combat intersymbol interference. The process of equalization involves some method of gathering the dispersed symbol energy back together into its original time interval.

Diversity Techniques and MIMO

Diversity is based on the fact that individual channels experience independent fading events. For example, multiple antennas that are spaced far enough apart will have independent fading. We can therefore compensate for error effects by providing multiple logical channels in some sense between transmitter and receiver and sending part of the signal over each channel. This technique does not eliminate errors but it does reduce the error rate, since we have spread the transmission out to avoid being subjected to the highest error rate that might occur. The other techniques (equalization, forward error correction) can then cope with the reduced error rate.

Some diversity techniques involve the physical transmission path and are referred to as ***space diversity***. For example, multiple nearby antennas, if spaced far enough apart, may be used to receive the message with the signals combined in some fashion to reconstruct the most likely transmitted signal. Another example is the use of collocated multiple directional antennas, each oriented to a different reception angle with the incoming signals again combined to reconstitute the transmitted signal.

With ***frequency diversity***, the signal is spread out over a larger frequency bandwidth or carried on multiple frequency carriers. The most important examples of this approach are orthogonal frequency division multiplexing (OFDM) and spread spectrum.

Time diversity techniques aim to spread the data out over time so that a noise burst affects fewer bits. This can be accomplished with interleaving or through a Rake receiver.

When these multiple signals are received, there are two basic ways they can be used:

1. **Selection diversity:** Choose one signal that is acceptable or the best.
2. **Diversity combining:** Combine the best signal with the other signals. Adjust the gain and phase so they add together to improve the overall output signal.

Example 5.6. Suppose a wireless channel has two possible quality levels. It has an 80% probability of having a bit error rate of 10^{-6}, but a 20% probability of having a bit error rate of 0.1. Assume independently varying signals can be received through two antennas, and the system uses selection diversity to choose the best signal. How does the overall performance improve?

For one signal, the performance is

$$P_b = Pr\{poor\} * (P_b \, for \, poor) + Pr\{good\} * (P_b \, for \, good)$$
$$P_b = 0.2(0.1) + 0.8(10^{-6}) \approx 0.02$$

For two diversity branches, the only case of poor performance would occur if both branches would be poor so no good signal could be found. The probability of both being poor is 0.2^2, so

$$P_b = 0.2^2(0.1) + (1 - 0.2^2)(10^{-6}) \approx 0.004$$

For k signals, $P_b \approx 0.2^k(0.1)$. This means that P_b drops one order of magnitude for each additional diversity branch.

Multiple-Input Multiple-Output (MIMO) Antennas If a transmitter and receiver implement a system with multiple antennas, this is called a **multiple-input multiple-output (MIMO)** system. These allow several of the mechanisms discussed in this chapter to be implemented as illustrated in Figure 5.9.

1. **Diversity:** Diversity can be accomplished to have multiple received signals through multiple transmit and/or receive antennas.
2. **Beam-forming:** Multiple antennas can be configured to create directional antenna patterns to focus and increase energy to intended recipients.
3. **Multi-user MIMO (MU-MIMO):** With enough MIMO antennas, directional antenna beams can be established to multiple users simultaneously.
4. **Multilayer transmission:** Multiple, parallel data streams can flow between a pair of transmit and receive antennas.

Modern systems implement up to 4 × 4 (4 input, 4 output) and 8 × 8 MIMO configurations. Antenna systems have been approved in specifications for as many as 8 per antenna array, and two-dimensional arrays of 64 antennas or more are being envisioned for future technologies.

The MIMO antenna architecture has become a key technology in evolving high-speed wireless networks, including IEEE 802.11 Wi-Fi LANs and Long Term Evolution (LTE) fourth-generation cellular. Together, MIMO and OFDM technologies are the cornerstone of emerging broadband wireless networks.

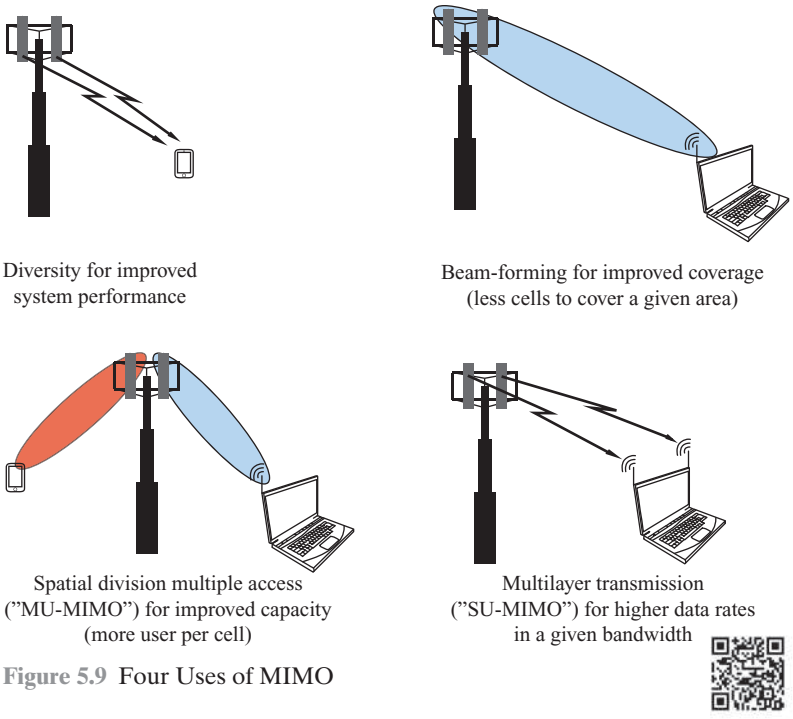


Figure 5.9 Four Uses of MIMO

Spread Spectrum and OFDM

Traditional communications, wireline or wireless, simply modulate a baseband signal up to a required transmission channel and frequency. No change to the original signal occurs. Two methods, however, have been used to overcome wireless channel impairments; the signals are significantly modified for transmission.

- **Orthogonal Frequency Division Multiplexing (OFDM)** splits a signal into many lower bit rate streams that are transmitted over carefully spaced frequencies. This can overcome frequency selective fading by using significantly lower bandwidth per stream with longer bit times. Each of these frequencies can then be amplified separately. This is briefly discussed later in this chapter, and Chapter 8 provides a thorough examination.
- **Spread spectrum** makes a signal use 100 times or more wider bandwidth, with lower energy density at each frequency. This can overcome frequency selective situations; even if some frequencies are poor, good overall average performance is achieved. This is examined briefly later in this chapter and in Chapter 9.

The remainder of the chapter provides an overview of signal encoding and error control techniques. Then it introduces more information about OFDM and spread spectrum.

Adaptive Modulation and Coding

Since the characteristics of a wireless channel can change 100s of times per second due to fading (e.g., 200 times/s for a 5 ms coherence time), modern systems use adaptive modulation and coding (AMC) to adjust their schemes just as quickly. Modulation and coding are discussed more in this chapter in Sections 5.5 and 5.6, in

more depth in Chapters 7 and 10. They essentially create signals that send as much information as possible for a given received signal strength and noise, then they detect and correct the errors. To adapt 100's of times per second, two features must be present in the protocols for a system.

1. Mechanisms to measure the quality of the wireless channel. These might include monitoring packet loss rates or sending special pilot signals expressly for measurement purposes.
2. Messaging mechanisms to communicate the signal quality indicators between transmitters and receivers, and also to communicate the new modulation and coding formats.

Bandwidth Expansion

All of the above correction mechanisms seek to increase the efficient use of the bandwidth of a channel, commonly measured in an efficiency of bps/Hz. But according to Shannon's theory there is a limit to this efficiency for a given signal to noise ratio. If throughput requirements are beyond what can be achieved in a given bandwidth, a series of bandwidth expansion approaches are used.

- **Carrier aggregation** combines multiple channels. For example, 802.11n and 802.11ac combine the 20 MHz channels from earlier 802.11 standards into 40, 80, or 160 MHz channels.
- **Frequency reuse** allows the same carrier frequencies to be reused when devices are sufficiently far enough away so the signal-to-interference ratio is low enough. This has traditionally been provided by breaking a cellular coverage area into large cells, called ***macro cells***, of several kilometers in diameter. Cells far enough away can reuse the frequencies. But now **small cells** with limited power and range are being used for the same frequency reuse objectives. Indoor small cells are commonly called **femtocells** and outdoor cells are provided by ***relays*** or **picocells**. These are discussed in conjunction with LTE in Chapter 14. This approach is called ***network densification*** because it allows frequencies to be reused many times.
- **Millimeter wave** (**mmWave**) bands are higher frequencies in the 30 GHz to 300 GHz bands that have more bandwidth available in wider bandwidth channels. Recall that $\lambda = c/f$, so 30 to 300 GHz has wavelengths of 10 to 1 mm. This is an example of using different carrier frequencies to achieve higher bandwidth, given spectrum regulations. mmWave bands are more difficult to use, however, since they are more susceptible to attenuation by obstructions and **atmospheric absorption**. IEEE 802.11ad uses mmWave bands within a single room. Future technologies, however, may use them for wider range communication, in conjunction with higher gain MIMO configurations.

5.5 DIGITAL SIGNAL ENCODING TECHNIQUES

A variety of methods are used to encode analog and digital data onto analog and digital signals. Many of these techniques are examined in Chapter 7, which is dedicated to signal encoding topics. Here, we only discuss the encoding of digital data onto analog signals, since most of today's wireless communication is the transmission of digital data.

The basis for analog signals is a continuous constant-frequency signal known as the carrier signal. This is represented by a sinusoidal function as follows.

$$s(t) = A\cos(2\pi f_c t + \theta) \tag{5.7}$$

This signal has an amplitude, A, a frequency, f, and a phase, θ. The frequency, f_c, that is used here is called the carrier frequency. This is chosen to be compatible with the transmission medium being used. In the case of wireless communication, frequencies and signal powers must also be used as specified by regulatory agencies.

Data is transmitted using a carrier signal by modulation. Modulation is the process of encoding source data onto the carrier signal. All modulation techniques involve sending information by changing one or more of the three fundamental frequency domain parameters: amplitude, frequency, and phase. Accordingly, there are three basic encoding or modulation techniques for transforming digital data into analog signals, as illustrated in Figure 5.10: amplitude-shift keying (ASK),

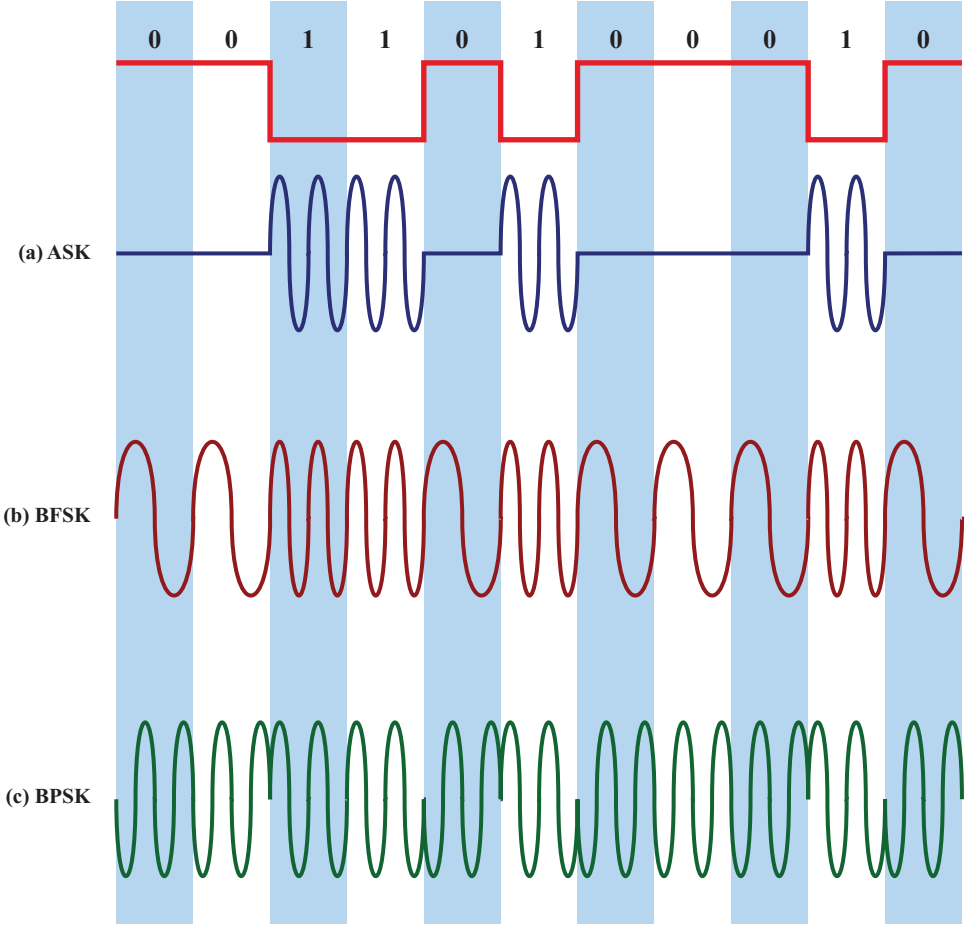


Figure 5.10 Modulation of Analog Signals for Digital Data

frequency-shift keying (FSK), and phase-shift keying (PSK). In all these cases, the resulting signal occupies a bandwidth centered on the carrier frequency.

- In **amplitude-shift keying (ASK)** the two binary values are represented by two different amplitudes of the carrier frequency. Commonly, one of the amplitudes is zero; that is, one binary digit is represented by the presence, at constant amplitude, of the carrier, the other by the absence of the carrier (Figure 5.10a).

$$\textbf{ASK} \qquad s(t) = \begin{cases} A\cos(2\pi f_c t) & \text{binary 1} \\ 0 & \text{binary 0} \end{cases} \tag{5.8}$$

- The most common form of **frequency-shift keying (FSK)** is binary FSK (BFSK), in which the two binary values are represented by two different frequencies near the carrier frequency (Figure 5.10b).

$$\textbf{BFSK} \qquad \text{B}s(t) = \begin{cases} A\cos(2\pi f_1 t) & \text{binary 1} \\ A\cos(2\pi f_2 t) & \text{binary 0} \end{cases} \tag{5.9}$$

- In **phase-shift keying (PSK)**, the phase of the carrier signal is shifted to represent data. The simplest scheme uses two phases to represent the two binary digits (Figure 5.10c) and is known as binary phase-shift keying.

$$\textbf{BPSK} \qquad s(t) = \begin{cases} A\cos(2\pi f_c t) \\ A\cos(2\pi f_c t + \pi) \end{cases} = \begin{cases} A\cos(2\pi f_c t) & \text{binary 1} \\ A\cos(2\pi f_c t) & \text{binary 0} \end{cases} \tag{5.10}$$

With two values of amplitude, frequency, or phase, one bit of information can be transmitted at a time. If, for example, four frequencies where used for FSK (which would then be called multilevel FSK or MFSK), two bits of information could be transmitted at a time. Each frequency could correspond to a two-bit sequence. This would effectively double the bit rate of the information transfer. If a scheme used M levels (M would always be a power of 2), the bit rate would increase by a factor of $L = \log_2(M)$ bits. If the same amount of transmitted power were still used, however, the bit error rate of the signal generally would also increase, creating a tradeoff between increased bit rates but also increased error rates.

Example 5.7 With $f_c = 250$ kHz and $M = 8$ ($L = 3$ bits), we can have the following frequency assignments for each of the 8 possible 3-bit data combinations if the frequencies are spaced apart by 50 kHz.

