7.5 SPREAD SPECTRUM MODULATION

An increasingly important form of communications is known as spread spectrum. This technique does not fit neatly into the categories defined in the preceding chapter, as it can be used to transmit either analog or digital data, using an analog signal.

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.[3] A more recent type of spread spectrum is direct sequence. Both of these techniques are used in various wireless communications standards and products.

After a brief overview, we look at these two spread spectrum techniques. We then examine a multiple access technique based on spread spectrum.

Figure 7.20 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 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 pseudo-random 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:

- We can gain 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.

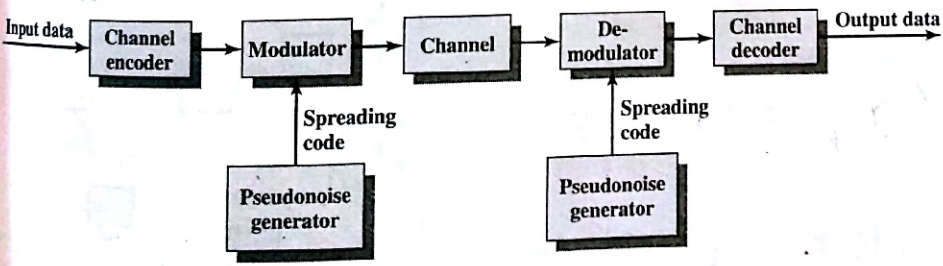


Figure 7.20 General Model of Spread Spectrum Digital Communication System

[3] 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; 11 August 1942). Lamarr considered this her contribution to the war effort and never profited from her invention.

- 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 know as code division multiplexing (CDM) or code division multiple access (CDMA).

7.6 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.

Basic Approach

Figure 7.21 shows an example of a frequency hopping signal. A number of channels are allocated for the FH signal. Typically, there are 2^k carrier frequencies forming 2^k channels. 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 802.11 wireless LAN standard uses a 300-ms interval. During that interval, some number of bits (possibly a fraction of a bit, as discussed subsequently) is transmitted using some encoding scheme. The sequence of channels used is dictated by a spreading code. Both transmitter and receiver use the same code to tune into a sequence of channels in synchronization.

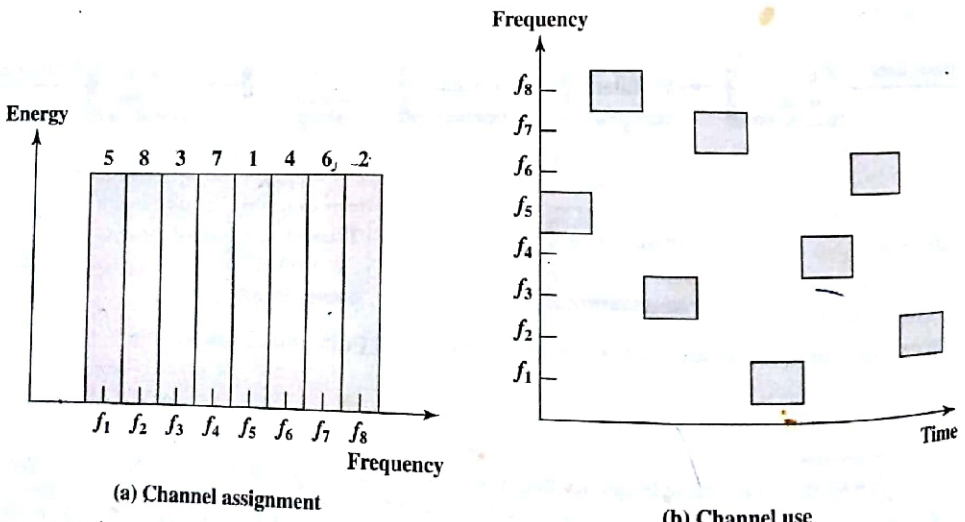


Figure 7.21 Frequency Hopping Example

A typical block diagram for a frequency hopping system is shown in Figure 7.22. 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 pseudonoise (PN), or pseudorandom number, source serves as an index into a table of frequencies; this is the spreading code referred to previously. Each k bits of the PN source specifies one of the 2^k carrier frequencies. 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.

