Pearson New International Edition

Digital Communications
Fundamentals and Applications

Bernard Sklar

Second Edition

ALWAYS LEARNING™

PEARSON®

Pearson New International Edition

Digital Communications
Fundamentals and Applications
Bernard Sklar
Second Edition

PEARSON®

The biorthogonal set is really two sets of orthogonal codes such that each codeword in one set has its antipodal codeword in the other set. The biorthogonal set consists of a *combination of orthogonal and antipodal* signals. With respect to z_{ij} of Equation (6.1), biorthogonal codes can be characterized as

$$z_{ij} = \begin{cases} 1 & \text{for } i = j \\ -1 & \text{for } i \neq j,\ |i - j| = \dfrac{M}{2} \\ 0 & \text{for } i \neq j,\ |i - j| \neq \dfrac{M}{2} \end{cases} \tag{6.8}$$

One advantage of a biorthogonal code over an orthogonal one for the same data set is that the biorthogonal code requires *one-half* as many code bits per codeword (compare the columns of the $\mathbf{B}_3$ matrix with those of the $\mathbf{H}_3$ matrix presented earlier). Thus the bandwidth requirements for biorthogonal codes are one-half the requirements for comparable orthogonal ones. Since antipodal signal vectors have better distance properties than orthogonal ones, it should come as no surprise that biorthogonal codes perform slightly better than orthogonal ones. For equally likely, equal-energy biorthogonal signals, the probability of codeword (symbol) error can be upper bounded, as follows [2]:

$$P_E(M) \leq (M - 2)Q\left(\sqrt{\frac{E_s}{N_0}}\right) + Q\left(\sqrt{\frac{2E_s}{N_0}}\right) \tag{6.9}$$

which becomes increasingly tight for fixed M as E_b/N_0 is increased. $P_B(M)$ is a complicated function of $P_E(M)$; we can approximate it with the relationship [2]

$$P_B(M) \approx \frac{P_E(M)}{2}$$

The approximation is quite good for $M > 8$. Therefore, we can write

$$P_B(M) \lesssim \tfrac{1}{2}\left[(M - 2)Q\left(\sqrt{\frac{E_s}{N_0}}\right) + Q\left(\sqrt{\frac{2E_s}{N_0}}\right)\right] \tag{6.10}$$

These biorthogonal codes offer improved P_B performance, compared with the performance of the orthogonal codes, and require only *half the bandwidth* of orthogonal codes.

6.1.3.3 Transorthogonal (Simplex) Codes

A code generated from an orthogonal set by deleting the first digit of each codeword is called a *transorthogonal* or *simplex code.* Such a code is characterized by

$$z_{ij} = \begin{cases} 1 & \text{for } i = j \\ \dfrac{-1}{M - 1} & \text{for } i \neq j \end{cases} \tag{6.11}$$

A simplex code represents the *minimum energy* equivalent (in the error-probability sense) of the equally likely orthogonal set. In comparing the error performance of orthogonal, biorthogonal, and simplex codes, we can state that simplex coding requires the minimum E_b/N_0 for a specified symbol error rate. However, for a *large value of M,* all three schemes are *essentially identical* in error performance. Biorthogonal coding requires half the bandwidth of the others. But for each of these codes, bandwidth requirements (and system complexity) grow exponentially with the value of *M;* therefore, such coding schemes are attractive only when large bandwidths are available.

6.1.4 Waveform-Coding System Example

Figure 6.4 illustrates an example of assigning a k-bit message from a message set of size $M = 2^k$, with a coded-pulse sequence from a code set of the same size. Each k-bit message chooses one of the generators yielding a coded-pulse sequence or codeword. The sequences in the coded set that replace the messages form a waveform set with good distance properties (e.g., orthogonal, biorthogonal). For the orthogonal code described in Section 6.1.3.1, each codeword consists of $M = 2^k$ pulses (representing code bits). Hence 2^k code bits replace k message bits. The chosen sequence then modulates a carrier wave using binary PSK, such that the phase ($\phi_j = 0$ or π) of the carrier during each code-bit duration, $0 \le t \le T_c$, corresponds to the amplitude ($j = -1$ or 1) of the jth bipolar pulse in the codeword. At the receiver in Figure 6.5, the signal is demodulated to baseband and fed to M correlators (or matched filters). For orthogonal codes, such as those characterized by the Hadamard matrix in Section 6.1.3.1, correlation is performed over a codeword duration that can be expressed as $T = 2^k T_c$. For a real-time communication system, messages may not be delayed; hence, the codeword duration must be the same as the message duration, and thus, T can also be expressed as $T = (\log_2 M)\, T_b = kT_b$,

