# برنامه درس کدگذاری کانال

# Channel Coding

## نیمسال اول ۱۳۹۸

### الف) عناوین درس:

۱- مقدمه (آشنایی با درس و مباحث مطرح در درس)

۲- مقدمه ریاضی

۳- کدهای قالبی خطی

۴- کدهای گردشی

۵- کدهای کانولوشن

۶- روشهای دکدینگ کدهای کانولوشن

۷- کدهای BCH ، کدهای Reed-Solomon

۸- سایر کدها در صورت وجود وقت

## ب)ارزیابی:

۱- امتحان اول : روز چهار شنبه ۹۸/۰۸/۲۲ (۸ نمره)

۲- امتحان دوم : (پایان ترم) دو شنبه ۹۸/۱۰/۱۶ (۸ نمره)

۳- تکلیف و پروژه درسی: (۴ نمره)

## د) گروه الکترونیکی درس

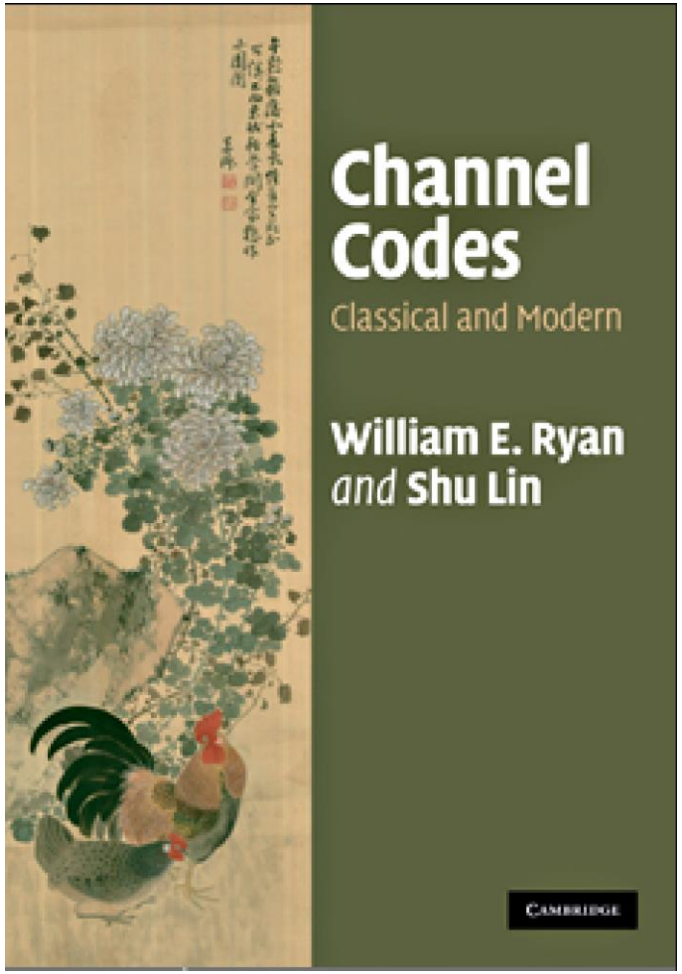
دانشجویان برای دریافت تکالیف،اخبار کلاس، نتایج امتحانها، دریافت صورت تکلیف کامپیوتری و ارسال نتایج آن و... لازم است در گروه تلگرام **Channel Coding** ثبت نام کنند.