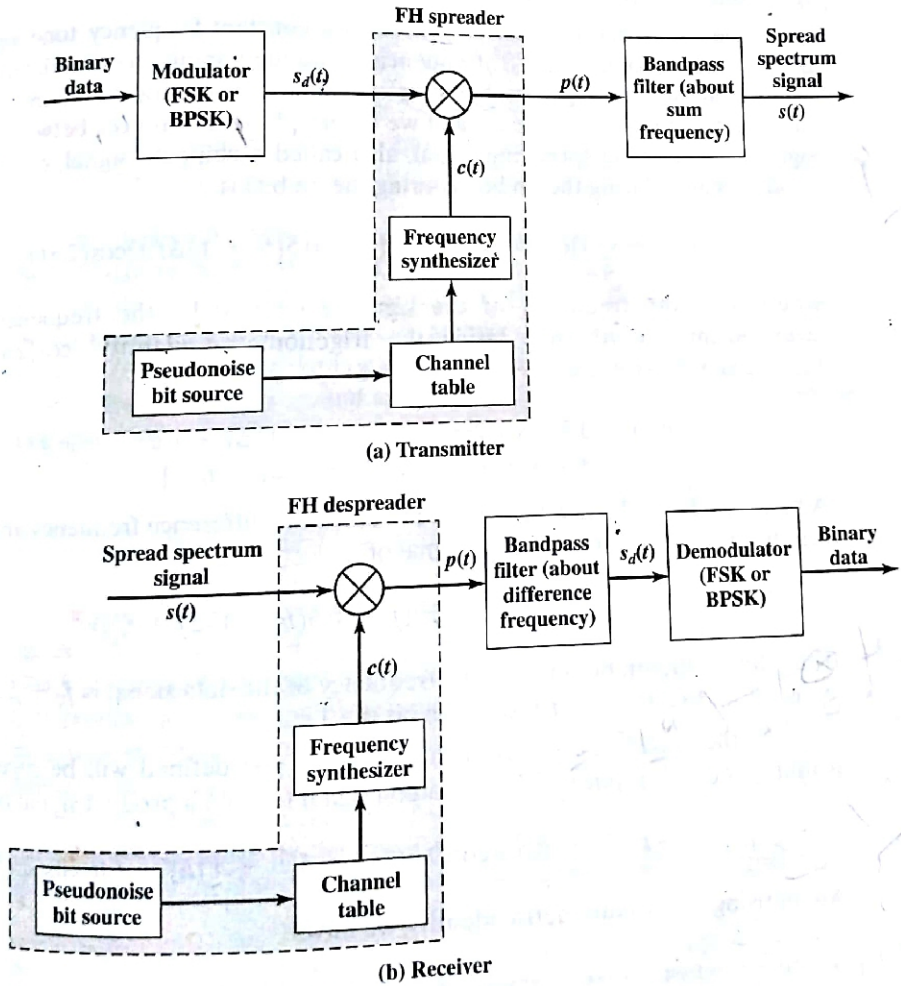


Figure 7.22 Frequency Hopping Spread Spectrum System

Figure 7.22 indicates that the two signals are multiplied. Let us give an example of how this works, using BFSK as the data modulation scheme. We can define the FSK input to the FHSS system as [compare to Equation (7.2)]:

$$s_d(t) = A\cos(2\pi(f_0 + 0.5(b_i + 1)\Delta f)t) \qquad \text{for } iT < t < (i+1)T \tag{7.16}$$

where

A = amplitude of signal
f_0 = base frequency
b_i = value of the ith bit of data (+1 for binary 1, −1 for binary 0)
Δf = frequency separation
T = bit duration; data rate = $1/T$

Thus, during the ith bit interval, the frequency of the data signal is f_0 if the data bit is −1 and $f_0 + \Delta f$ if the data bit is +1.

The frequency synthesizer generates a constant-frequency tone whose frequency hops among a set of 2^k frequencies, with the hopping pattern determined by k bits from the PN sequence. For simplicity, assume the duration of one hop is the same as the duration of one bit and we ignore phase differences between the data signal $s_d(t)$ and the spreading signal, also called a chipping signal, $c(t)$. Then the product signal during the ith hop (during the ith bit) is

$$p(t) = s_d(t)c(t) = A\cos(2\pi(f_0 + 0.5(b_i + 1)\Delta f)t)\cos(2\pi f_i t)$$

where f_i is the frequency of the signal generated by the frequency synthesizer during the ith hop. Using the trigonometric identity[4] $\cos(x)\cos(y) = (1/2)(\cos(x + y) + \cos(x - y))$, we have

$$\begin{aligned} p(t) = {} & 0.5A[\cos(2\pi(f_0 + 0.5(b_i + 1)\Delta f + f_i)t) \\ & + \cos(2\pi(f_0 + 0.5(b_i + 1)\Delta f - f_i)t)] \end{aligned}$$