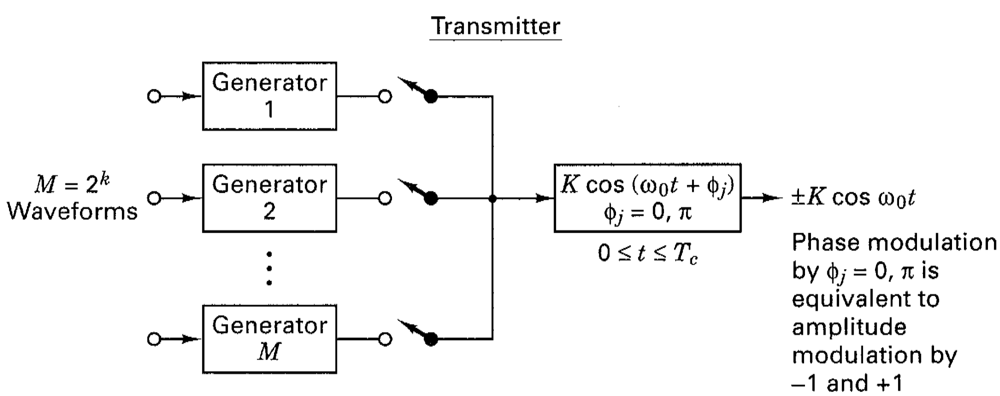

Figure 6.4 Waveform-encoded system (transmitter).

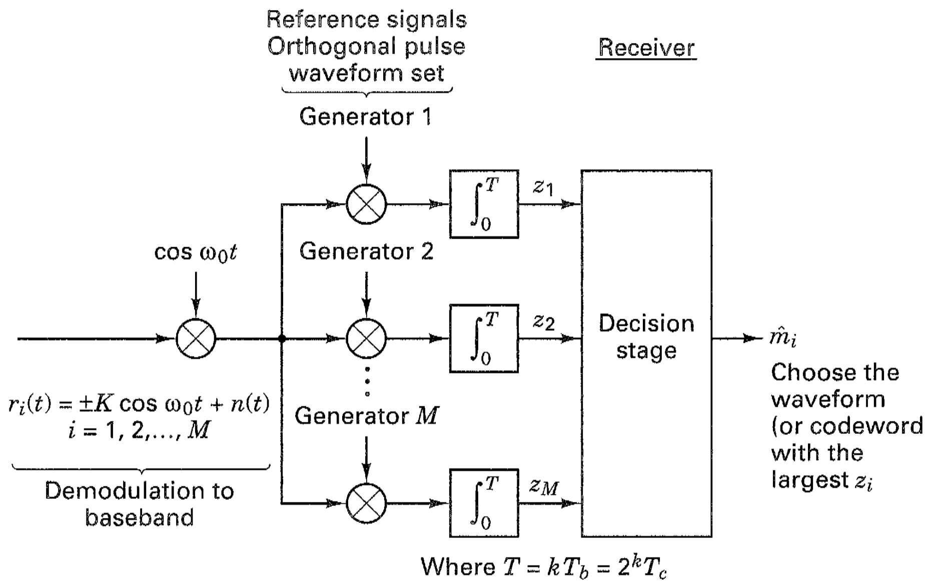

Figure 6.5 Waveform-encoded system with coherent detection (receiver).

where T_b is the message-bit duration. Note that the time duration of a message bit is M/k times longer than that of a code bit. In other words, the code bits or coded pulses (which are PSK modulated) must move at a rate M/k faster than the message bits. For such orthogonally coded waveforms and an AWGN channel, the expected value at the output of each correlator, at time T, is zero, except for the correlator corresponding to the transmitted codeword.

What is the advantage of such orthogonal waveform coding compared with simply sending one bit or one pulse at a time? One can compare the bit-error performance with and without such coding by comparing Equation (4.79) for coherent detection of antipodal signals with Equation (6.7) for the coherent detection of orthogonal codewords. For a given size k-bit message (say, $k = 5$) and a desired bit-error probability (say, 10^{-5}), the detection of orthogonal codewords (each having a 5-bit meaning) can be accomplished with about 2.9 dB less E_b/N_0 than the bit-by-bit detection of antipodal signals. (The demonstration is left as an exercise for the reader in Problem 6.28.) One might have guessed this result by comparing the performance curves for orthogonal signaling in Figure 4.28 with the binary (antipodal) curve in Figure 4.29. What price do we pay for this error-performance improvement? The cost is more transmission bandwidth. In this example, transmission of an uncoded message consists of sending 5 bits. With coding, how many coded pulses must be transmitted for each message sequence? With the waveform coding of this example, each 5-bit message sequence is represented by $M = 2^k = 2^5 = 32$ code bits or coded pulses. The 32 coded pulses in a codeword must be sent in the same time duration as the corresponding 5 bits from which they stem. Thus, the required transmission bandwidth is 32/5 times that of the uncoded case. In general, the bandwidth needed for such orthogonally coded signals is M/k times greater than that needed for the uncoded case. Later, more efficient ways to trade off the benefits of coding versus bandwidth [3, 4] will be examined.