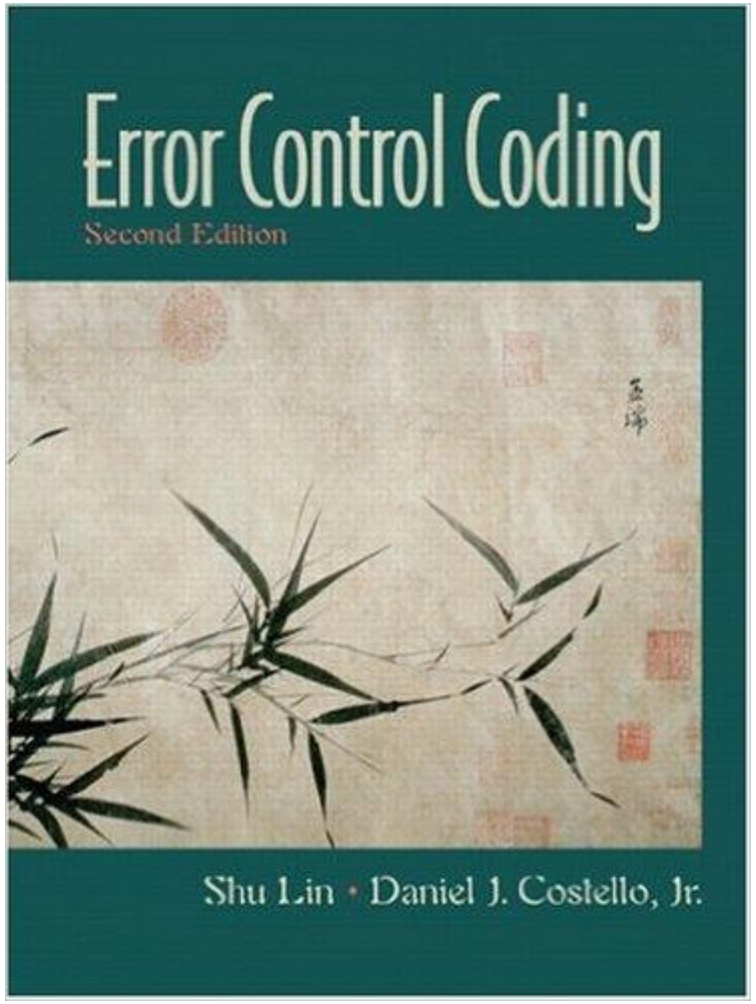
با توجه به تعطیلی برخی روزهای زوج هفته دو جلسه کلاس اضافی در تاریخهای ۹۸/۰۷/۱۰ ، ۹۸/۰۷/۱۷ و ۹۸/۰۷/۲۴ پیش بینی شده است.