A bandpass filter (Figure 7.22) is used to block the difference frequency and pass the sum frequency, yielding an FHSS signal of

$$s(t) = 0.5A\cos(2\pi(f_0 + 0.5(b_i + 1)\Delta f + f_i)t) \tag{7.17}$$

Thus, during the ith bit interval, the frequency of the data signal is $f_0 + f_i$ if the data bit is −1 and $f_0 + f_i + \Delta f$ if the data bit is +1.

At the receiver, a signal of the form $s(t)$ just defined will be received. This is multiplied by a replica of the spreading signal to yield a product signal of the form

$$p(t) = s(t)c(t) = 0.5A\cos(2\pi(f_0 + 0.5(b_i + 1)\Delta f + f_i)t)\cos(2\pi f_i t)$$

Again using the trigonometric identity, we have

[4] See the math refresher document at WilliamStallings.com/StudentSupport.html for a summary of trigonometric identities.

$$p(t) = s(t)c(t) = 0.25A[\cos(2\pi(f_0 + 0.5(b_i + 1)\Delta f + f_i + f_i)t) + \cos(2\pi(f_0 + 0.5(b_i + 1)\Delta f)t)]$$

A bandpass filter (Figure 7.22) is used to block the sum frequency and pass the difference frequency, yielding a signal of the form of $s_d(t)$, defined in Equation (7.16):

$$0.25A \cos(2\pi(f_0 + 0.5(b_i + 1)\Delta f)t)$$

FHSS Using MFSK

A common modulation technique used in conjunction with FHSS is multiple FSK (MFSK). Recall from section 7.2 that MFSK uses $M = 2^L$ different frequencies to encode the digital input L bits at a time. The transmitted signal is of the form [Equation (7.3)]:

$$s_i(t) = A \cos 2\pi f_i t, \qquad 1 \le i \le M$$

where

$f_i = f_c + (2i - 1 - M)f_d$
f_c = denotes the carrier frequency
f_d = denotes the difference frequency
M = number of different signal elements = 2^L
L = number of bits per signal element

For FHSS, the MFSK signal is translated to a new frequency every T_c seconds by modulating the MFSK signal with the FHSS carrier signal. The effect is to translate the MFSK signal into the appropriate FHSS channel. For a data rate of R, the duration of a bit is $T = 1/R$ seconds and the duration of a signal element is $T_s = LT$ seconds. If T_c is greater than or equal to T_s, the spreading modulation is referred to as **slow-frequency-hop spread spectrum**; otherwise it is known as **fast-frequency-hop spread spectrum**.[5] To summarize,

Slow-frequency-hop spread spectrum	$T_c \ge T_s$
Fast-frequency-hop spread spectrum	$T_c < T_s$

Figure 7.23 shows an example of slow FHSS, using the MFSK example from Figure 7.4. That is, $M = 4$, and the same sequence of input bits is used in both examples. The display in the figure shows the frequency transmitted (y-axis) as a function of time (x-axis). Each column represents a time unit T_s in which a single 2-bit signal element is transmitted. The shaded rectangle in the column indicates the frequency transmitted during that time unit. Each pair of columns corresponds to the selection of a frequency band based on a 2-bit PN sequence. Thus, for the first pair of columns, governed by PN sequence 00, the lowest band of frequencies is used. For the second pair of columns, governed by PN sequence 11, the highest band of frequencies is used.

[5] Some authors use a somewhat different definition (e.g., [PICK82]) of multiple hops per bit for fast frequency hop, multiple bits per hop for slow frequency hop, and one hop per bit if neither fast nor slow. The more common definition, which we use, relates hops to signal elements rather than bits.