6.2 TYPES OF ERROR CONTROL

Before we discuss the details of structured redundancy, let us describe the two basic ways such redundancy is used for controlling errors. The first, *error detection and retransmission,* utilizes *parity bits* (redundant bits added to the data) to detect that an error has been made. The receiving terminal does not attempt to correct the error; it simply requests that the transmitter retransmit the data. Notice that a two-way link is required for such dialogue between the transmitter and receiver. The second type of error control, *forward error correction* (FEC), requires a one-way link only, since in this case the parity bits are designed for both the detection and correction of errors. We shall see that not all error patterns can be corrected; error-correcting codes are classified according to their error-correcting capabilities.

6.2.1 Terminal Connectivity

Communication terminals are often classified according to their connectivity with other terminals. The possible connections, shown in Figure 6.6, are termed *simplex* (not to be confused with the simplex or transorthogonal codes), *half-duplex,* and *full-duplex.* The simplex connection in Figure 6.6a is a one-way link. Transmissions

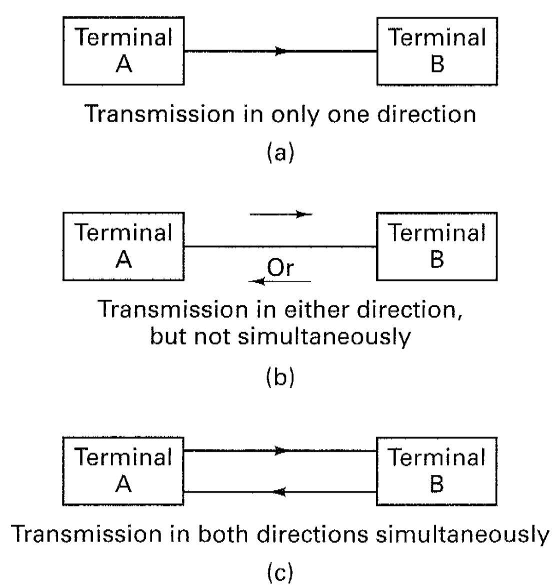

Figure 6.6 Terminal connectivity classifications. (a) Simplex. (b) Half-duplex. (c) Full-duplex.

are made from terminal A to terminal B only, never in the reverse direction. The half-duplex connection in Figure 6.6b is a link whereby transmissions may be made in either direction but not simultaneously. Finally, the full-duplex connection in Figure 6.6c is a two-way link, where transmissions may proceed in both directions simultaneously.

6.2.2 Automatic Repeat Request

When the error control consists of error detection only, the communication system generally needs to provide a means of alerting the transmitter that an error has been detected and that a retransmission is necessary. Such error control procedures are known as *automatic repeat request* or automatic retransmission query (ARQ) methods. Figure 6.7 illustrates three of the most popular ARQ procedures. In each of the diagrams, time is advancing from left to right. The first procedure, called *stop-and-wait ARQ,* is shown in Figure 6.7a. It requires a half-duplex connection only, since the transmitter waits for an acknowledgment (ACK) of each transmis-