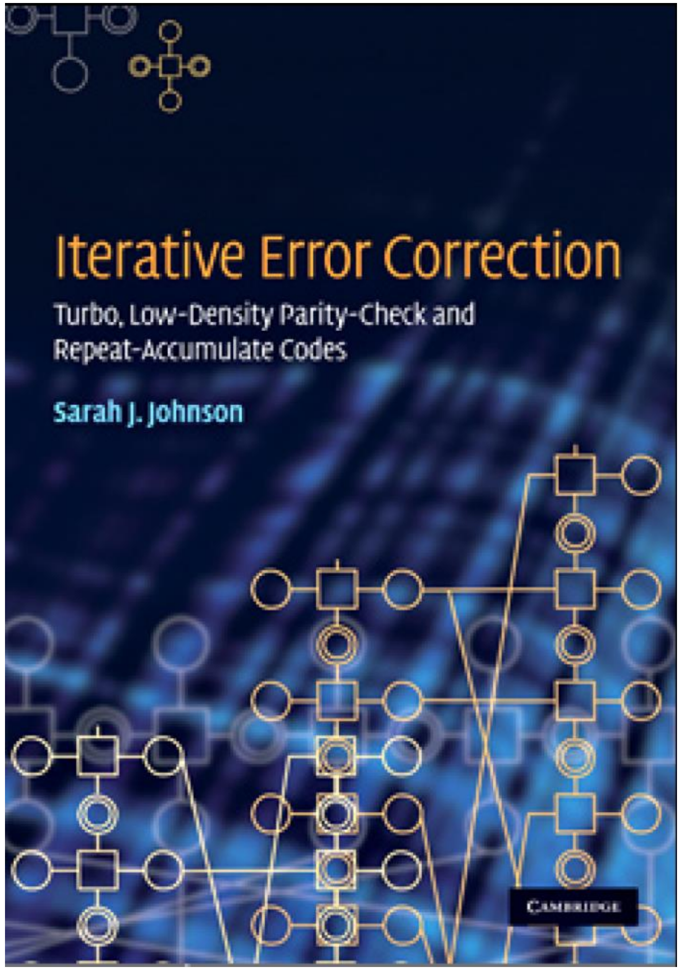
**برنامه درسی**

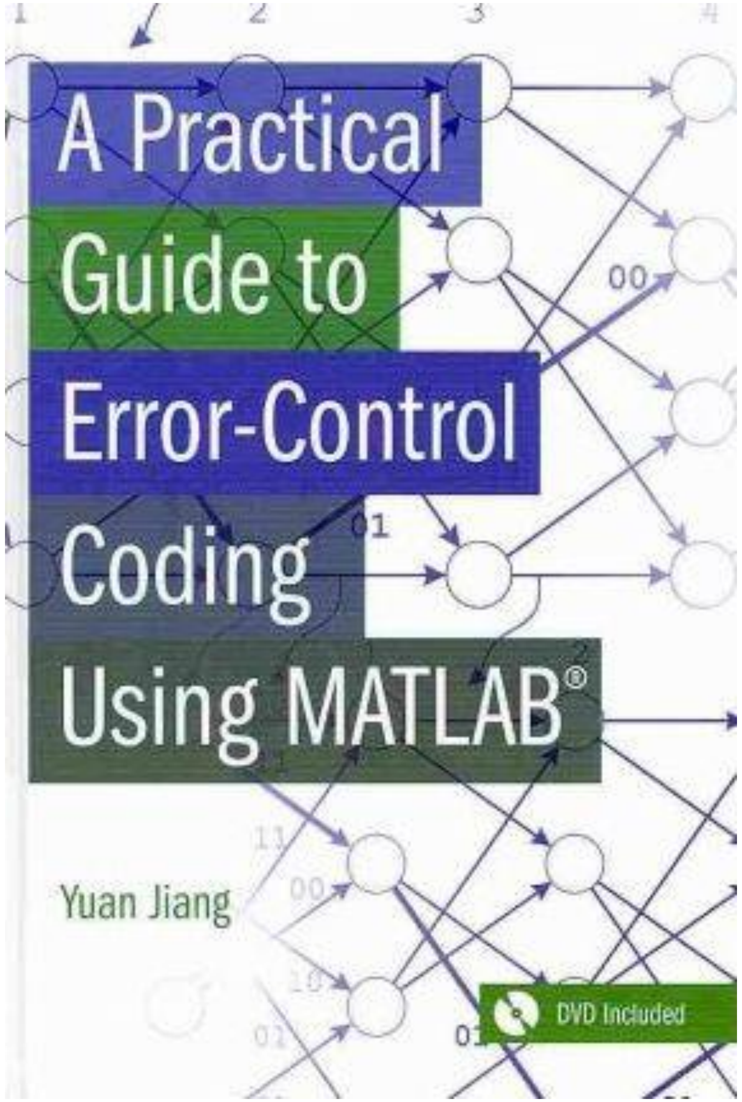
| Subject | Date | Subject | Date |
|---|---|---|---|
| | ۹۸/۰۹/۰۹ | | ۹۸/۰۶/۳۰ |
| | ۹۸/۰۹/۱۱ | | ۹۸/۰۷/۰۲ |
| | ۹۸/۰۹/۱۶ | | ۹۸/۰۷/۰۶ |
| | ۹۸/۰۹/۱۸ | | ۹۸/۰۷/۰۸ |
| | ۹۸/۰۹/۲۳ | کلاس جبرانی | ۹۸/۰۷/۱۰ |
| | ۹۸/۰۹/۲۵ | | ۹۸/۰۷/۱۳ |
| | ۹۸/۰۹/۳۰ | | ۹۸/۰۷/۱۵ |
| | ۹۸/۱۰/۰۲ | کلاس جبرانی | ۹۸/۰۷/۱۷ |
| | ۹۸/۱۰/۰۷ | | ۹۸/۰۷/۲۰ |
| | ۹۸/۱۰/۰۹ | | ۹۸/۰۷/۲۲ |
| Final | ۹۸/۱۰/۱۶ | کلاس جبرانی | ۹۸/۰۷/۲۴ |
| | | | ۹۸/۰۷/۲۷ |
| | | | ۹۸/۰۷/۲۹ |
| | | | ۹۸/۰۸/۰۴ |
| | | | ۹۸/۰۸/۰۶ |
| | | | ۹۸/۰۸/۱۱ |
| | | | ۹۸/۰۸/۱۳ |
| | | | ۹۸/۰۸/۱۸ |
| | | | ۹۸/۰۸/۲۰ |
| | | Midterm | ۹۸/۰۸/۲۲ |
| | | | ۹۸/۰۸/۲۵ |
| | | | ۹۸/۰۸/۲۷ |
| | | | ۹۸/۰۹/۰۲ |
| | | | ۹۸/۰۹/۰۴ |