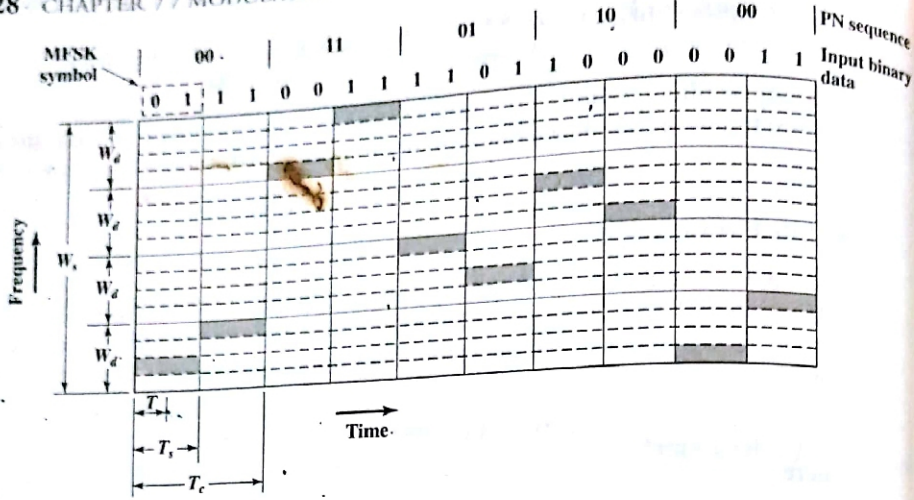


Figure 7.23 Slow-Frequency-Hop Spread Spectrum Using MFSK ($M = 4, k = 2$)

Here we have $M = 4$, which means that four different frequencies are used to encode the data input 2 bits at a time. Each signal element is a discrete frequency tone, and the total MFSK bandwidth is $W_d = Mf_d$. We use an FHSS scheme with $k = 2$. That is, there are $4 = 2^k$ different channels, each of width W_d. The total FHSS bandwidth is $W_s = 2^k W_d$. Each 2 bits of the PN sequence is used to select one of the four channels. That channel is held for a duration of two signal elements, or four bits ($T_c = 2T_s = 4T$).

Figure 7.24 shows an example of fast FHSS, using the same MFSK example. Again, $M = 4$ and $k = 2$. In this case, however, each signal element is represented by two frequency tones. Again, $W_d = Mf_d$ and $W_s = 2^k W_d$. In this example $T_s = 2T_c = 2T$. In general, fast FHSS provides improved performance compared to slow FHSS in the face of noise or jamming. For example, if 3 or more frequencies (chips) are used for each signal element, the receiver can decide which signal element was sent on the basis of a majority of the chips being correct.

FHSS Performance Considerations

Typically, a large number of frequencies are used in FHSS that W_s is much larger than 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 an MFSK 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 noise power density per Hertz of

$$\frac{E_b}{N_j} = \frac{E_b W_d}{S_j}$$

If frequency hopping is used, the jammer must jam all 2^k frequencies. With a fixed power, this reduces the jamming power in any one frequency band to $S_j/2^k$. The gain in signal-to-noise ratio, or processing gain, is

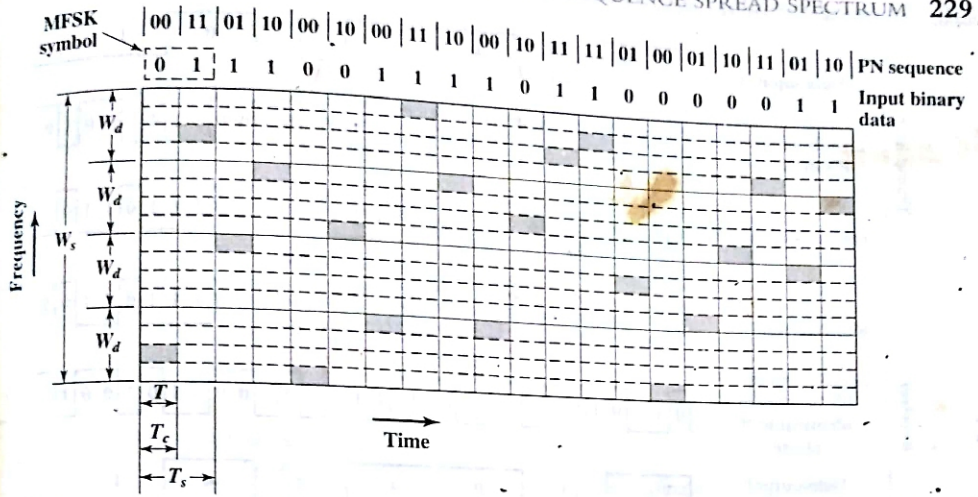


Figure 7.24 Fast-Frequency-Hop Spread Spectrum Using MFSK ($M = 4, k = 2$)

$$G_P = 2^k = \frac{W_s}{W_d} \tag{7.18}$$

7.7 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.