$f_1 = 75$ kHz 000 $f_2 = 125$ kHz 001 $f_3 = 175$ kHz 010 $f_4 = 225$ kHz 011
$f_5 = 275$ kHz 100 $f_6 = 325$ kHz 101 $f_7 = 375$ kHz 110 $f_8 = 425$ kHz 111

When using a multilevel scheme, more than one of the signal characteristics can be changed. For example, 16-level Quadrature Amplitude Modulation (16QAM) uses various combinations of amplitudes and phases to create 16 different combinations. This would transmit four bits at a time. Figure 5.11 illustrates 16QAM in what is known as a constellation diagram. The amplitude of the signal will be the distance from the origin. For example, symbol 1111 would be transmitted with an amplitude of $\sqrt{1^2 + 1^2} = \sqrt{2}$. The phase of the signal would be the angle of the point in the

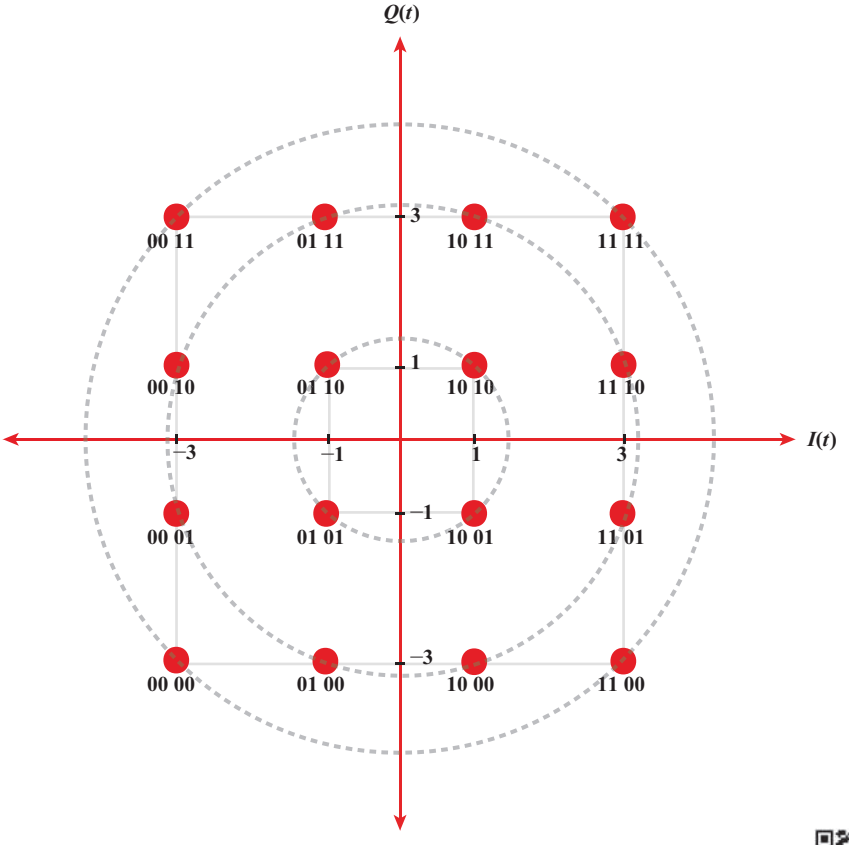
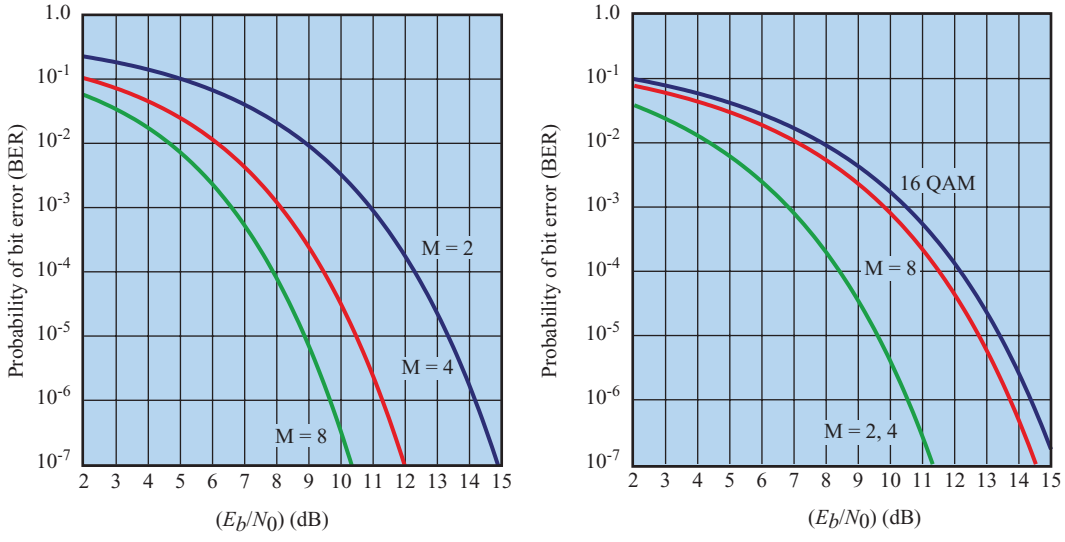


Figure 5.11 16QAM Constellation Diagram

constellation, which for 1111 would be $\tan^{-1}(1/1) = 45°$. By looking carefully at the constellation, we can see that there would be three different possible amplitudes, and 12 different possible phases. The frequencies are the same every time. Some modulation schemes used in communication systems involve 64QAM or 256QAM.

Figure 5.12 shows the received E_b/N_0 versus bit error rate curves for multilevel FSK in Figure 5.12a and QPSK (called quadrature phase-shift keying, which is really a version of 4QAM), 16QAM, and 64QAM in Figure 5.12b. Even though 64QAM will allow more data to be packed into each symbol, and hence a higher data rate, the bit error rate is worse for the same E_b/N_0. Note again that this is the *received* E_b/N_0, so if the wireless channel is good, then the E_b/N_0 will be higher and a constellation like 64QAM might be used to achieve a high data rate. If, however, the received signal strength is low, then E_b/N_0 would be too low and the 64QAM BER would be unacceptable. Only 16QAM or QPSK might be possible. It is very useful, therefore, to have **adaptive modulation and coding (AMC)**, because we know that received signal strength can vary greatly from one coherence time to the next. The system could monitor channel conditions and use QPSK for a period of time then switch to 64QAM later.



(a) Multilevel FSK (MFSK)

(b) Multilevel PSK (MPSK) and 16QAM

Figure 5.12 Theoretical Bit Error Rate for Multilevel FSK, PSK, and QAM

Chapter 7 provides more details on these schemes and illustrates error rate performance. It also discusses modulation of analog data and the encoding of voice signals.

5.6 CODING AND ERROR CONTROL

In earlier sections, we talked about transmission impairments and the effect of data rate and signal-to-noise ratio on bit error rate. Regardless of the design of the transmission system, there will be errors, resulting in the change of one or more bits in a transmitted frame.

Three approaches are in common use for coping with data transmission errors:

- Error detection codes
- Error correction codes, also called **forward error correction (FEC)** codes
- Automatic repeat request (ARQ) protocols

An **error detection** code simply detects the presence of an error. Typically, such codes are used in conjunction with a protocol at the data link or transport level that uses an ARQ scheme. With an ARQ scheme, a receiver discards a block of data in which an error is detected and the transmitter retransmits that block of data. FEC codes are designed not just to detect but correct errors, avoiding the need for retransmission. FEC schemes are frequently used in wireless transmission, where retransmission schemes are highly inefficient and error rates may be high. Some wireless protocols use Hybrid ARQ, which is a combination of FEC and ARQ.

Error Detection

In what follows, we assume that data are transmitted as one or more contiguous sequences of bits, called *frames*. Let us define these probabilities with respect to errors in transmitted frames:

- P_b: Probability of a single bit error; also known as the bit error rate (BER)
- P_1: Probability that a frame arrives with no bit errors
- P_2: Probability that, with an error detection algorithm in use, a frame arrives with one or more undetected errors
- P_3: Probability that, with an error detection algorithm in use, a frame arrives with one or more detected bit errors but no undetected bit errors

First consider the case when no means are taken to detect errors. Then the probability of detected errors (P_3) is zero. To express the remaining probabilities, assume the probability that any bit is in error (P_b) is constant and independent for each bit. Then, we have

$$P_1 = (1 - P_b)^F$$
$$P_2 = 1 - P_1$$

where F is the number of bits per frame. In words, the probability that a frame arrives with no bit errors decreases when the probability of a single bit error increases, as you would expect. Also, the probability that a frame arrives with no bit errors decreases with increasing frame length; the longer the frame, the more bits it has and the higher the probability that one of these is in error.

Example 5.8 A system has a defined objective for connections that the BER should be less than 10^{-6} on at least 90% of observed 1-minute intervals. Suppose now that we have the rather modest user requirement that on average one frame with an undetected bit error should occur per day on a continuously used 1 Mbps channel, and let us assume a frame length of 1000 bits. The number of frames that can be transmitted in a day comes out to 8.64×10^7, which yields a required frame error rate of $P_2 = 1/(8.64 \times 10^7) = 1.16 \times 10^{-8}$. But if we assume a value of P_b of 10^{-6}, then $P_1 = (0.999999)^{1000} = 0.999$ and therefore $P_2 = 10^{-3}$, which is about five orders of magnitude too large to meet our requirement. This means that $(8.64 \times 10^7)^* P_2 = 86{,}400$ frames with undetected bit errors would occur per day for P_b of 10^{-6}.

This is the kind of result that motivates the use of error detection techniques. All of these techniques operate on the following principle (Figure 5.13). For a given frame of bits, the transmitter adds additional bits that constitute an error-detecting code. This code is calculated as a function of the other transmitted bits. Typically, for a data block of k bits, the error detection algorithm yields an error detection code of $n - k$ bits, where $(n - k) < k$. The error detection code, also referred to as the ***check bits***, is appended to the data block to produce a frame of n bits, which is then transmitted. The receiver separates the incoming frame into the k bits of data and $(n - k)$ bits of the error detection code. The receiver performs the same error detection calculation on the data bits and compares this value with the value of the incoming error detection code. A detected error occurs if and only if there is a mismatch. Thus, P_3 is the probability that a frame contains errors and that the error

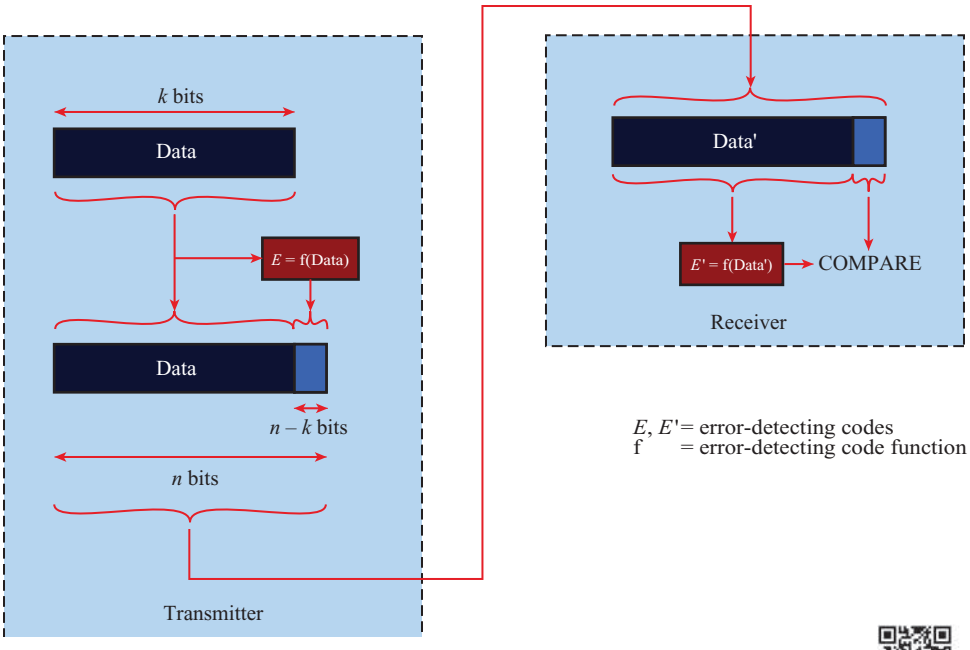


Figure 5.13 Error Detection Process

detection scheme will detect that fact. P_2 is known as the *residual error rate* and is the probability that an error will be undetected despite the use of an error detection scheme.

Parity Check The simplest error detection scheme is to append a parity bit to the end of a block of data. A typical example is character transmission, in which a parity bit is attached to each 7-bit character. The value of this bit is selected so that the character has an even number of 1s (even parity) or an odd number of 1s (odd parity).

Example 5.9 If the transmitter is transmitting 1110001 and using odd parity, it will append a 1 and transmit 11110001. The receiver examines the received character and, if the total number of 1s is odd, assumes that no error has occurred. If one bit (or any odd number of bits) is erroneously inverted during transmission (e.g., 11100001), then the receiver will detect an error.

Note, however, that if two (or any even number) of bits are inverted due to error, an undetected error occurs. Typically, even parity is used for synchronous transmission and odd parity for asynchronous transmission.

The use of the parity bit is not foolproof, as noise impulses are often long enough to destroy more than one bit, especially at high data rates.

Cyclic Redundancy Check One of the most common, and one of the most powerful, error-detecting codes is the **cyclic redundancy check (CRC)**, which can be described as follows. Given a k-bit block of bits, or message, the transmitter generates an $(n - k)$-bit sequence, known as a frame check sequence (FCS), such that the resulting

frame, consisting of n bits, is exactly divisible by some predetermined number. The receiver then divides the incoming frame by that number and, if there is no remainder, assumes there was no error.[3]

To clarify this, we present the procedure in three ways: modulo 2 arithmetic, polynomials, and digital logic.

Modulo 2 Arithmetic Modulo 2 arithmetic uses binary addition with no carries, which is just the exclusive-OR (XOR) operation. Binary subtraction with no carries is also interpreted as the XOR operation: For example:

$$\begin{array}{rrr} 1111 & 1111 & 11001 \\ +\ 1010 & -0101 & \times\ \ 11 \\ \hline 0101 & 1010 & 11001 \\ & & 11001 \\ \hline & & 101011 \end{array}$$

Now define:

T = n-bit frame to be transmitted

D = k-bit block of data, or message, the first k bits of T

F = $(n - k)$-bit FCS, the last $(n - k)$ bits of T

P = pattern of $n - k + 1$ bits; this is the predetermined divisor

We would like T/P to have no remainder. It should be clear that

$$T = 2^{n-k}D + F$$

That is, by multiplying D by 2^{n-k}, we have in effect shifted it to the left by $n - k$ bits and padded out the result with zeroes. Adding F yields the concatenation of D and F, which is T. We want T to be exactly divisible by P. Suppose that we divide $2^{n-k}D$ by P:

$$\frac{2^{n-k}D}{P} = Q + \frac{R}{P} \tag{5.11}$$

There is a quotient and a remainder. Because division is modulo 2, the remainder is always at least one bit shorter than the divisor. We will use this remainder as our FCS. Then,

$$T = 2^{n-k}D + R \tag{5.12}$$

Does this R satisfy our condition that T/P have no remainder? To see that it does, consider

$$\frac{T}{P} = \frac{2^{n-k}D + R}{P} = \frac{2^{n-k}D}{P} + \frac{R}{P}$$

Substituting Equation (5.11), we have

$$\frac{T}{R} = Q + \frac{R}{P} + \frac{R}{P}$$

[3]This procedure is slightly different from that of Figure 5.13. As shall be seen, the CRC process could be implemented as follows. The receiver could perform a division operation on the incoming k data bits and compare the result to the incoming (n − k) check bits.

However, any binary number added to itself modulo 2 yields zero. Thus,

$$\frac{T}{P} = Q + \frac{R + R}{P} = Q$$

There is no remainder, and therefore T is exactly divisible by P. Thus, the FCS is easily generated: Simply divide $2^{n-k}D$ by P and use the $(n - k)$-bit remainder as the FCS. On reception, the receiver will divide T by P and will get no remainder if there have been no errors.