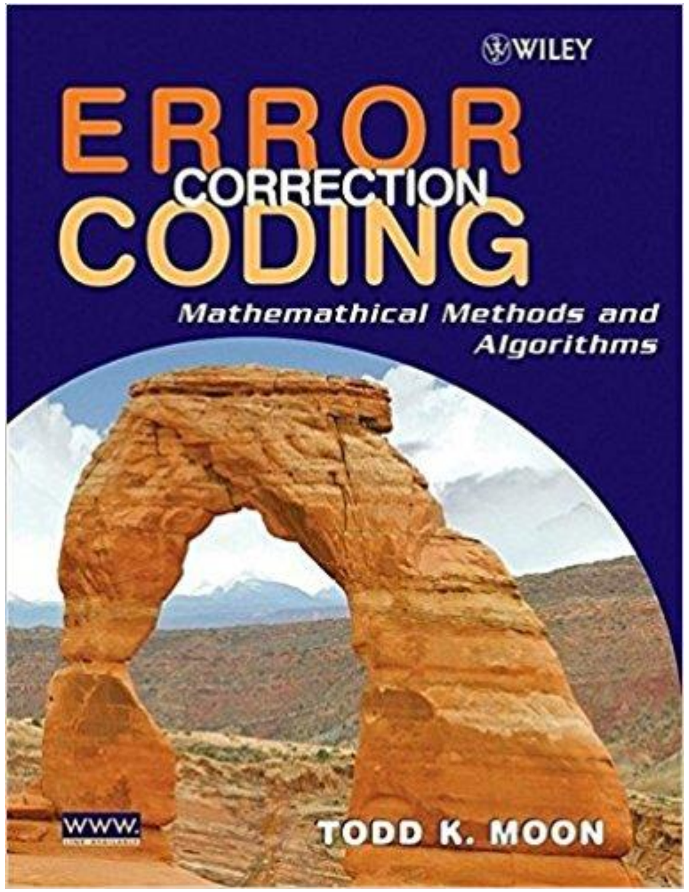
# مراجع درس

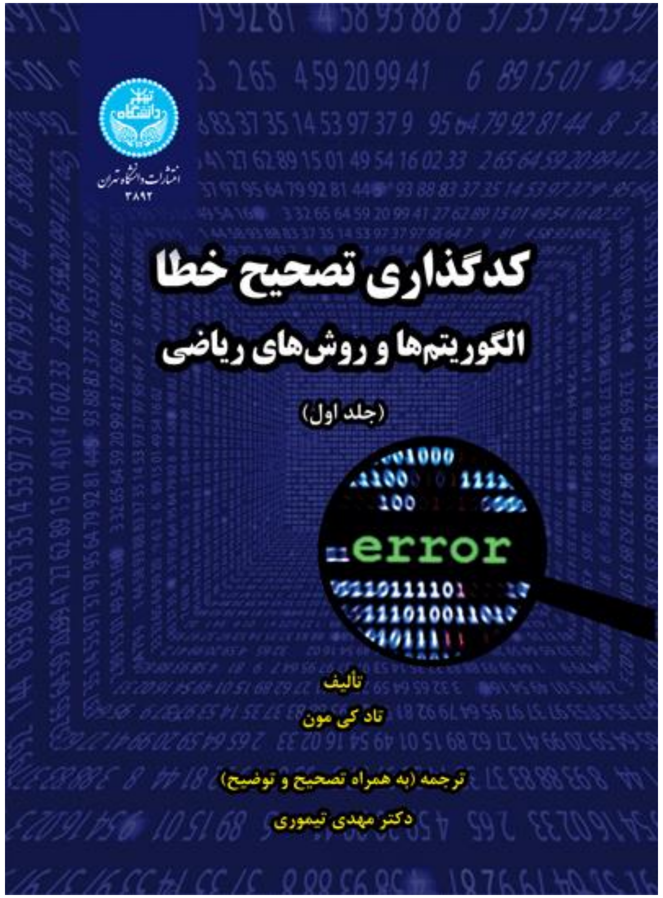

Channel Codes
Classical and Modern
William E. Ryan
and Shu Lin
CAMBRIDGE


Error Control Coding
Second Edition
Shu Lin · Daniel J. Costello, Jr.