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:

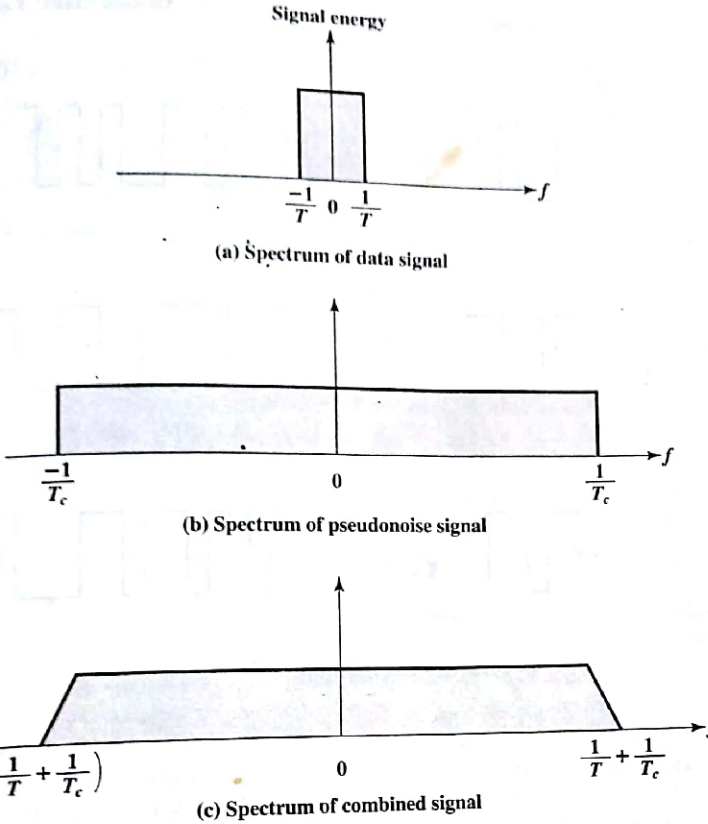
$$0 \oplus 0 = 0 \qquad 0 \oplus 1 = 1 \qquad 1 \oplus 0 = 1 \qquad 1 \oplus 1 = 0$$

Figure 7.25 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.

DSSS Using BPSK

To see how this technique works out in practice, assume that a BPSK modulation scheme is to be used. Rather than represent binary data with 1 and 0, it is more convenient for our purposes to use +1 and −1 to represent the two binary digits. In that case, a BPSK signal can be represented as was shown in Equation (7.5):

$$s_d(t) = Ad(t)\cos(2\pi f_c t) \tag{7.19}$$



(a) Spectrum of data signal

(b) Spectrum of pseudonoise signal

(c) Spectrum of combined signal

Figure 7.28 Approximate Spectrum of Direct Sequence Spread Spectrum Signal

CDMA

7.8 CODE DIVISION MULTIPLE ACCESS

Basic Principles CDMA is a multiplexing technique used with spread spectrum. The scheme works in the following manner. We start with a data signal with rate D, which we call the bit data rate. We break each bit into k *chips* according to a fixed pattern that is specific to each user, called the user's code. The new channel has a chip data rate of kD chips per second. As an illustration we consider a simple example[6] with $k = 6$. It is simplest to characterize a code as a sequence of 1s and -1s. Figure 7.29 shows the codes for three users, A, B, and C, each of which is communicating with the same base station receiver, R. Thus, the code for user A is $c_A = <1, -1, -1, 1, -1, 1>$. Similarly, user B has code $c_B = <1, 1, -1, -1, 1, 1>$, and user C has $c_C = <1, 1, -1, 1, 1, -1>$.

We now consider the case of user A communicating with the base station. The base station is assumed to know A's code. For simplicity, we assume that communi-

[6]This example was provided by Prof. Richard Van Slyke of the Polytechnic University of Brooklyn.