Example 5.10

1. Given

$$\begin{aligned} \text{Message } D &= 1010001101 \text{ (10 bits)} \\ \text{Pattern } P &= 110101 \text{ (6 bits)} \\ \text{FCS } R &= \text{to be calculated (5 bits)} \end{aligned}$$

   Thus, $n = 15$, $k = 10$, and $(n-k) = 5$.
2. The message is multiplied by 2^5, yielding 101000110100000.
3. This product is divided by P:

```
                  1 1 0 1 0 1 0 1 1 0  ← Q
            ┌──────────────────────────
P → 1 1 0 1 0 1 │ 1 0 1 0 0 0 1 1 0 1 0 0 0 0 0  ← 2^(n-k)D
          1 1 0 1 0 1
          ─────────
            1 1 1 0 1 1
            1 1 0 1 0 1
            ─────────
              1 1 1 0 1 0
              1 1 0 1 0 1
              ─────────
                1 1 1 1 1 0
                1 1 0 1 0 1
                ─────────
                  1 0 1 1 0 0
                  1 1 0 1 0 1
                  ─────────
                    1 1 0 0 1 0
                    1 1 0 1 0 1
                    ─────────
                      0 1 1 1 0  ← R
```

4. The remainder is added to $2^5 D$ to give $T = 101000110101110$, which is transmitted.
5. If there are no errors, the receiver receives T intact. The received frame is divided by P:

```
                  1 1 0 1 0 1 0 1 1 0  ← Q
            ┌──────────────────────────
P → 1 1 0 1 0 1 │ 1 0 1 0 0 0 1 1 0 1 0 1 1 1 0  ← T
          1 1 0 1 0 1
          ─────────
            1 1 1 0 1 1
            1 1 0 1 0 1
            ─────────
              1 1 1 0 1 0
              1 1 0 1 0 1
              ─────────
                1 1 1 1 1 0
                1 1 0 1 0 1
                ─────────
                  1 0 1 1 1 1
                  1 1 0 1 0 1
                  ─────────
                    1 1 0 1 0 1
                    1 1 0 1 0 1
                    ─────────
                              0  ← R
```

Because there is no remainder, it is assumed that there have been no errors.

The pattern P is chosen to be one bit longer than the desired FCS, and the exact bit pattern chosen depends on the type of errors expected. At minimum, both the high- and low-order bits of P must be 1.

There is a concise method for specifying the occurrence of one or more errors. An error results in the reversal of a bit. This is equivalent to taking the XOR of the bit and 1 (modulo 2 addition of 1 to the bit): $0 + 1 = 1; 1 + 1 = 0$. Thus, the errors in an n-bit frame can be represented by an n-bit field with 1s in each error position. The resulting frame T_r can be expressed as

$$T_r = T \oplus E$$

where

T = transmitted frame
E = error pattern with 1s in positions where errors occur
T_r = received frame

If there is an error ($E \neq 0$), the receiver will fail to detect the error if and only if T_r is divisible by P, which is equivalent to E divisible by P. Intuitively, this seems an unlikely occurrence.

Polynomials A second way of viewing the CRC process is to express all values as polynomials in a dummy variable X, with binary coefficients. The coefficients correspond to the bits in the binary number. Arithmetic operations are again modulo 2. The CRC process can now be described as

$$\frac{X^{n-k}D(X)}{P(X)} = Q(X) + \frac{R(X)}{P(X)}$$

$$T(X) = X^{n-k}D(X) + R(X)$$

Compare these equations with Equations (5.11) and (5.12).

An error $E(X)$ will only be undetectable if it is divisible by $P(X)$. It can be shown [PETE61, RAMA88] that all of the following errors are not divisible by a suitably chosen $P(X)$ and hence are detectable:

- All single-bit errors, if $P(X)$ has more than one nonzero term
- All double-bit errors, as long as $P(X)$ has a factor with at least three terms
- Any odd number of errors, as long as $P(X)$ contains a factor $(X + 1)$
- Any burst error[4] for which the length of the burst is less than or equal to $n - k$; that is, less than or equal to the length of the FCS
- A fraction of error bursts of length $n - k + 1$; the fraction equals $1 - 2^{-(n-k-1)}$
- A fraction of error bursts of length greater than $n - k + 1$; the fraction equals $1 - 2^{-(n-k)}$

In addition, it can be shown that if all error patterns are considered equally likely, then for a burst error of length $r + 1$, the probability of an undetected error [i.e., $E(X)$ is divisible by $P(X)$] is $1/2^{r-1}$ and for a longer burst, the probability is $1/2^r$,

[4]A burst error of length B is a contiguous sequence of B bits in which the first and last bits and any number of intermediate bits are received in error.

Example 5.11 Continuing with Example 5.10, for $D = 1010001101$, we have $D(X) = X^9 + X^7 + X^3 + X^2 + 1$, and for $P = 110101$, we have $P(X) = X^5 + X^4 + X^2 + 1$. We should end up with $R = 01110$, which corresponds to $R(X) = X^3 + X^2 + X$. Figure 5.14 shows the polynomial division that corresponds to the binary division in the preceding example.

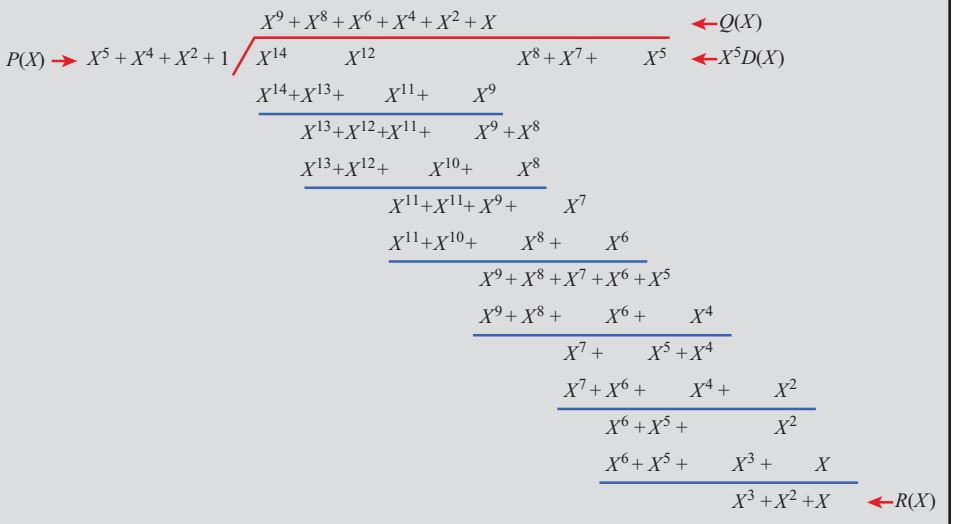


Figure 5.14 Polynomial Division for Example 5.10

where r is the length of the FCS. This means there are 2^r possible error patterns, and only one of those patterns will go undetected.

Four versions of $P(X)$ have been widely used:

$$\text{CRC-12} = X^{12} + X^{11} + X^3 + X^2 + X + 1$$

$$\text{CRC-16} = X^{16} + X^{15} + X^2 + 1$$

$$\text{CRC-CCITT} = X^{16} + X^{12} + X^5 + 1$$

$$\text{CRC-32} = X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$$

The CRC-12 system is used for transmission of streams of 6-bit characters and generates a 12-bit FCS. Both CRC-16 and CRC-CCITT are popular for 8-bit characters, in the United States and Europe, respectively, and both result in a 16-bit FCS. This would seem adequate for most applications, although CRC-32 is specified as an option in some point-to-point synchronous transmission standards.

Block Error Correction Codes

Error detection is a useful technique, found in data link control protocols, such as high-level data link control (HDLC), and in transport protocols, such as TCP. However, correction of errors using an error detection code requires that block of

data be retransmitted, using the ARQ discipline explained in detail in Section 10.4. For wireless applications, this approach is inadequate for two reasons.

1. The bit error rate on a wireless link can be quite high, which would result in a large number of retransmissions.
2. In some cases, especially satellite links, the propagation delay is very long compared to the transmission time of a single frame. The result is a very inefficient system. As is discussed in Section 10.4, the common approach to retransmission is to retransmit the frame in error plus all subsequent frames. With a long data link, an error in a single frame necessitates retransmitting many frames.

Instead, it would be desirable to enable the receiver to correct errors in an incoming transmission on the basis of the bits in that transmission. Figure 5.15 shows in general how this is done. On the transmission end, each k-bit block of data is mapped into an n-bit block ($n > k$) called a **codeword**, using an FEC encoder. The codeword is then transmitted; in the case of wireless transmission a modulator produces an analog signal for transmission. During transmission, the signal is subject to noise, which may produce bit errors in the signal. At the receiver, the incoming signal is demodulated to produce a bit string that is similar to the original codeword but may contain errors. This block is passed through an FEC decoder, with one of five possible outcomes:

1. If there are no bit errors, the input to the FEC decoder is identical to the original codeword, and the decoder produces the original data block as output.

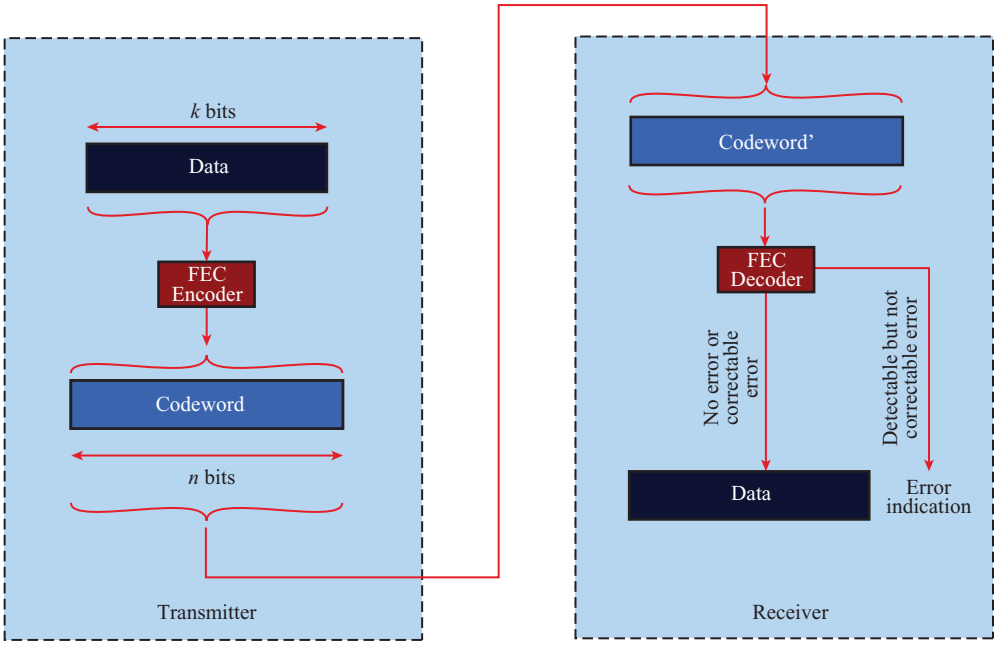


Figure 5.15 Forward Error Correction Process

2. For certain error patterns, it is possible for the decoder to detect and correct those errors. Thus, even though the incoming data block differs from the transmitted codeword, the FEC decoder is able to map this block into the original data block.
3. For certain error patterns, the decoder can detect but not correct the errors. In this case, the decoder simply reports an uncorrectable error.
4. For certain, typically rare, error patterns, the decoder detects an error, but does not correct it properly. It assumes a certain block of data was sent when in reality of different one was sent.
5. For certain even more rare error patterns, the decoder does not detect that any errors have occurred and maps the incoming n-bit data block into a k-bit block that differs from the original k-bit block.

How is it possible for the decoder to correct bit errors? In essence, error correction works by adding redundancy to the transmitted message. Consider an example where a binary 0 or 1 were to be sent, but instead the codewords that were sent were either 0000 or 1111. The redundancy makes it possible for the receiver to deduce what the original message was, even in the face of a certain level of error rate. If a 0010 were received, we could assume that a 0000 was sent corresponding to the original binary 0, since only one bit change would have occurred to make this happen. There is, however, a much more unlikely yet possible scenario were a 1111 was sent. The decoder would then make a mistake by assuming a 0 was sent. Consider if another received codeword were 0011. In this case, the decoder would not be able to decide since it would be equally likely that 0000 or 1111 was sent.

In this section, we look at a widely used form of error correction code known as a block error correction code. We begin with a discussion of general principles. Before proceeding, we note that in many cases, the error correction code follows the same general layout as shown for error detection codes in Figure 5.13. That is, the FEC algorithm takes as input a k-bit block and adds $(n - k)$ check bits to that block to produce an n-bit block; all of the bits in the original k-bit block show up in the n-bit block. For some FEC algorithms, such as the convolutional code, the FEC algorithm maps the k-bit input into an n-bit codeword in such a way that the original k bits do not appear in the codeword.

Block Code Principles

To begin, we define a term that shall be of use to us. The **Hamming distance** $d(v_1, v_2)$ between two n-bit binary sequences v_1 and v_2 is the number of bits in which v_1 and v_2 disagree. For example, if

$$v_1 = 011011, \qquad v_2 = 110001$$

then

$$d(v_1, v_2) = 3$$

Suppose we wish to transmit blocks of data of length k bits. Instead of transmitting each block as k bits, we map each k-bit sequence into a unique n-bit codeword.

Example 5.12 For $k = 2$ and $n = 5$, we can make the following assignment:

Data block	Codeword
00	00000
01	00111
10	11001
11	11110

Now, suppose that a codeword block is received with the bit pattern 00100. This is not a valid codeword and so the receiver has detected an error. Can the error be corrected? We cannot be sure which data block was sent because 1, 2, 3, 4, or even all 5 of the bits that were transmitted may have been corrupted by noise. However, notice that it would require only a single bit change to transform the valid codeword 00000 into 00100. It would take two bit changes to transform 00111 to 00100, three bit changes to transform 11110 to 00100, and it would take four bit changes to transform 11001 into 00100. Thus, we can deduce that the most likely codeword that was sent was 00000 and that therefore the desired data block is 00. This is error correction. In terms of Hamming distances, we have

$$\mathrm{d}(00000, 00100) = 1; \mathrm{d}(00111, 00100) = 2;$$
$$\mathrm{d}(11001, 00100) = 4; \mathrm{d}(11110, 00100) = 3$$

So the rule we would like to impose is that if an invalid codeword is received, then the valid codeword that is closest to it (minimum distance) is selected. This will only work if there is a unique valid codeword at a minimum distance from each invalid codeword.