Iterative Error Correction
Turbo, Low-Density Parity-Check and
Repeat-Accumulate Codes
Sarah J. Johnson
CAMBRIDGE


A Practical
Guide to
Error-Control
Coding
Using MATLAB®
Yuan Jiang
DVD Included


WILEY
ERROR
CORRECTION
CODING
Mathemathical Methods and
Algorithms
WWW.
TODD K. MOON


انتشارات دانشگاه تهران
۳۸۹۲
کدگذاری تصحیح خطا
الگوریتم‌ها و روش‌های ریاضی
(جلد اول)
error
تألیف
تاد کی مون
ترجمه (به همراه تصحیح و توضیح)
دکتر مهدی تیموری

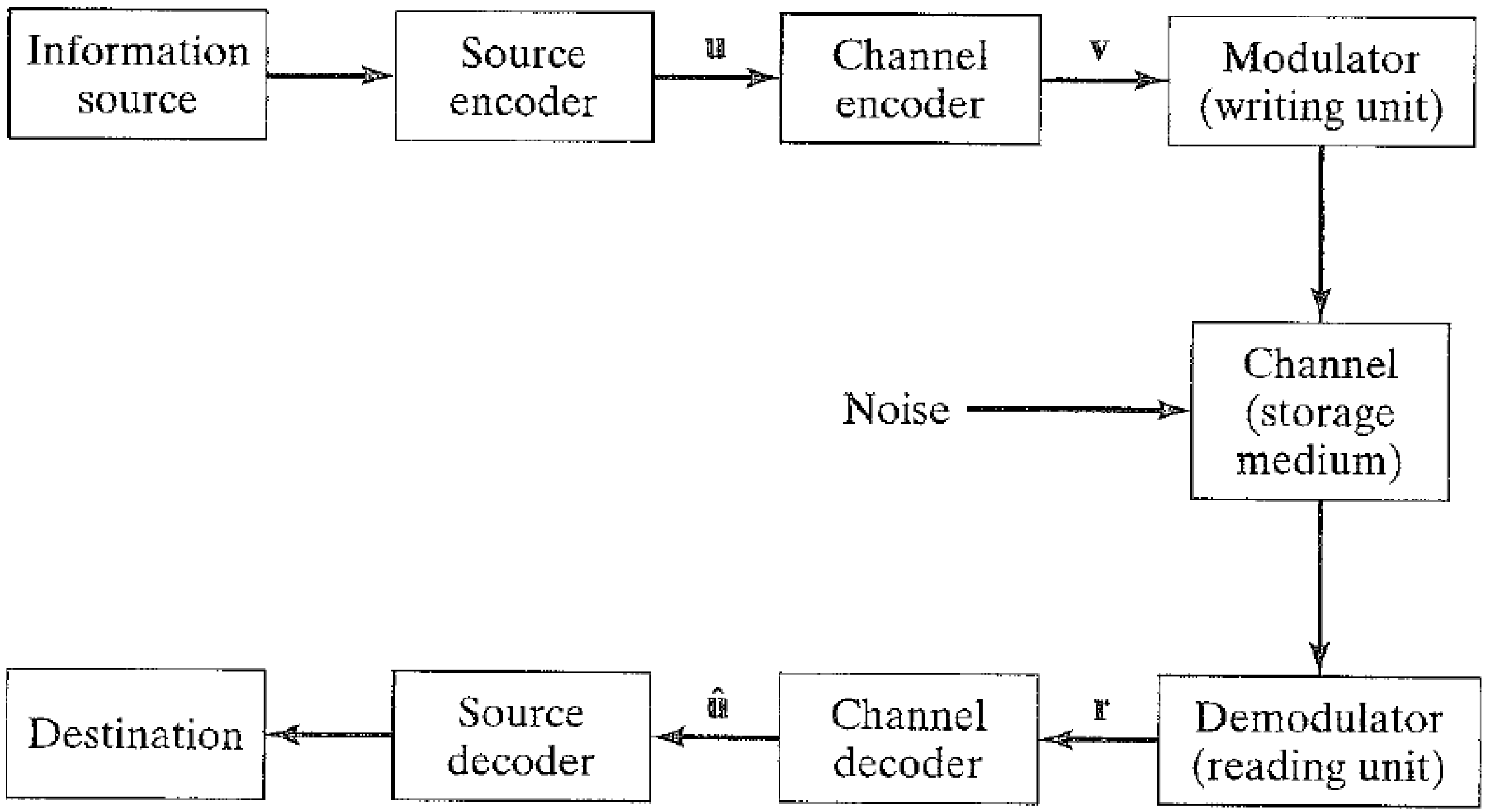



FIGURE 1.1: Block diagram of a typical data transmission or storage system.