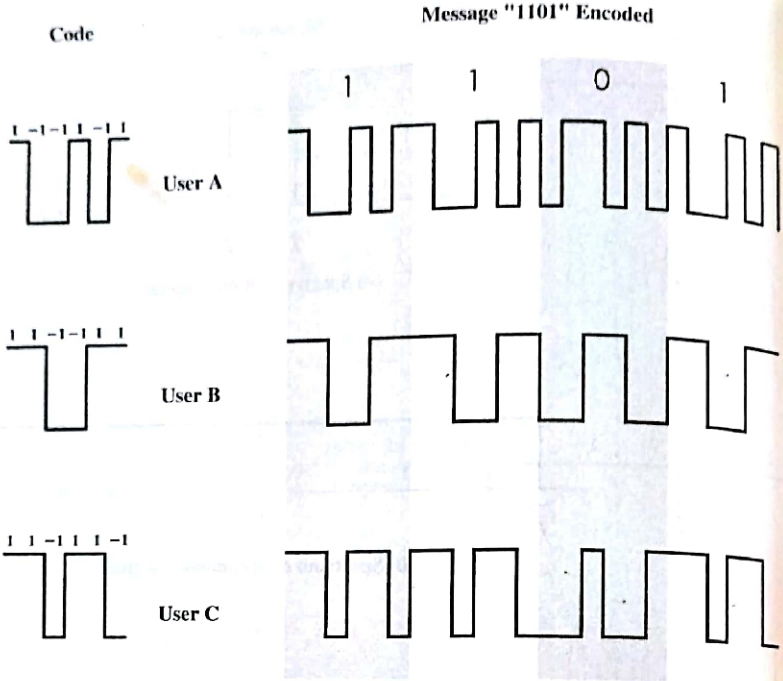


Figure 7.29 CDMA Example

cation is already synchronized so that the base station knows when to look for codes. If A wants to send a 1 bit, A transmits its code as a chip pattern <1, −1, −1, 1, −1, 1>. If a 0 bit is to be sent, A transmits the complement (1s and −1s reversed) of its code, <−1, 1, 1, −1, 1, −1>. At the base station the receiver decodes the chip patterns. In our simple version, if the receiver R receives a chip pattern $d = <d1, d2, d3, d4, d5, d6>$, and the receiver is seeking to communicate with a user u so that it has at hand u's code, $<c1, c2, c3, c4, c5, c6>$, the receiver performs electronically the following decoding function:

$$S_u(d) = (d1 \times c1) + (d2 \times c2) + (d3 \times c3) + (d4 \times c4) + (d5 \times c5) + (d6 \times c6)$$

The subscript u on S simply indicates that u is the user that we are interested in. Let's suppose the user u is actually A and see what happens. If A sends a 1 bit, then d is <1, −1, −1, 1, −1, 1> and the preceding computation using S_A becomes

$$S_A(1, -1, -1, 1, -1, 1) = [1 \times 1] + [(-1) \times (-1)] + [(-1) \times (-1)] + [(-1) \times (-1)] + [1 \times 1] = 6$$

If A sends a 0 bit that corresponds to $d = <-1, 1, 1, -1, 1, -1>$, we get

$$S_A(-1, 1, 1, -1, 1, -1) = [-1 \times 1] + [1 \times (-1)] + [1 \times (-1)] + [(-1) \times 1] + [1 \times (-1)] + [1 \times (-1)] + [(-1) \times 1] = -6$$

Please note that it is always the case that $-6 \leq S_A(d) \leq 6$ no matter what sequence of −1s and 1s comprise d, and that the only values of d resulting in the

extreme values of 6 and −6 are A's code and its complement, respectively. So if S_A produces a +6, we say that we have received a 1 bit from A; if S_A produces a −6, we say that we have received a 0 bit from user A; otherwise, we assume that someone else is sending information or there is an error. So why go through all this? The reason becomes clear if we see what happens if user B is sending and we try to receive it with S_A, that is, we are decoding with the wrong code, A's. If B sends a 1 bit, then $d = <1, 1, -1, -1, 1, 1>$. Then

$$S_A(1, 1, -1, -1, 1, 1) = [1 \times 1] + [1 \times (-1)] + [(-1) \times (-1)] + [(-1) \times 1] + [1 \times (-1)] + [1 \times 1] = 0$$