For our example, it is not true that for every invalid codeword there is one and only one valid codeword at a minimum distance. There are $2^5 = 32$ possible codewords of which 4 are valid, leaving 28 invalid codewords. For the invalid codewords, we have the following:

Invalid Codeword	Minimum Distance	Valid Codeword	Invalid Codeword	Minimum Distance	Valid Codeword
00001	1	00000	10000	1	00000
00010	1	00000	10001	1	11001
00011	1	00111	10010	2	00000 or 11110
00100	1	00000	10011	2	00111 or 11001
00101	1	00111	10100	2	00000 or 11110
00110	1	00111	10101	2	00111 or 11001
01000	1	00000	10110	1	11110
01001	1	11001	10111	1	00111
01010	2	00000 or 11110	11000	1	11001
01011	2	00111 or 11001	11010	1	11110
01100	2	00000 or 11110	11011	1	11001
01101	2	00111 or 11001	11100	1	11110
01110	1	11110	11101	1	11001
01111	1	00111	11111	1	11110

There are eight cases in which an invalid codeword is at a distance 2 from two different valid codewords. Thus, if one such invalid codeword is received, an error in 2 bits could have caused it and the receiver has no way to choose between the two alternatives. An error is detected but cannot be corrected. The only remedy is retransmission. However, in every case in which a single bit error occurs, the resulting codeword is of distance 1 from only one valid codeword and the decision can be made. This code is therefore capable of correcting all single-bit errors but cannot correct double bit errors. Another way to see this is to look at the pairwise distances between valid codewords:

$$\mathrm{d}(00000, 00111) = 3;\ \mathrm{d}(00000, 11001) = 3;\ \mathrm{d}(00000, 11110) = 4;$$
$$\mathrm{d}(00111, 11001) = 4;\ \mathrm{d}(00111, 11110) = 3;\ \mathrm{d}(11001, 11110) = 3;$$

The minimum distance between valid codewords is 3. Therefore, a single bit error will result in an invalid codeword that is a distance 1 from the original valid codeword but a distance at least 2 from all other valid codewords. As a result, the code can always correct a single-bit error. Note that the code also will always detect a double-bit error.

The preceding example illustrates the essential properties of a block error-correcting code. An (n, k) block code encodes k data bits into n-bit codewords. Thus the design of a block code is equivalent to the design of a function of the form $\mathbf{v}_c = \mathrm{f}(\mathbf{v}_d)$, where $\mathbf{v}_d$ is a vector of k data bits and $\mathbf{v}_c$ is a vector of n codeword bits.

With an (n, k) block code, there are 2^k valid codewords out of a total of 2^n possible codewords. The ratio of redundant bits to data bits, $(n - k)/k$, is called the **redundancy** of the code, and the ratio of data bits to total bits, k/n, is called the **code rate**. The code rate is a measure of how much additional bandwidth is required to carry data at the same data rate as without the code. For example, a code rate of 1/2 requires double the bandwidth of an uncoded system to maintain the same data rate. Our example has a code rate of 2/5 and so requires a bandwidth 2.5 times the bandwidth for an uncoded system. For example, if the data rate input to the encoder is 1 Mbps, then the output from the encoder must be at a rate of 2.5 Mbps to keep up.

For a code consisting of the codewords $\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_s$, where $s = 2^k$, the minimum distance $d_{\min}$ of the code is defined as

$$d_{\min} = \min_{i \neq j} \lfloor d(\mathbf{w}_i, \mathbf{w}_j) \rfloor$$

It can be shown that the following conditions hold. For a given positive integer t, if a code satisfies $d_{\min} \geq 2t + 1$, then the code can correct all bit errors up to and including errors of t bits. If $d_{\min} \geq 2t$, then all errors $\leq t - 1$ bits can be corrected and errors of t bits can be detected but not, in general, corrected. Conversely, any code for which all errors of magnitude $\leq t$ are corrected must satisfy $d_{\min} \geq 2t + 1$, and any code for which all errors of magnitude $\leq t - 1$ are corrected and all errors of magnitude t are detected must satisfy $d_{\min} \geq 2t$.

Another way of putting the relationship between $d_{\min}$ and t is to say that the maximum number of guaranteed correctable errors per codeword satisfies

$$t = \left\lfloor \frac{d_{\min} - 1}{2} \right\rfloor$$

where $\lfloor x \rfloor$ means the largest integer not to exceed x (e.g., $\lfloor 6.3 \rfloor = 6$). Furthermore, if we are concerned only with error detection and not error correction, then the number of errors, t, that can be detected satisfies

$$t = d_{\min} - 1$$

To see this, consider that if $d_{\min}$ errors occur, this could change one valid codeword into another. Any number of errors less than $d_{\min}$ cannot result in another valid codeword.

The design of a block code involves a number of considerations:

1. For given values of n and k, we would like the largest possible value of $d_{\min}$.
2. The code should be relatively easy to encode and decode, requiring minimal memory and processing time.
3. We would like the number of extra bits, $(n - k)$, to be small, to reduce bandwidth.
4. We would like the number of extra bits, $(n - k)$, to be large, to reduce error rate.

Clearly, the last two objectives are in conflict, and trade-offs must be made.

Let us examine Figure 5.16. The literature on error-correcting codes frequently includes graphs of this sort to demonstrate the effectiveness of various encoding schemes. Recall from earlier in this section that modulation schemes can be chosen to reduce the required E_b/N_0 value to achieve a given bit error rate. Modulation has to do with the definition of signal elements to represent bits. This modulation also has an effect on E_b/N_0. In Figure 5.16, the curve on the right is for an uncoded modulation system; the shaded region represents the area in which potential improvement can be achieved. In this region, a smaller BER is achieved for a given E_b/N_0, and conversely, for a given BER, a smaller E_b/N_0 is required. The other curve is a typical result of a code rate of one-half (equal number of data and check bits). Note that at an error rate of 10^{-6}, the use of coding allows a reduction in E_b/N_0 of 2.77 dB. This reduction is referred to as the **coding gain**, which is defined as the reduction, in decibels, in the required E_b/N_0 to achieve a specified BER of an error-correcting coded system compared to an uncoded system using the same modulation.

It is important to realize that the BER for the second rate 1/2 curve refers to the rate of uncorrected errors and that the E_b value refers to the energy per data bit. Because the rate is 1/2, there are two bits on the channel for each data bit, effectively reducing the data throughput by 1/2 as well. The energy per coded bit is half that of the energy per data bit, or a reduction of 3 dB. If we look at the energy per coded bit for this system, then we see that the channel bit error rate is about 2.4×10^{-2}, or 0.024.

Finally, note that below a certain threshold of E_b/N_0, the coding scheme actually degrades performance. In our example of Figure 5.16, the threshold occurs at about 5.4 dB. Below the threshold, the extra check bits add overhead to the system that reduces the energy per data bit causing increased errors. Above the threshold, the error-correcting power of the code more than compensates for the reduced E_b, resulting in a coding gain.

Commonly used error correction block codes are Hamming, Cyclic, BCH, and Reed-Solomon codes. Chapter 10 provides details on these specific block error correction codes.

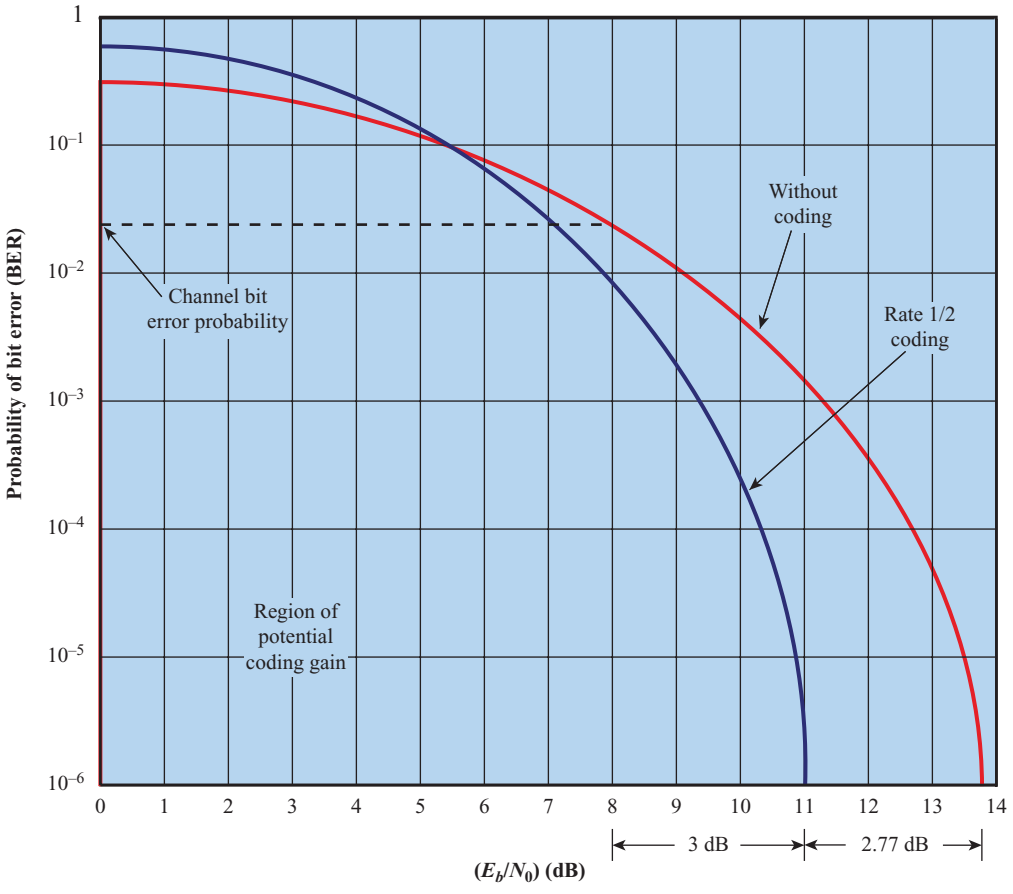


Figure 5.16 How Coding Improves System Performance

Low-Density Parity-Check Codes

For decades, researchers were not able to approach the Shannon limit for data capacity for a given channel bandwidth, at least not within practical computational hardware constraints. There were some capacity-approaching codes developed in the 1960s, called low-density parity-check (LDPC) codes, which were rediscovered in the 1990s. They only then became of practical use, since their computational complexity was at first prohibitive. They have since been enhanced and become popular, for example in the latest generation of IEEE 802.11 standards.

LDPC uses very long block codes, normally longer than 1000 bits. To check for errors among these bits, a series of parity equations are implemented, usually organized in an **H** matrix. For example, one might require the following:

$$b_{10} + b_{13} + b_{45} + b_{192} = 0$$

Each equation should have at least three bits added together, and there will be hundreds of such equations for 1000 bits.

To visualize a few of these equations, see Figure 5.17. This is a Tanner graph. The nodes in the top row correspond to each of the data bits and are called *variable nodes*. The nodes in the bottom row are called *constraint nodes*, and these correspond to the equations. For example, constraint node c_1 corresponds to the following equation:

$$v_3 + v_4 + v_5 + v_6 = 0$$

LDPC uses an iterative decoding procedure as follows:

1. The procedure starts with the variable nodes at the top. These nodes use external information, mainly from the demodulator, to determine their estimates for their bit values. If they use a soft decoding approach, they also estimate the probabilities that the bits should be 0 or 1.
2. These estimates are then sent to the constraint nodes to see if the estimated values satisfy all of the equations. If so, the decoding stops since an acceptable answer has been found. If not, the constraint nodes combine the information sent to them from their connected variable nodes to determine which bits are most likely to be different than their estimates. This corresponds to the most likely bit changes that are needed to satisfy the equations.
3. The estimates from the constraint nodes are sent to the variable nodes. Since variable nodes are connected to multiple constraint nodes, the variable nodes combine the newly acquired information to update their estimates of their bit values and probabilities.
4. These are sent again to the constraint nodes. If the equations are now satisfied, then stop. Otherwise, continue the decoding process.

This decoding procedure is known as *message passing* or *belief propagation*. The performance of LDPC codes can be impressive, approaching Shannon capacity within a fraction of a dB when using long codes.

Convolutional Codes

Block codes are one of the two widely used categories of error correction codes for wireless transmission; the other is convolutional codes. A (n, k) block code processes data in blocks of k bits at a time, producing a block of n bits $(n > k)$ as output

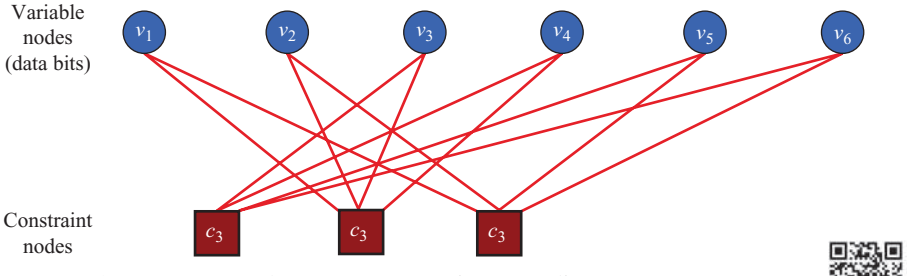


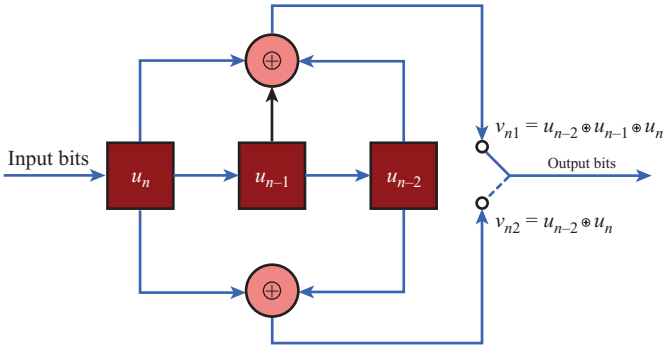
Figure 5.17 Tanner Graph for LDPC Iterative Decoding

for every block of k bits as input. If data are transmitted and received in a more or less continuous stream, a block code, particularly one with a large value of n, may not be as convenient as a code that generates redundant bits continuously so that error checking and correcting are carried out continuously. This is the function of convolutional codes.

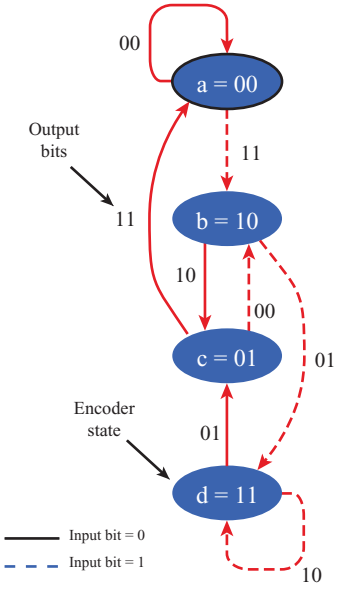
A convolutional code is defined by three parameters: n, k, and K. An (n, k, K) code processes input data k bits at a time and produces an output of n bits for each incoming k bits. So far this is the same as the block code. In the case of a convolutional code, n and k are generally quite small numbers. The difference is that convolutional codes have memory, which is characterized by the *constraint factor* K. In essence, the current n-bit output of an (n, k, K) code depends not only on the value of the current block of k input bits but also on the previous $K - 1$ blocks of k input bits. Hence, the current output of n bits is a function of the last $K \times k$ input bits.

Convolutional codes are best understood by looking at a specific example. We use the example shown in Figure 5.18. There are two alternative representations of the code shown in the figure. Figure 5.18a is a shift register, which is most convenient for describing and implementing the encoding process. Figure 5.18b is an equivalent representation that is useful in discussing the decoding process.

For an (n, k, K) code, the shift register contains the most recent $K \times k$ input bits; the register is initialized to all zeros.[5] The encoder produces n output bits, after



(a) Encoder shift register



(b) Encoder state diagram

Figure 5.18 Convolutional Encoder with $(n, k, K) = (2, 1, 3)$

[5] In some of the literature, the shift register is shown with one less storage cell and with the input bits feeding the XOR circuits as well as a storage cell; the depictions are equivalent.

which the oldest k bits from the register are discarded and k new bits are shifted in. Thus, although the output of n bits depends on $K \times k$ input bits, the rate of encoding is n output bits per k input bits. As in a block code, the code rate is therefore k/n. The most commonly used binary encoders have $k = 1$ and hence a shift register length of K. Our example is of a (2, 1, 3) code (Figure 5.18a). The shift register holds $K \times k = 3 \times 1$ *bits* u_n, u_{n-1}, and u_{n-2}. For each new input bit u_n, two output bits v_{n1} and v_{n2} are produced using the three most recent bits. The first output bit produced here is from the upper logic circuit ($v_{n1} = u_n \oplus u_{n-1} \oplus u_{n-2}$), and the second output bit from the lower logic circuit ($v_{n2} = u_n \oplus u_{n-2}$).