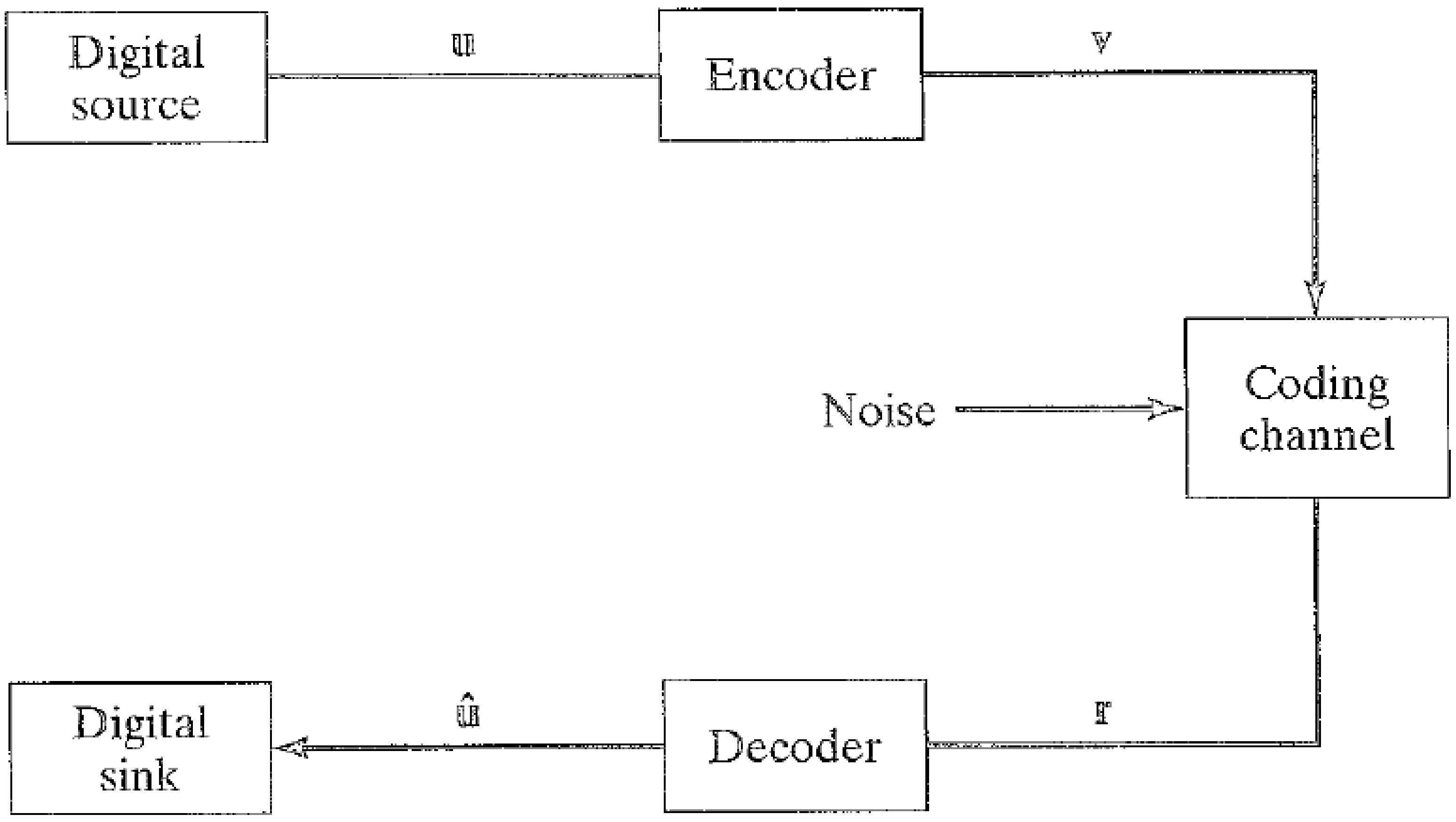



FIGURE 1.2: Simplified model of a coded system.