Thus, the unwanted signal (from B) does not show up at all. You can easily verify that if B had sent a 0 bit, the decoder would produce a value of 0 for S_A again. This means that if the decoder is linear and if A and B transmit signals S_A and S_B, respectively, at the same time, then $S_A(s_A + s_B) = S_A(s_A) + S_A(s_B) = S_A(s_A)$ since the decoder ignores B when it is using A's code. The codes of A and B that have the property that $S_A(c_B) = S_B(c_A) = 0$ are called *orthogonal*. Such codes are very nice to have but there are not all that many of them. More common is the case when $S_X(c_Y)$ is small in absolute value when $X \times Y$. Then it is easy to distinguish between the two cases when $X = Y$ and when $X \neq Y$. In our example $S_A(c_C) = S_c(c_A) = 0$, but $S_B(c_C) = S_C(c_B) = 2$. In the latter case the C signal would make a small contribution to the decoded signal instead of 0. Using the decoder, S_u, the receiver can sort out transmission from u even when there may be other users broadcasting in the same cell.

Table 7.3 summarizes the example from the preceding discussion.

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.

CDMA for Direct Sequence Spread Spectrum

Let us now look at CDMA from the viewpoint of a DSSS system using BPSK. Figure 7.30 depicts a configuration in which there are n users, each transmitting using a different, orthogonal, PN sequence (compare Figure 7.26). For each user, the data stream to be transmitted, $d_i(t)$, is BPSK modulated to produce a signal with a bandwidth of W_s and then multiplied by the spreading code for that user, $c_i(t)$. All of the signals, plus noise, are received at the receiver's antenna. Suppose that the receiver is attempting to recover the data of user 1. The incoming signal is multiplied by the spreading code of user 1 and then demodulated. The effect of this is to narrow the bandwidth of that portion of the incoming signal corresponding to user 1 to the original bandwidth of the unspread signal, which is proportional to the data rate. Because the remainder of the incoming signal is orthogonal to the spreading code of user 1, that remainder still has the bandwidth W_s. Thus the unwanted signal energy remains spread over a large bandwidth and the wanted signal is concentrated in a narrow bandwidth. The bandpass filter at the demodulator can therefore recover the desired signal.

The most prominent specification for wireless LANs (WLANs) was developed by the IEEE 802.11 working group. We look first at the overall architecture of IEEE 802 standards and then at the specifics of IEEE 802.11.

14.1 IEEE 802 ARCHITECTURE

The architecture of a LAN is best described in terms of a layering of protocols that organize the basic functions of a LAN. This section opens with a description of the standardized protocol architecture for LANs, which encompasses physical, medium access control, and logical link control layers. We then look in more detail at medium access control and logical link control.

Protocol Architecture

Protocols defined specifically for LAN and MAN (metropolitan area network) transmission address issues relating to the transmission of blocks of data over the network. In OSI terms, higher-layer protocols (layer 3 or 4 and above) are independent of network architecture and are applicable to LANs, MANs, and WANs. Thus, a discussion of LAN protocols is concerned principally with lower layers of the OSI model.

Figure 14.1 relates the LAN protocols to the OSI architecture (Figure 4.3). This architecture was developed by the IEEE 802 committee and has been adopted by all organizations working on the specification of LAN standards. It is generally referred to as the IEEE 802 reference model.[1]

Working from the bottom up, the lowest layer of the IEEE 802 reference model corresponds to the **physical layer** of the OSI model and includes such functions as

- Encoding/decoding of signals (e.g., PSK, QAM, etc.)
- Preamble generation/removal (for synchronization)
- Bit transmission/reception

In addition, the physical layer of the 802 model includes a specification of the transmission medium and the topology. Generally, this is considered "below" the lowest layer of the OSI model. However, the choice of transmission medium and topology is critical in LAN design, and so a specification of the medium is included. For some of the IEEE 802 standards, the physical layer is further subdivided into sublayers. In the case of IEEE 802.11, two sublayers are defined:

- **Physical layer convergence procedure (PLCP):** Defines a method of mapping 802.11 MAC layer protocol data units (MPDUs) into a framing format suitable for sending and receiving user data and management information between two or more stations using the associated PMD sublayer
- **Physical medium dependent sublayer (PMD):** Defines the characteristics of, and method of transmitting and receiving, user data through a wireless medium between two or more stations

[1] A supporting document at this book's Web site provides an overview of the key organizations involved in developing communication and protocol standards, including the IEEE 802 Standards Committee.