For any given input of k bits, there are $2^{k(K-1)}$ different functions that map the k input bits into n output bits. Which function is used depends on the history of the previous $(K - 1)$ input blocks of k bits each. We can therefore represent a convolutional code using a finite-state machine. The machine has $2^{k(K-1)}$ states, and the transition from one state to another is determined by the most recent k bits of inputs and produces n output bits. The initial state of the machine corresponds to the all-zeros state. For our example (Figure 5.18b) there are four states, one for each possible pair of values for the previous two bits. The next input bit causes a transition and produces an output of two bits. For example, if the last two bits were 10 ($u_{n-1} = 1, u_{n-2} = 0$) and the next bit is 1 ($u_n = 1$), then the current state is state b (10) and the next state is d (11). The output is

$$v_{n1} = u_{n-2} \oplus u_{n-1} \oplus u_n = 0 \oplus 1 \oplus 1 = 0$$

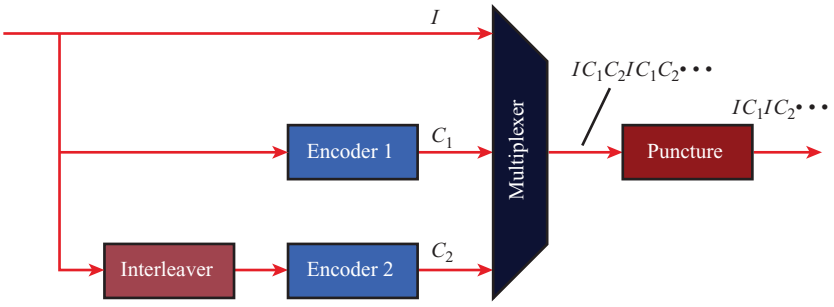
$$v_{n2} = 0 \oplus 1 = 1$$

Convolutional codes provide good performance in noisy channels where a high proportion of the bits are in error. Thus, they have found increasing use in wireless applications.

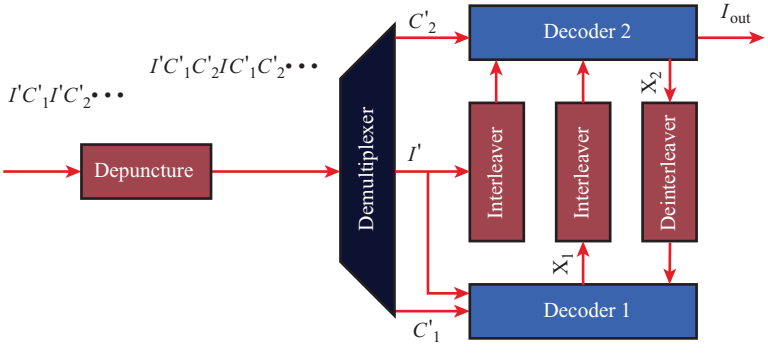
Turbo Coding As higher and higher speeds are used in wireless applications, error correction continues to pose a major design challenge. **Turbo codes** have emerged as a popular choice for third- and fourth-generation wireless systems. Turbo codes exhibit performance, in terms of bit error probability, that is very close to the Shannon limit and can be efficiently implemented for high-speed use. A number of different turbo encoders and decoders have been introduced, most of which are based on convolutional encoding. In this subsection, we give a general overview.

Figure 5.19a depicts a turbo encoder. In this scheme, the encoder is replicated. One copy of the encoder receives a stream of input bits and produces a single output check bit C_1 for each input bit. The input to the other encoder is an interleaved version of the input bit stream, producing a sequence of C_2 check bits. The initial input bit plus the two check bits are then multiplexed to produce the sequence $I_1C_{11}C_{21}I_2C_{12}C_{22}\ldots$, that is, the first input bit followed by the first bit from encoder one, followed by the first bit from encoder 2, and so on. The resulting sequence has a code rate of 1/3. A code rate of 1/2 can be achieved by taking only half of the check bits, alternating between outputs from the two encoders; this process is called *puncturing*. Rates of 1/3 and 1/2 are both found in third- and fourth-generation systems.

Note that each encoder only produces a single check bit for each input bit and that the input bit is preserved. In the convolutional encoders we have discussed so far (e.g., Figure 5.18a), the input bits are not preserved, and there are multiple output



(a) Encoder



(b) Decoder

Figure 5.19 Turbo Encoding and Decoding

bits (n output check bits for k input bits). For turbo coding, a variation of the convolutional code, known as a recursive systematic convolutional code (RSC), is used. Figure 5.20 shows how a turbo encoder can be implemented using two RSC coders.

Figure 5.19b is a general diagram of a turbo decoder. The received data is depunctured, if necessary, by estimating the missing check bits or by setting the missing bits to 0. Decoder 1 operates first, using the estimated I' and C_1' values received from the demodulator. These values are not simply 0 or 1, but rather are larger or smaller values given the demodulator's confidence in its decision. This is called *soft decision decoding*. Decoder 1 then produces correction (X_1) values. The I' and X_1 values are fed into decoder 2, together with the C_2' values. Interleaving must be performed to align bits properly. Decoder 2 uses all of its input to produce correction values X_2. These are fed back to Decoder 1 for a second iteration of the decoding algorithm, being first deinterleaved for alignment. After sufficient iterations to produce a high level of confidence, an output bit is generated. This may take several iterations to produce a good result, which could cause significant delay. Turbo coding's use of interleaving, parallel encoding, puncturing, soft decision decoding, and feedback gives it high performance.

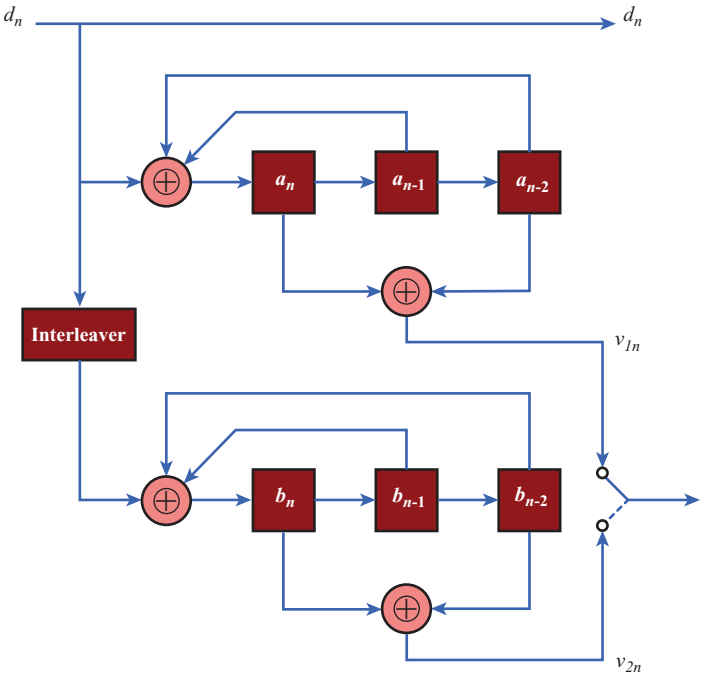


Figure 5.20 Parallel Concatenation of Two RSC Encoders

Automatic Repeat Request

Automatic repeat request (ARQ) is a mechanism used in data link control and transport protocols and relies on the use of an error detection code, such as the cyclic redundancy check (CRC) described earlier in this section. In what follows, we refer to the block of data that is transmitted from one protocol entity to another as a protocol data unit (PDU); this term was introduced in Chapter 4.

Error control mechanisms detect and correct errors that occur in the transmission of PDUs. The model that we will use, which covers the typical case, is illustrated in Figure 5.21b. Data are sent as a sequence of PDUs; PDUs arrive in the same order in which they are sent; and each transmitted PDU suffers an arbitrary and variable amount of delay before reception. In addition, we admit the possibility of two types of errors:

- **Lost PDU:** A PDU fails to arrive at the other side. For example, a noise burst may damage a PDU to the extent that the receiver is not aware that a PDU has been transmitted.
- **Damaged PDU:** A recognizable PDU does arrive, but some of the bits are in error (have been altered during transmission) and cannot be corrected.

The most common techniques for error control are based on some or all of the following ingredients:

- **Error detection:** The receiver detects errors and discards PDUs that are in error.

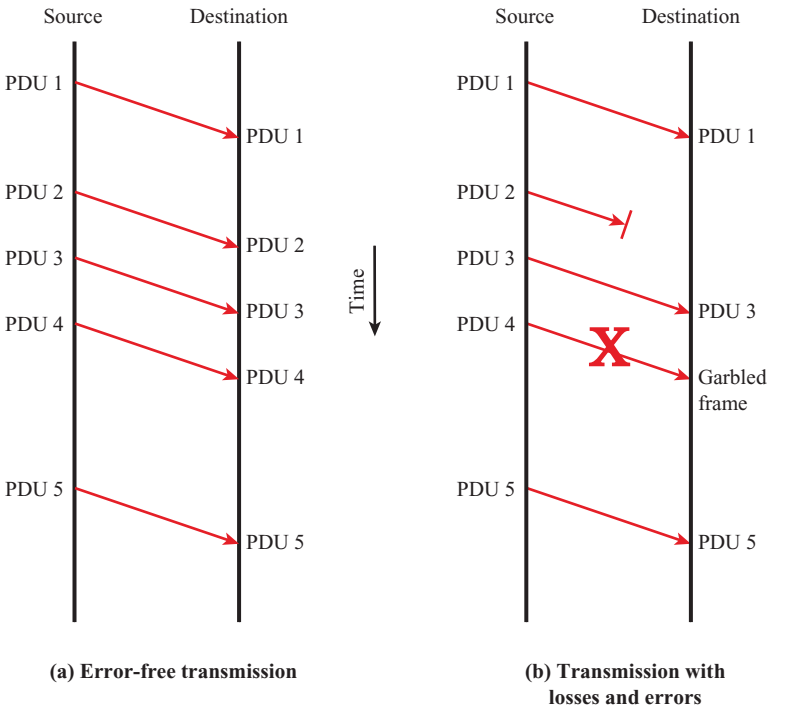


Figure 5.21 Model of PDU Transmission

- **Positive acknowledgment:** The destination returns a positive acknowledgment to successfully received, error-free PDUs.
- **Retransmission after timeout:** The source retransmits a PDU that has not been acknowledged after a predetermined amount of time.
- **Negative acknowledgment and retransmission:** The destination returns a negative acknowledgment to PDUs in which an error is detected. The source retransmits such PDUs.

Collectively, these mechanisms are all referred to as **automatic repeat request (ARQ)**; the effect of ARQ is to turn an unreliable data link into a reliable one. The most commonly used version of ARQ is known as go-back-N ARQ. In go-back-N ARQ, a station may send a series of PDUs sequentially numbered modulo some maximum value. The number of unacknowledged PDUs outstanding is determined by window size, using the sliding-window flow control technique. When no errors occur, the destination will acknowledge incoming PDUs with RR = receive ready or with a piggybacked acknowledgment on a data PDU. If the destination station detects an error in a PDU, it sends a negative acknowledgment (REJ = reject) for that PDU. The destination station will discard that PDU and all future incoming PDUs until the PDU in error is correctly received. Thus the source station, when it receives a REJ, must retransmit the PDU in error plus all succeeding PDUs that had been transmitted in the interim. Hence, the name go-back-N to retransmit these PDUs.

Consider that station A is sending PDUs to station B. After each transmission, A sets an acknowledgment timer for the PDU just transmitted. Suppose that B has previously successfully received PDU $(i - 1)$ and A has just transmitted PDU i. The go-back-N technique takes into account the following contingencies:

1. **Damaged PDU.** If the received PDU is invalid (i.e., B detects an error), B discards the PDU and takes no further action as the result of that PDU. There are two subcases:
   a. Within a reasonable period of time, A subsequently sends PDU $(i + 1)$. B receives PDU $(i + 1)$ out of order since it is expecting PDU (i) and sends a REJ i. A must retransmit PDU i and all subsequent PDUs.
   b. A does not soon send additional PDUs. B receives nothing and returns neither an RR nor a REJ. When A's timer expires, it transmits an RR PDU that includes a bit known as the P bit, which is set to 1. B interprets the RR PDU with a P bit of 1 as a command that must be acknowledged by sending an RR indicating the next PDU that it expects, which is PDU i. When A receives the RR, it retransmits PDU i.
2. **Damaged RR.** There are two subcases:
   a. B receives PDU i and sends RR $(i + 1)$, which suffers an error in transit. Because acknowledgments are cumulative (e.g., RR 6 means that all PDUs through 5 are acknowledged), it may be that A will receive a subsequent RR to a subsequent PDU and that it will arrive before the timer associated with PDU i expires.
   b. If A's timer expires, it transmits an RR command as in Case 1b. It sets another timer, called the P-bit timer. If B fails to respond to the RR command, or if its response suffers an error in transit, then A's P-bit timer will expire. At this point, A will try again by issuing a new RR command and restarting the P-bit timer. This procedure is tried for a number of iterations. If A fails to obtain an acknowledgment after some maximum number of attempts, it initiates a reset procedure.
3. **Damaged REJ.** If a REJ is lost, this is equivalent to Case 1b.

Figure 5.22 is an example of the PDU flow for go-back-N ARQ. It sends RRs only for even numbered PDUs. Because of the propagation delay on the line, by the time that an acknowledgment (positive or negative) arrives back at the sending station, it has already sent two additional PDUs beyond the one being acknowledged. Thus, when a REJ is received to PDU 5, not only PDU 5 but PDUs 6 and 7 must be retransmitted. Thus, the transmitter must keep a copy of all unacknowledged PDUs.

Hybrid Automatic Repeat Request In practical wireless system implementation, neither FEC nor ARQ is an adequate one-size-fits-all solution. FEC may add unnecessary redundancy (i.e., use extra bandwidth) if channel conditions are good and ARQ with error detection may cause excessive delays from retransmissions in poor channel conditions. Therefore, a solution known as ***Hybrid Automatic Repeat Request (HARQ)*** has been widely implemented in today's systems; it uses a combination of FEC to correct the most common errors and ARQ for retransmission when FEC cannot make corrections. Going beyond this basic concept, the following additional approaches may be implemented.

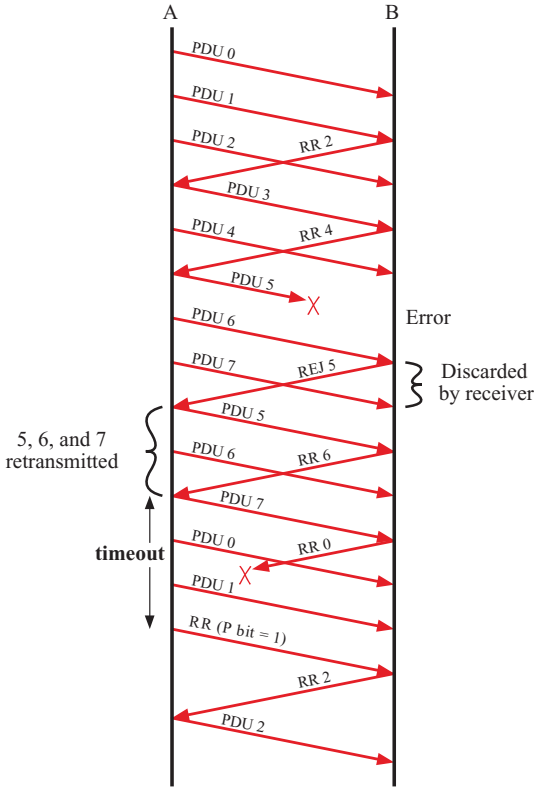


Figure 5.22 Go-back-N ARQ

- **Soft Decision Decoding:** The decoding process can provide not just an assessment of a bit being 0 or 1, but also levels of confidence in those results.
- **Chase Combining:** Previous frames that were not corrected by FEC need not be discarded. The soft decision information can be stored and then combined with soft decision information from retransmissions. If using turbo coding as seen in Figure 5.19, the decoders in the receivers now use information from multiple frames not just the current frame. This will result in stronger FEC capabilities. In *chase combining,* the exact same frames are retransmitted each time and soft combined.
- **Incremental Redundancy:** Each time a sender retransmits, different coding information can be provided. This can accomplish two goals.
  1. Lower Overhead. The initial packets can include less coding; if enough, then the packet can be successful and overhead can be avoided. For example, the first frame may only include a few bytes of an error detection code like CRC, with later frames then including FEC after the first frame has errors.
  2. Stronger Correction. The retransmissions can provide different coding at the same coding rates or stronger coding at lower coding rates. If adapted to the current wireless channel environment, this will increase the probability of a successful transmission by the second or third frames.