The major engineering problem that is addressed in this book is to design and implement the channel encoder/decoder pair such that (1) information can be transmitted (or recorded) in a noisy environment as fast (or as densely) as possible; (2) the information can be reliably reproduced at the output of the channel decoder; and (3) the cost of implementing the encoder and decoder falls within acceptable limits.

## TYPES OF CODES

Two structurally different types of codes are in common use today: *block codes* and *convolutional codes*. The encoder for a block code divides the information sequence into message blocks of $k$ information bits (symbols) each. A message block is represented by the binary $k$-tuple $\mathbf{u} = (u_0, u_1, \cdots, u_{k-1})$, called a *message*. (In block coding, the symbol $\mathbf{u}$ is used to denote a $k$-bit message rather than the entire information sequence.) There are a total of $2^k$ different possible messages. The encoder transforms each message $\mathbf{u}$ independently into an $n$-tuple $\mathbf{v} = (v_0, v_1, \cdots, v_{n-1})$ of discrete symbols, called a *codeword*. (In block coding, the symbol $\mathbf{v}$ is used to denote an $n$-symbol block rather than the entire encoded sequence.) Therefore, corresponding to the $2^k$ different possible messages, there are $2^k$ different possible codewords at the encoder output. This set of $2^k$ codewords of length $n$ is called an $(n, k)$ *block code*. The ratio $R = k/n$ is called the *code rate*, and it can be interpreted as the number of information bits entering the encoder per transmitted symbol. Because the $n$-symbol output codeword depends only on the corresponding $k$-bit input message; that is, each message is encoded independently, the encoder is memoryless and can be implemented with a combinational logic circuit.

TABLE 1.1: A binary block code with $k = 4$ and $n = 7$.

| Messages | Codewords |
|---|---|
| (0 0 0 0) | (0 0 0 0 0 0 0) |
| (1 0 0 0) | (1 1 0 1 0 0 0) |
| (0 1 0 0) | (0 1 1 0 1 0 0) |
| (1 1 0 0) | (1 0 1 1 1 0 0) |
| (0 0 1 0) | (1 1 1 0 0 1 0) |
| (1 0 1 0) | (0 0 1 1 0 1 0) |
| (0 1 1 0) | (1 0 0 0 1 1 0) |
| (1 1 1 0) | (0 1 0 1 1 1 0) |
| (0 0 0 1) | (1 0 1 0 0 0 1) |
| (1 0 0 1) | (0 1 1 1 0 0 1) |
| (0 1 0 1) | (1 1 0 0 1 0 1) |
| (1 1 0 1) | (0 0 0 1 1 0 1) |
| (0 0 1 1) | (0 1 0 0 0 1 1) |
| (1 0 1 1) | (1 0 0 1 0 1 1) |
| (0 1 1 1) | (0 0 1 0 1 1 1) |
| (1 1 1 1) | (1 1 1 1 1 1 1) |

The encoder for a convolutional code also accepts $k$-bit blocks of the information sequence $\mathbf{u}$ and produces an encoded sequence (code sequence) $\mathbf{v}$ of $n$-symbol blocks. (In convolutional coding, the symbols $\mathbf{u}$ and $\mathbf{v}$ are used to denote sequences of blocks rather than a single block.) However, each encoded block depends not only on the corresponding $k$-bit message block at the same time unit but also on $m$ previous message blocks. Hence, the encoder has a *memory order* of $m$. The set of all possible encoded output sequences produced by the encoder forms the code. The ratio $R = k/n$ is called the *code rate*. Because the encoder contains memory, it must be implemented with a sequential logic circuit.