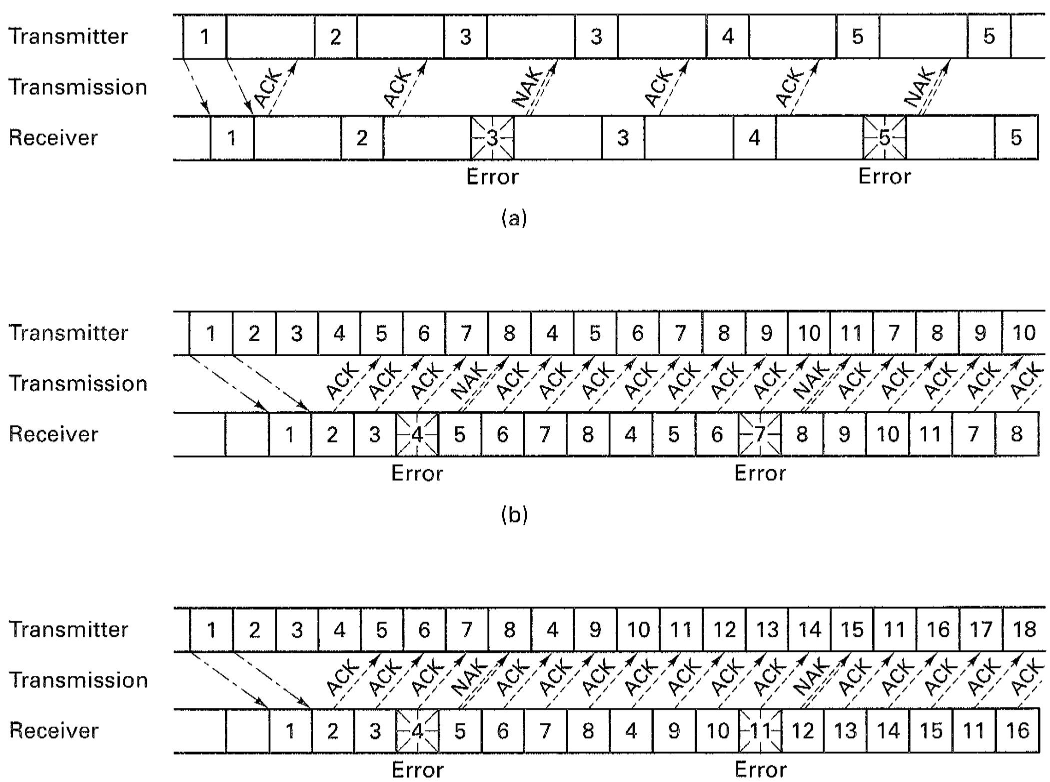

Figure 6.7 Automatic repeat request (ARQ). (a) Stop-and-wait ARQ (half-duplex). (b) Continuous ARQ with pullback (full-duplex). (c) Continuous ARQ with selective repeat (full-duplex).

sion before it proceeds with the next transmission. In the figure, the third transmission block is received in error; therefore, the receiver responds with a negative acknowledgment (NAK), and the transmitter retransmits this third message block before transmitting the next in the sequence. The second ARQ procedure, called *continuous ARQ with pullback,* is shown in Figure 6.7b. Here a full-duplex connection is necessary. Both terminals are transmitting simultaneously; the transmitter is sending message data and the receiver is sending acknowledgment data. Notice that a sequence number has to be assigned to each block of data. Also, the ACKs and NAKs need to reference such numbers, or else there needs to be *a priori* knowledge of the propagation delays, so that the transmitter knows which messages are associated with which acknowledgments. In the example of Figure 6.7b, there is a fixed separation of four blocks between the message being transmitted and the acknowledgment being simultaneously received. For example, when message 8 is being sent, a NAK corresponding to the corrupted message 4 is being received. In the ARQ procedure, the transmitter "pulls back" to the message in error and retransmits all message data, starting with the corrupted message. The final method, called *continuous ARQ with selective repeat,* is shown in Figure 6.7c. Here, as with the second ARQ procedure, a full-duplex connection is needed. In this procedure, however, only the corrupted message is repeated; then, the transmitter continues the transmission sequence where it had left off instead of repeating any subsequent correctly received messages.

The choice of which ARQ procedure to choose is a trade-off between the requirements for efficient utilization of the communications resource and the need to provide full-duplex connectivity. The half-duplex connectivity required in Figure 6.7a is less costly than full-duplex; the associated inefficiency can be measured by the blank time slots. The more efficient utilization illustrated in Figures 6.7b and c requires the more costly full-duplex connectivity.

The major advantage of ARQ over forward error correction (FEC) is that error detection requires much simpler decoding equipment and much less redundancy than does error correction. Also, ARQ is adaptive in the sense that information is retransmitted only when errors occur. On the other hand, FEC may be desirable in place of, or in addition to, error detection, for any of the following reasons:

1. A reverse channel is not available or the delay with ARQ would be excessive.
2. The retransmission strategy is not conveniently implemented.
3. The expected number of errors, without corrections, would require excessive retransmissions.

6.3 STRUCTURED SEQUENCES

In Section 4.8 we considered digital signaling by means of $M = 2^k$ signal waveforms (M-ary signaling), where each waveform contains k bits of information. We saw that in the case of orthogonal M-ary signaling, we can decrease P_B by increasing M