- **Puncturing:** To provide the various coding rates for incremental redundancy, a different FEC coding algorithm could be used each time. A simpler approach, however, is puncturing, which removes bits to increase the coding rate.

Example 5.13 Consider an FEC coder that produces a 1/3 rate code that is punctured to become a 1/2 rate code. Say there are 100 bits of data that become a 300 bit FEC codeword. To become a 1/2 rate FEC codeword, there need to be 2 bits of codeword for every 1 bit of data, hence a 200-bit codeword. This means 100 bits, 1 of every 3 bits of the original FEC code, need to be punctured. At the receiver, the missing 100 bits would be replaced before decoding. These could just be replaced with random numbers, which would mean that roughly 50 of those would coincidentally be correct and the other 50 incorrect. The original FEC code may actually still be plenty effective enough to correct those errors if the received signal-to-noise ratio is relatively good. If not, a later retransmission might use less puncturing or puncture different bits.

In general, a punctured code is weaker than an unpunctured code at the same rate. However, simply puncturing the same code to achieve different coding rates allows the decoder structure to remain the same, instead of having multiple decoders for different code rates. The benefits of this reduction in complexity can outweigh the reduction in performance from puncturing. Used with HARQ incremental redundancy, puncturing will take a single output from an FEC coder and remove more or different bits each time.

- **Adaptive Modulation and Coding:** Systems will use channel quality information (CQI) to estimate the best modulation and coding to work with HARQ. For example, LTE uses the CQI to determine the highest modulation and coding rate that would provide a 10% block error rate for the first HARQ transmission. Also, if the CQI changes in the middle of an HARQ process, the modulation and coding might be adapted.
- **Parallel HARQ Processes:** Some systems wait until the HARQ process finishes for one frame before sending the next frame; this is known as a stop-and-wait protocol. The process of waiting for an ACK or NACK, followed by possible multiple retransmissions can be time consuming, however. Therefore, some HARQ implementations allow for multiple open HARQ operations to be occurring at the same time. This is known as an *N-channel Stop-and-Wait* protocol.

5.7 ORTHOGONAL FREQUENCY DIVISION MULTIPLEXING (OFDM)

This section looks at OFDM-based techniques that have created great expansion in the capacity of wireless networks. The main air interface technology to move from third to fourth-generation cellular is OFDM. OFDM also allowed the expansion of IEEE 802.11 data rates. We first look at the basic mechanisms of OFDM, namely orthogonal carriers and transmitter design based on the inverse fast Fourier

transform. Then we look at the ways OFDM is used in practical systems for multiple access.

OFDM Basics

OFDM, also called *multicarrier modulation*, uses multiple carrier signals at different frequencies, sending some of the bits on each channel. This is similar to FDM. However, in the case of OFDM, many **subcarriers** are dedicated to a single data source.

Figure 5.23 illustrates a conceptual understanding of OFDM. Actual transmitter operation is simplified, but the basic concept can first be understood here.

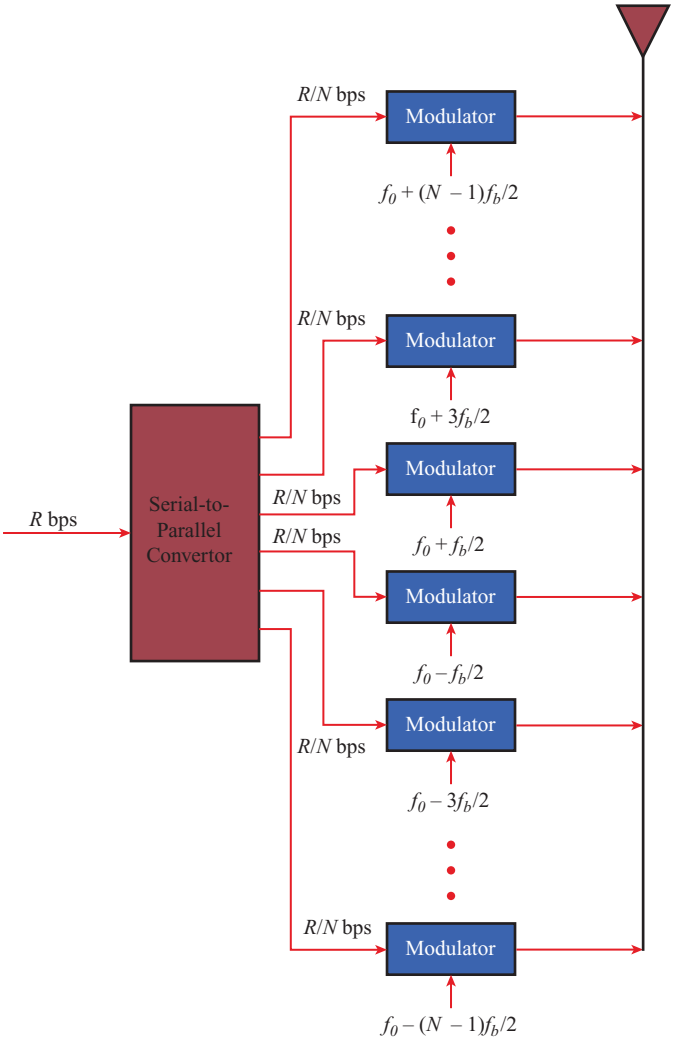


Figure 5.23 Conceptual Understanding of Orthogonal Frequency-Division Multiplexing

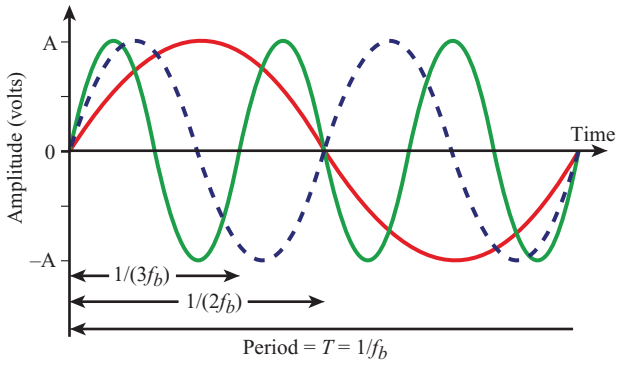
Suppose we have a binary data stream operating at R bps and an available bandwidth of Nf_b, centered at f_0. The entire bandwidth could be used to send the data stream, in which case each bit duration would be $1/R$. The alternative is to split the data stream into N substreams, using a serial-to-parallel converter. Each substream has a data rate of R/N bps and is transmitted on a separate subcarrier, with a spacing between adjacent subcarriers of f_b. Now the bit duration is N/R, which is substantially longer and creates special capabilities to overcome multipath fading.

OFDM relies on a principle known as *orthogonality*. If two subcarriers are placed at just that right spacing, each subcarrier signal can be retrieved at the receiver even if the signals overlap. The result is shown in Figure 5.24b; it looks like the signals are packed too close together because they overlap substantially. Previous FDM approaches are illustrated in Figure 5.24c, which assumed signals should be spaced sufficiently apart in frequency to (1) avoid overlap in the frequency bands and (2) to provide extra spacing known as *guard bands* to prevent the effects of adjacent carrier interference from out-of-band emissions. But OFDM is able to drastically improve the use of frequency spectrum. In Figure 5.24b over 5.24c, the number of signals that can be supported has increased by a factor of 6!

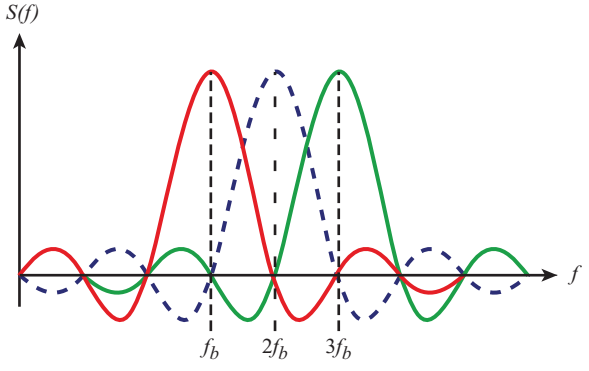
OFDM has several advantages. First, frequency selective fading only adversely affects some subcarriers and not the whole signal. If the data stream is protected by a forward error correction code, this type of fading is easily handled. But more importantly, OFDM overcomes the problems of intersymbol interference (ISI) that is caused by the multitude of multipath signals. The symbol times are very long compared to the spread of the delays from the multipath signals. In addition, OFDM adds a set of extra bits, known as the cyclic prefix, to the beginning of each set of bits to reduce this multipath effect even further.

OFDM Implementation Even though OFDM dates back some 40 years, it was only until the 1990s that an advance was made to make OFDM an economical technology. Figure 5.23 showed a conceptual understanding of OFDM where a data stream is split into many lower bit rate streams and then modulated on many different subcarriers. Such an approach, however, would result in a very expensive transmitter and receiver since it would have some many expensive oscillators to generate each subcarrier frequency. Fortunately, OFDM can instead be implemented by taking advantage of the properties of the **fast Fourier transform (FFT)** and **inverse fast Fourier transform (IFFT)**.

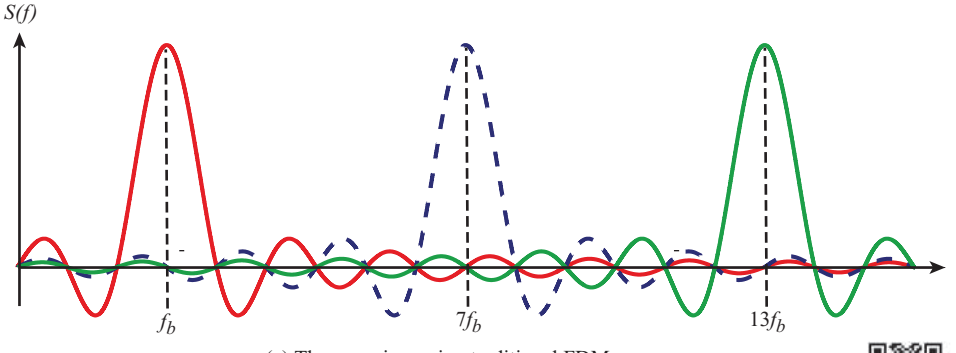
The implementation of OFDM using the FFT and IFFT is illustrated in Figure 5.25. The data stream undergoes a serial to parallel (S/P) operation, which takes a sample from each carrier and make a group of samples called an **OFDM symbol**. Each value in a sense gives a weight for each subcarrier. Then the IFFT (not FFT) takes the values for these subcarriers and computes the time domain data stream to be transmitted, which are a combination of these subcarriers. The IFFT operation has the effect of ensuring that the subcarriers do not interfere with each other. These values are put back into a serial stream with a P/S operation, then the stream is modulated onto the carrier using one oscillator. At the receiver, the reverse operation is conduced. An FFT module is used to map the incoming signal back to the M subcarriers, from which the data streams are recovered as the weights for each subcarrier are retrieved for each sample.



(a) Three subcarriers in time domain



(b) Three orthogonal subcarriers in frequency domain



(c) Three carriers using traditional FDM

Figure 5.24 Illustration of Orthogonality of OFDM

Orthogonal Frequency Division Multiple Access (OFDMA)

Multiple access strategies share a wireless channel by making use of scheduled times (time division multiple access), random access times (carrier sense multiple access), scheduled use of frequencies (frequency division multiple access), coded spreading of signals (direct sequence spread spectrum), and/or coded frequency hopping of signals (frequency hopping spread spectrum). Throughout this text, one of the

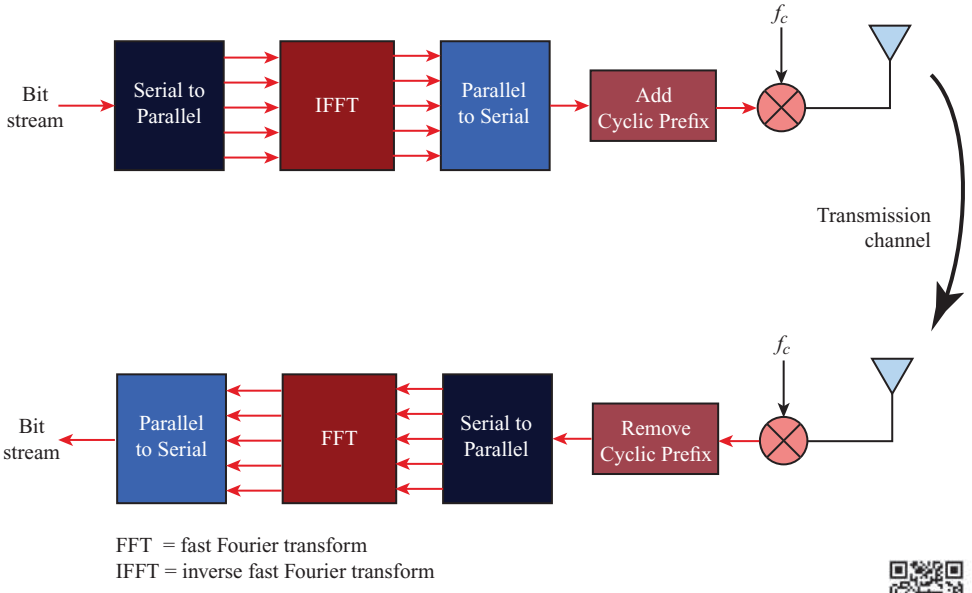


Figure 5.25 IFFT Implementation of OFDM

defining attributes of a technology is how it accomplishes multiple access, both in terms of the approaches just mentioned and the protocols that are used for mobile devices to cooperate.

OFDMA uses a combination of FDMA and TDMA by allowing different users to use a subset of the subcarriers at different times. All technologies that use OFDM do not use OFDMA. For example, some forms of 802.11 use OFDM for the signal transmission, but CSMA for multiple access. The transmitter uses the full set of subcarriers when transmitting. LTE only uses OFDMA on the downlink, but instead uses a single-carrier approach on the uplink.

OFDMA uses OFDM, which employs multiple closely spaced subcarriers, but the subcarriers are divided into groups of subcarriers since it would not be computationally feasible (because of hundreds of subcarriers) or sufficient (since each subcarrier only carries a small capacity) to schedule by individual subcarrier. Each group is named a subchannel. In the downlink, a subchannel may be intended for different receivers. Figure 5.26 contrasts OFDM and OFDMA; in the OFDMA case the use of adjacent subcarriers to form a subchannel is illustrated. Depending on the technology and specifics of an expected wireless channel characteristic, subchannels can be formed using adjacent subcarriers, regularly spaced subcarriers, or randomly spaced subcarriers.

Single-carrier FDMA (SC-FDMA) is a relatively recently developed multiple access technique which has similar structure and performance to OFDMA. One prominent advantage of SC-FDMA over OFDMA is the lower peak-to-average power ratio (PAPR) of the transmit waveform, which benefits the mobile user in terms of battery life, power efficiency, and lower cost.

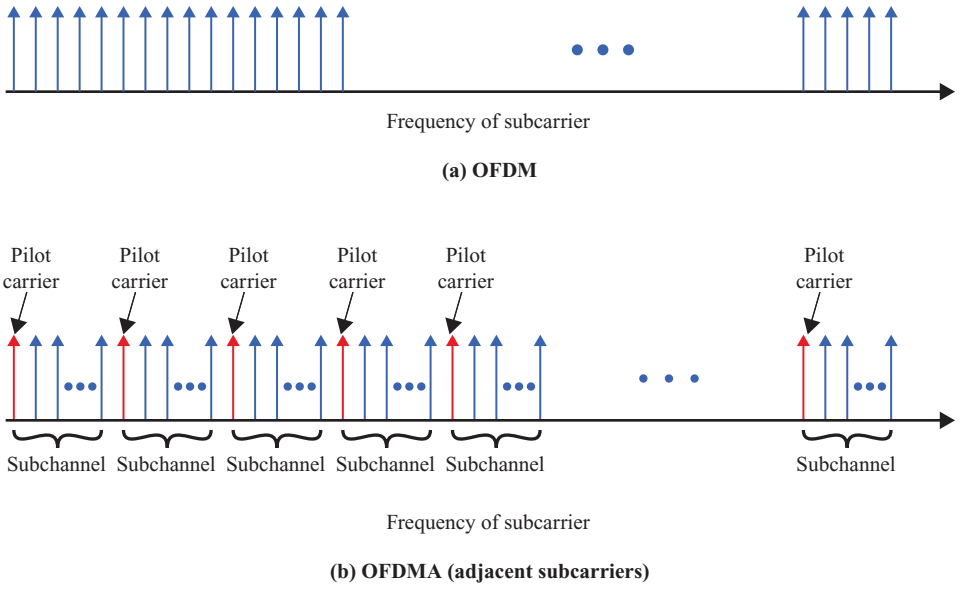


Figure 5.26 OFDM and OFDMA

Opportunistic Scheduling

Subchannelization defines subchannels, called *Resource Blocks* by LTE, which can be allocated to subscriber stations (SSs) depending on their channel conditions and data requirements. Particular power levels could also be prescribed to those stations in order to optimize throughput and limit interference.

One might think that the time varying and multipath nature of wireless communications would limit the ability for effective use of the wireless channel, but the opposite is actually true. Such variations provide opportunities that can be exploited. Since channel conditions change and are expected to change, resource allocations can adjust in a dynamic fashion. Hence, the term **opportunistic scheduling** has been used. Particular implementations and equipment providers can approach this problem in ways that provide them competitive advantage, since most standards do not prescribe OFDMA scheduling approaches. There are a variety of considerations when scheduling such subchannels.

- **Efficiency:** One could schedule subchannels based on the users with the highest signal-to-interference-plus-noise ratio (SINR) for that time slot. Those users could use adaptive modulation and coding to obtain much higher throughput than others with poorer SINR. The total efficiency and capacity would be highest; the time-varying nature of the channel would be exploited to the highest benefit.
- **Fairness:** If scheduling is only based on efficiency, however, some users (likely those far from base stations) would receive little or no throughput. Fairness could also be a consideration. A completely fair allocation might give the same number of subchannels or the same throughput to all users, but this could

sacrifice efficiency. A popular approach that finds a compromise is known as *proportional fairness*, in which every user computes the following metric during a resource allocation decision.

$$\text{Proportional fairness metric} = \frac{r_i}{\bar{r}_i}$$

This is the ratio of the rate that could be obtained for user i in that time slot for that subchannel, r_i, divided by the average rate that has been obtained for user i in that subchannel, $\bar{r}_i$. In essence, users are compared against themselves and not against others. Those which have a good opportunity *for them* will have a better chance at being scheduled.

- **Requirements:** Applications such as audio and video may have requirements on delay and jitter. These should be considered.
- **Priority:** In some situations, priority users such as police, fire, ambulance, or other public safety workers could need special priorities in emergency situations, regardless of their channel conditions. Note, however, that even for those users their channel conditions may improve significantly within a few milliseconds.

5.8 SPREAD SPECTRUM

An important form of communications is known as **spread spectrum**. The spread spectrum technique was developed initially for military and intelligence requirements. The essential idea is to spread the information signal over a wider bandwidth to make jamming and interception more difficult. The first type of spread spectrum developed is known as frequency hopping.[6] A more recent type of spread spectrum is direct sequence. Both of these techniques are used in various wireless communications standards and products, most notably 2G and 3G cellular, Bluetooth, and earlier generations of IEEE 802.11 WLANs. Spread spectrum approaches are mandated in many unlicensed spectrum allocations, like the ISM bands at 2.4 GHz and 5 GHz that are used for these technologies. This is because many users can share the same frequencies using spread spectrum with minimal impact on each other and without a need for any central control.

After a brief overview, we look at the two spread spectrum techniques. We then examine the code division multiple access technique that is based on spread spectrum.

The Concept of Spread Spectrum

Figure 5.27 highlights the key characteristics of any spread spectrum system. Input is fed into a channel encoder that produces an analog signal with a relatively narrow bandwidth around some center frequency. This signal is further modulated using

[6]Spread spectrum (using frequency hopping) was invented, believe it or not, by Hollywood screen siren Hedy Lamarr in 1940 at the age of 26. She and a partner who later joined her effort were granted a patent in 1942 (U.S. Patent 2,292,387; August 11, 1942). Lamarr considered this her contribution to the war effort and never profited from her invention.

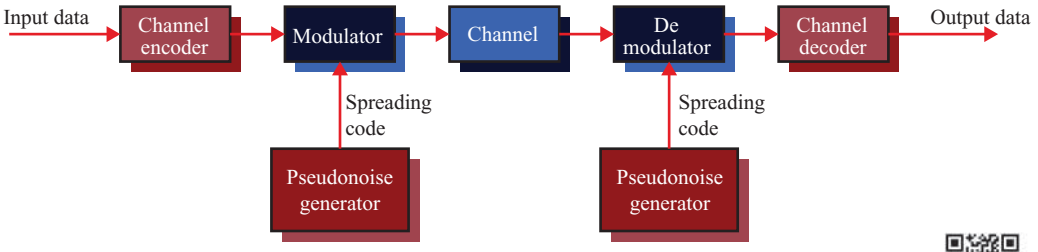


Figure 5.27 General Model of Spread Spectrum Digital Communication System

a sequence of digits known as a spreading code or spreading sequence. Typically, but not always, the spreading code is generated by a pseudonoise, or pseudorandom number, generator. The effect of this modulation is to increase significantly the bandwidth (spread the spectrum) of the signal to be transmitted. On the receiving end, the same digit sequence is used to demodulate the spread spectrum signal. Finally, the signal is fed into a channel decoder to recover the data.

Several things can be gained from this apparent waste of spectrum:

- The signals gains immunity from various kinds of noise and multipath distortion. The earliest applications of spread spectrum were military, where it was used for its immunity to jamming.
- It can also be used for hiding and encrypting signals. Only a recipient who knows the spreading code can recover the encoded information.
- Several users can independently use the same higher bandwidth with very little interference. This property is used in cellular telephony applications with a technique known as code division multiplexing (CDM) or code division multiple access (CDMA).

Frequency Hopping Spread Spectrum

With **frequency hopping spread spectrum** (FHSS), the signal is broadcast over a seemingly random series of radio frequencies, hopping from frequency to frequency at fixed intervals. A receiver, hopping between frequencies in synchronization with the transmitter, picks up the message. Would-be eavesdroppers hear only unintelligible blips. Attempts to jam the signal on one frequency succeed only at knocking out a few bits of it.

Figure 5.28 shows an example of a frequency hopping (FH) signal. A number of channels, C, are allocated for the FH signal. For example, IEEE 802.15.1, Bluetooth, uses $C = 80$. The spacing between carrier frequencies and hence the width of each channel usually corresponds to the bandwidth of the input signal. The transmitter operates in one channel at a time for a fixed interval; for example, the IEEE 802.15.1 Bluetooth standard uses a 0.625-ms interval. During that interval, some number of bits (possibly a fraction of a bit, as discussed subsequently) is transmitted using some encoding scheme. A spreading code dictates the sequence of channels used. Both the transmitter and receiver use the same code to tune into a sequence of channels in synchronization.

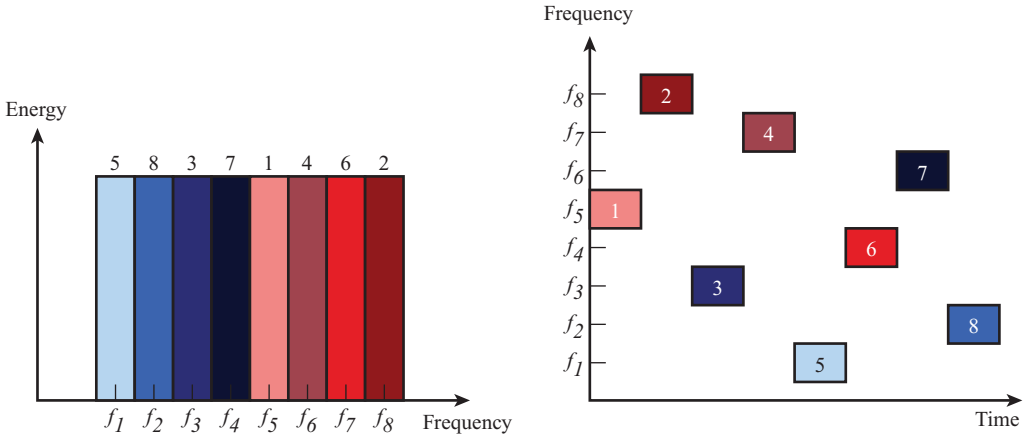


Figure 5.28 Frequency Hopping Example

A typical block diagram for a frequency hopping system is shown in Figure 5.29. For transmission, binary data are fed into a modulator using some digital-to-analog encoding scheme, such as frequency-shift keying (FSK) or binary phase-shift keying (BPSK). The resulting signal $s_d(t)$ is centered on some base frequency. A pseudo-noise (PN), or pseudorandom number, source serves as an index into a table of frequencies; this is the spreading code referred to previously. At each successive interval (each k PN bits), a new carrier frequency $c(t)$ is selected. This frequency is then modulated by the signal produced from the initial modulator to produce a new signal $s(t)$ with the same shape but now centered on the selected carrier frequency. On reception, the spread spectrum signal is demodulated using the same sequence of PN-derived frequencies and then demodulated to produce the output data.

FHSS Performance Considerations Typically, a large number of frequencies are used in FHSS so that the total FHSS bandwidth W_s is much larger than the bandwidth of each individual channel, W_d. One benefit of this is that a large value of k results in a system that is quite resistant to noise and jamming. For example, suppose we have a transmitter with bandwidth W_d and noise jammer of the same bandwidth and fixed power S_j on the signal carrier frequency. Then we have a ratio of signal energy per bit to jammer interference power density per hertz of

$$\frac{E_b}{I_j} = \frac{E_b}{S_j/W_d} = \frac{E_b W_d}{S_j}$$

If frequency hopping is used, the jammer must jam all C frequencies. With a fixed power, this reduces the jamming power in any one frequency band to S_j/C. The gain in signal-to-noise ratio, or processing gain, is

$$G_P = C = \frac{W_s}{W_d} \tag{5.13}$$

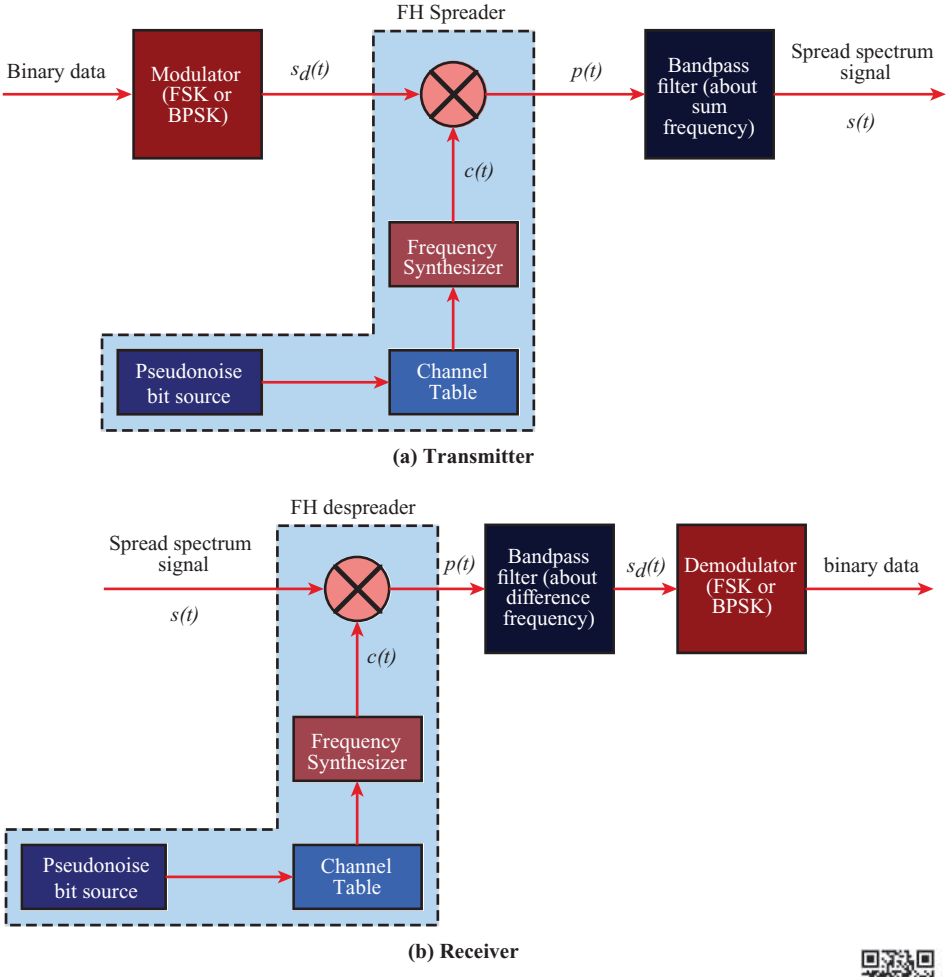


Figure 5.29 Frequency Hopping Spread Spectrum System

Direct Sequence Spread Spectrum

For **direct sequence spread spectrum** (DSSS), each bit in the original signal is represented by multiple bits in the transmitted signal, using a spreading code. The spreading code spreads the signal across a wider frequency band in direct proportion to the number of bits used. Therefore, a 10-bit spreading code spreads the signal across a frequency band that is 10 times greater than a 1-bit spreading code. Since the bits in the PN sequence are much smaller, they are sometimes called *chips*; the sequence is then called the *chip sequence*.

One technique for direct sequence spread spectrum is to combine the digital information stream with the spreading code bit stream using an exclusive-OR (XOR). The XOR obeys the following rules:

$$0 \oplus 0 = 0 \quad 0 \oplus 1 = 1 \quad 1 \oplus 0 = 1 \quad 1 \oplus 1 = 0$$

Figure 5.30 shows an example. Note that an information bit of one inverts the spreading code bits in the combination, while an information bit of zero causes the spreading code bits to be transmitted without inversion. The combination bit stream has the data rate of the original spreading code sequence, so it has a wider bandwidth than the information stream. In this example, the spreading code bit stream is clocked at four times the information rate. Figure 5.31 shows the implementation of this approach.

DSSS Performance Considerations The spectrum spreading achieved by the direct sequence technique is easily determined (Figure 5.32). In our example, the information signal has a bit width of T, which is equivalent to a data rate of $1/T$. In that case, the spectrum of the signal, depending on the encoding technique, is roughly $2/T$. Similarly, the spectrum of the PN signal is $2/T_c$. Figure 5.32c shows the resulting spectrum spreading. The total spectrum is $2/T_c + 2/T$, which is approximately $2/T_c$, since $2/T$ is small in comparison. One technology uses 128 chips per symbol, so $T/T_c = 128$. The amount of spreading that is achieved is a direct result of the data rate of the PN stream.

As with FHSS, we can get some insight into the performance of DSSS by looking at its effectiveness against jamming. The jamming power is reduced by a factor of (T_c/T) through the use of spread spectrum. The inverse of this factor is the gain in signal-to-noise ratio for the transmitted signal:

$$G_P = \frac{T}{T_c} = \frac{R_c}{R} \approx \frac{W_s}{W_d} \tag{5.14}$$

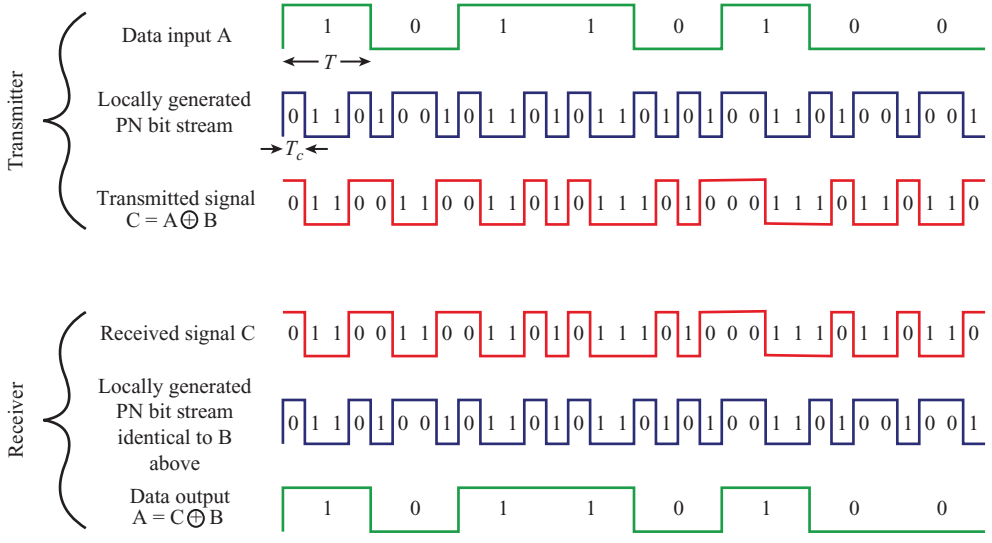


Figure 5.30 Example of Direct Sequence Spread Spectrum

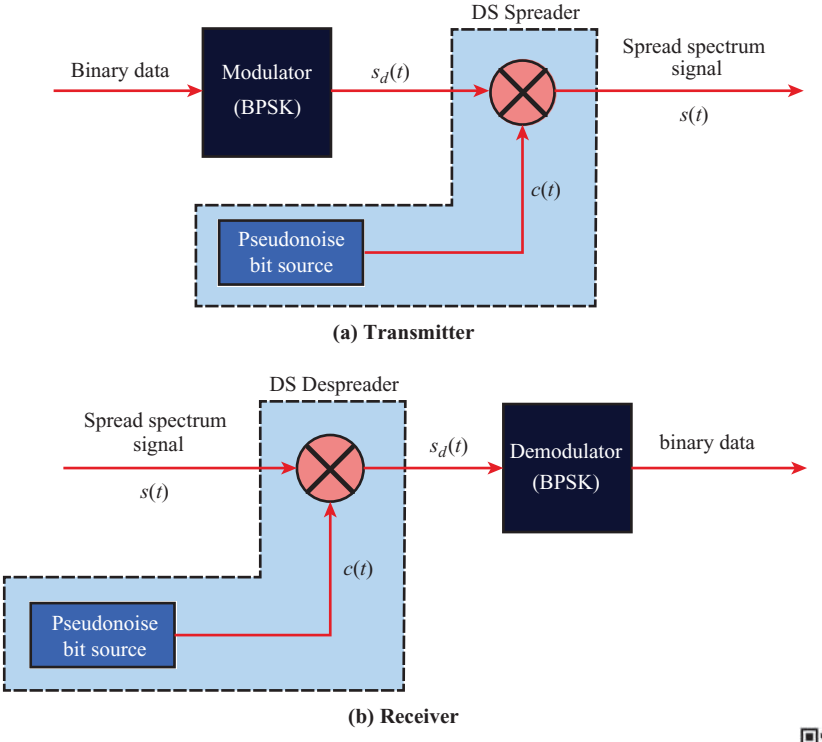


Figure 5.31 Direct Sequence Spread Spectrum System

where R_c is the spreading bit rate, R is the data rate, W_d is the signal bandwidth, and W_s is the spread spectrum signal bandwidth.

Code Division Multiple Access

CDMA is a multiplexing technique used with spread spectrum to share the wireless medium. It simply uses different PN sequences for different users. For frequency hopping the result is easy to understand, since the data can only be retrieved if the receiver knows the PN hopping sequence.

For DSSS, consider again Figure 5.31. The receiver will detect a signal that has a combination of many users utilizing different code. Using ideal codes (called "orthogonal codes"), only the desired signal will be retrieved after accomplishing the despreading multiplication operation. All other signals will be canceled to zero. Such codes are very nice to have but there are not all that many of them. Therefore, practical systems will use codes that are non-ideal. The result out of the despreader will be the desired signal but with what looks like a level of noise from the other users.

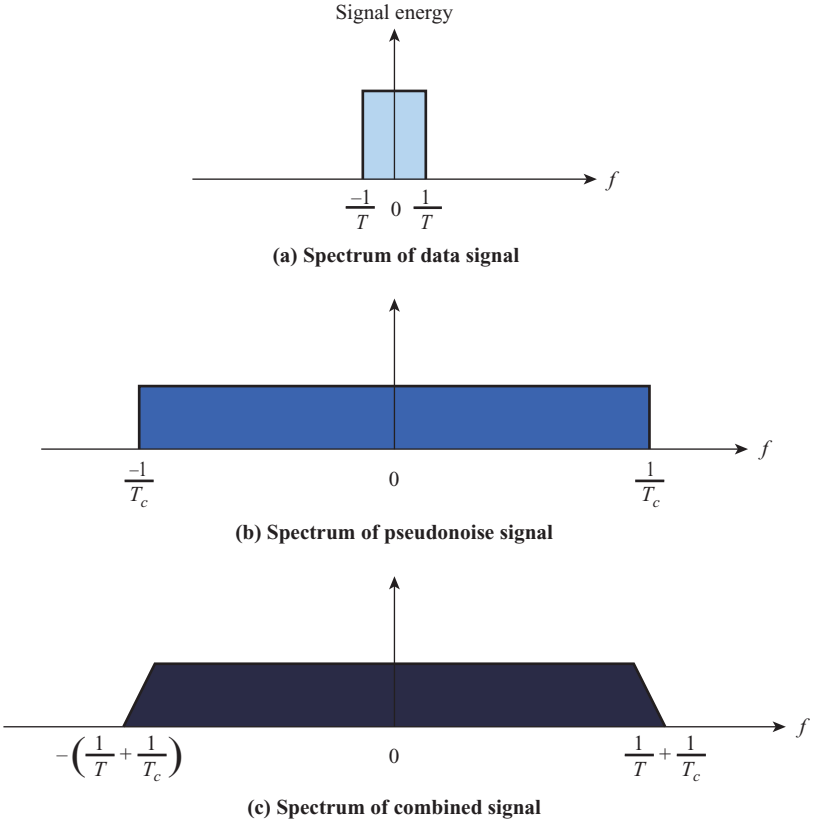


Figure 5.32 Approximate Spectrum of Direct Sequence Spread Spectrum Signal

In practice, the CDMA receiver can filter out the contribution from unwanted users or they appear as low-level noise. However, if there are many users competing for the channel with the user the receiver is trying to listen to, or if the signal power of one or more competing signals is too high, perhaps because it is very near the receiver (the "near/far" problem), the system breaks down. Many CDMA systems use tight power control of transmitting devices so none of the competing signals can be too strong.

The limit on the number of users in the system is derived from this understanding. A common measurement for a CDMA system is the **rise-over-thermal**, which compares the total contribution of this noise from the users in the system to the background thermal noise of the environment. One system uses 7 dB rise-over-thermal as a key performance metric to limit additional users.

5.9 RECOMMENDED READING

For references on specific topics, please consult the chapters that cover those topics in more detail.

5.10 KEY TERMS, REVIEW QUESTIONS, AND PROBLEMS

Key Terms

adaptive modulation and coding (AMC)	flat fading	multi-user MIMO (MU-MIMO)
amplitude-shift keying (ASK)	forward error correction (FEC)	orthogonal frequency division multiple access (OFDMA)
antenna	free space loss	parity check
antenna gain	frequency hopping spread spectrum	phase-shift keying (PSK)
atmospheric absorption	frequency selective fading	picocells
attenuation	frequency-shift keying (FSK)	radiation pattern
code division multiple access (CDMA)	ground wave propagation	reflections
code rate	Hamming distance	refraction
codeword	hybrid automatic repeat request (H-ARQ)	scattering
coding gain	isotropic antenna	signal-to-noise ratio (SNR)
cyclic redundancy check (CRC)	large-scale fading	small cells
diffraction	line of sight (LOS)	small-scale fading
direct sequence spread spectrum	mmWave	sky wave propagation
diversity	multiple-input multiple-output (MIMO)	slow fading
Doppler spread	multipath	spectrum
error detection	noise	spread spectrum
fading	orthogonal frequency division multiplexing (OFDM)	subcarriers
fast fading		thermal noise
femtocells		turbo coding

Review Questions

5.1 What is an isotropic antenna?
5.2 What information is available from a radiation pattern?
5.3 What is fading?
5.4 What is the difference between diffraction and scattering?
5.5 What is the difference between fast and slow fading?
5.6 What is the difference between flat and selective fading?
5.7 Indicate three major advantages of digital transmission over analog transmission.
5.8 How are binary values represented in amplitude-shift keying?
5.9 What is QAM?
5.10 What is a parity bit?
5.11 What is the CRC?
5.12 Why would you expect a CRC to detect more errors than a parity bit?
5.13 What two key elements comprise error control?
5.14 Explain how Go-back-N ARQ works.
5.15 Briefly define OFDM, OFDMA, and SC-FDMA.
5.16 What is the relationship between the bandwidth of a signal before and after it has been encoded using spread spectrum?
5.17 List three benefits of spread spectrum.

5.18 What is frequency hopping spread spectrum?

5.19 What is direct sequence spread spectrum?

5.20 What is CDMA?

Problems

5.1 For radio transmission in free space, signal power is reduced in proportion to the square of the distance from the source, whereas in wire transmission, the attenuation is a fixed number of dB per kilometer. The following table is used to show the dB reduction relative to some reference for free-space radio and uniform wire. Fill in the missing numbers to complete the table.

Distance (km)	Radio (dB)	Wire (dB)
1	−6	−3
2		
4		
8		
16		

5.2 It turns out that the depth in the ocean to which airborne electromagnetic signals can be detected grows with the wavelength. Therefore, the military got the idea of using very long wavelengths corresponding to about 30 Hz to communicate with submarines throughout the world. If we want to have an antenna that is about one-half wavelength long, how long would that be?

5.3 The audio power of the human voice is concentrated at about 300 Hz. Antennas of the appropriate size for this frequency are impracticably large, so that to send voice by radio the voice signal must be used to modulate a higher (carrier) frequency for which the natural antenna size is smaller.

a. What is the length of an antenna one-half wavelength long for sending radio at 300 Hz?

b. An alternative is to use a modulation scheme for transmitting the voice signal by modulating a carrier frequency, so that the bandwidth of the signal is a narrowband centered on the carrier frequency. Suppose we would like a half-wave antenna to have a length of 1 m. What carrier frequency would we use?

5.4 Stories abound of people who receive radio signals in fillings in their teeth. Suppose you have one filling that is 2.5 mm (0.0025 m) long that acts as a radio antenna. That is, it is equal in length to one-half the wavelength. What frequency do you receive?

5.5 It is often more convenient to express distance in km rather than m and frequency in MHz rather than Hz. Rewrite Equation (5.1) using these dimensions.

5.6 Suppose a transmitter produces 50 W of power.

a. Express the transmit power in units of dBm and dBW.

b. If the transmitter's power is applied to a unity gain antenna with a 900 MHz carrier frequency, what is the received power in dBm at a free space distance of 100 m?

c. Repeat (b) for a distance of 10 km.

d. Repeat (c) but assume a receiver antenna gain of 2.

5.7 Show that doubling the transmission frequency or doubling the distance between transmitting antenna and receiving antenna attenuates the power received by 6 dB.

5.8 What is the purpose of using modulo 2 arithmetic rather than binary arithmetic in computing an FCS?

5.9 Consider a frame consisting of two characters of four bits each. Assume that the probability of bit error is 10^{-3} and that it is independent for each bit.

a. What is the probability that the received frame contains at least one error?
b. Now add a parity bit to each character. What is the probability?

5.10 Using the CRC-CCITT polynomial, generate the 16-bit CRC code for a message consisting of a 1 followed by 15 0s. Use long division.

5.11 For $P = 110011$ and $M = 11100011$, find the CRC.

5.12 Calculate the pairwise Hamming distances among the following codewords:
a. 00000, 10101, 01010
b. 000000, 010101, 101010, 110110

5.13 For a given positive integer t, if a code satisfies $d_{min} \geq 2t + 1$, then the code can correct all bit errors up to and including errors of t bits. Prove this assertion. *Hint:* Start by observing that for a codeword **w** to be decoded as another codeword **w'**, the received sequence must be at least as close to **w'** as to **w**.

5.14 The simplest form of flow control, known as **stop-and-wait flow control**, works as follows. A source entity transmits a frame. After the destination entity receives the frame, it indicates its willingness to accept another frame by sending back an acknowledgment to the frame just received. The source must wait until it receives the acknowledgment before sending the next frame. The destination can thus stop the flow of data simply by withholding acknowledgment. Consider a half-duplex point-to-point link using a stop-and-wait scheme, in which a series of messages is sent, with each message segmented into a number of frames. Ignore errors and frame overhead.
a. What is the effect on line utilization of increasing the message size so that fewer messages will be required? Other factors remain constant.
b. What is the effect on line utilization of increasing the number of frames for a constant message size?
c. What is the effect on line utilization of increasing frame size?

5.15 For an 18 Mbps LTE data stream with a symbol time of 66.67 μs, how many subcarriers are created?

5.16 LTE assigns subcarriers in *resource blocks* of 180 kHz. Given the information in Problem 5.15, how many subcarriers are in a resource block? Approximate $B_S \approx r_b$.

5.17 The following table illustrates the operation of an FHSS system for one complete period of the PN sequence.

Time	0	1	2	3	4	5	6	7	8	9	10	11
Input data	0	1	1	1	1	1	1	0	0	0	1	0
Frequency	f_1		f_3		f_{23}		f_{22}		f_8		f_{10}	
PN sequence	001				110				011			

Time	12	13	14	15	16	17	18	19
Input data	0	1	1	1	1	0	1	0
Frequency	f_1		f_3		f_2		f_2	
PN sequence	001				001			

a. The system makes use of a form of FSK. What form of FSK is it?
b. What is the number of bits per signal element (symbol)?
c. What is the number of FSK frequencies?
d. What is the length of a PN sequence per hop?
e. Is this a slow or fast FH system?
f. What is the total number of possible carrier frequencies?
g. Show the variation of the base, or demodulated, frequency with time.