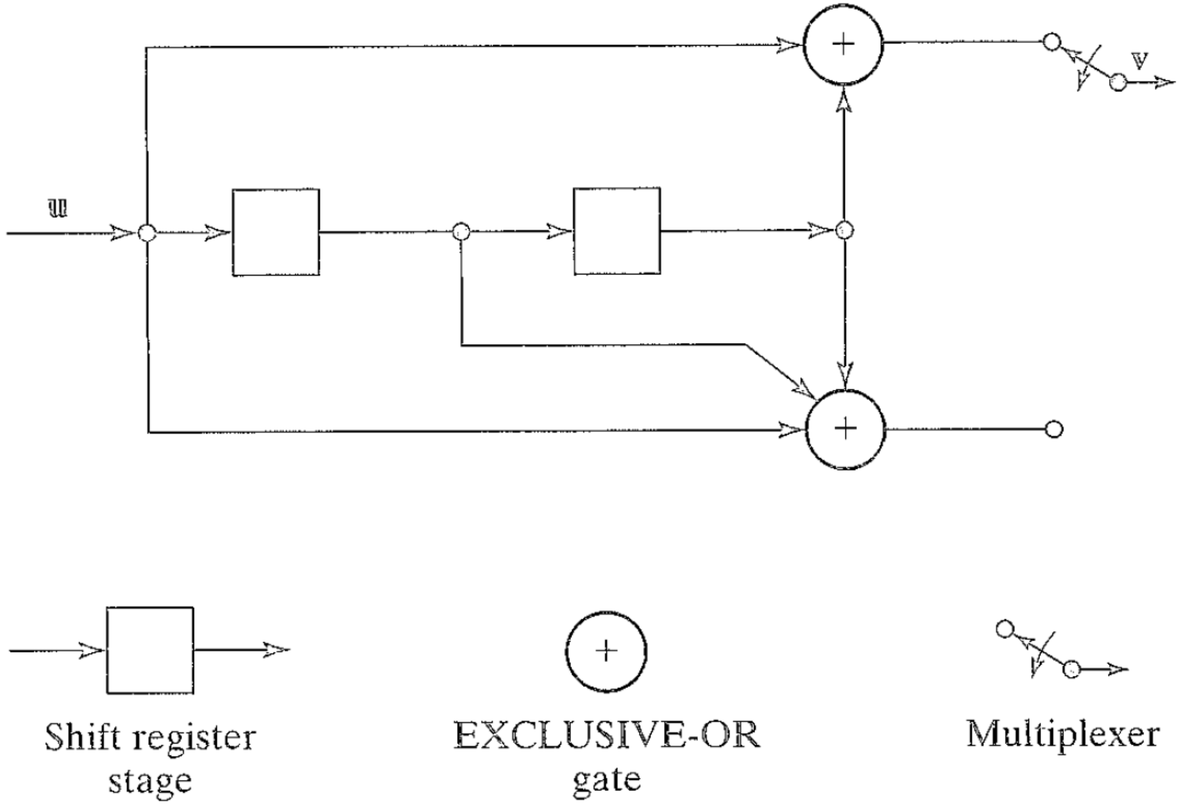



FIGURE 1.3: A binary feed-forward convolutional encoder with $k = 1$, $n = 2$, and $m = 2$.

$$\mathbf{u} = (1\,1\,0\,1\,0\,0\,0\cdots). \qquad \mathbf{v} = (1\,1, 1\,0, 1\,0, 0\,0, 0\,1, 1\,1, 0\,0, 0\,0, 0\,0, \cdots).$$

# MODULATION AND CODING

The modulator in a communication system must select a waveform of duration $T$ seconds that is suitable for transmission for each encoder output symbol. In the case of a binary code, the modulator must generate one of two signals, $s_1(t)$ for an encoded "1" or $s_2(t)$ for an encoded "0". For a wideband channel, the optimum choice of signals is

$$
\begin{aligned}
s_1(t) &= \sqrt{\frac{2E_s}{T}} \cos 2\pi f_0 t, \quad 0 \le t \le T \\
s_2(t) &= \sqrt{\frac{2E_s}{T}} \cos(2\pi f_0 t + \pi) \qquad\qquad (1.1) \\
&= -\sqrt{\frac{2E_s}{T}} \cos 2\pi f_0 t, \quad 0 \le t \le T,
\end{aligned}
$$

where the carrier frequency $f_0$ is a multiple of $1/T$, and $E_s$ is the energy of each signal. This form of modulation is called *binary-phase-shift-keying* (BPSK), since the phase of the carrier $\cos 2\pi f_0 t$ is shifted between 0 and $\pi$, depending on the encoder output. The BPSK-modulated waveform corresponding to the codeword $\mathbf{v} = (1\,1\,0\,1\,0\,0\,0)$ in Table 1.1 is shown in Figure 1.4.

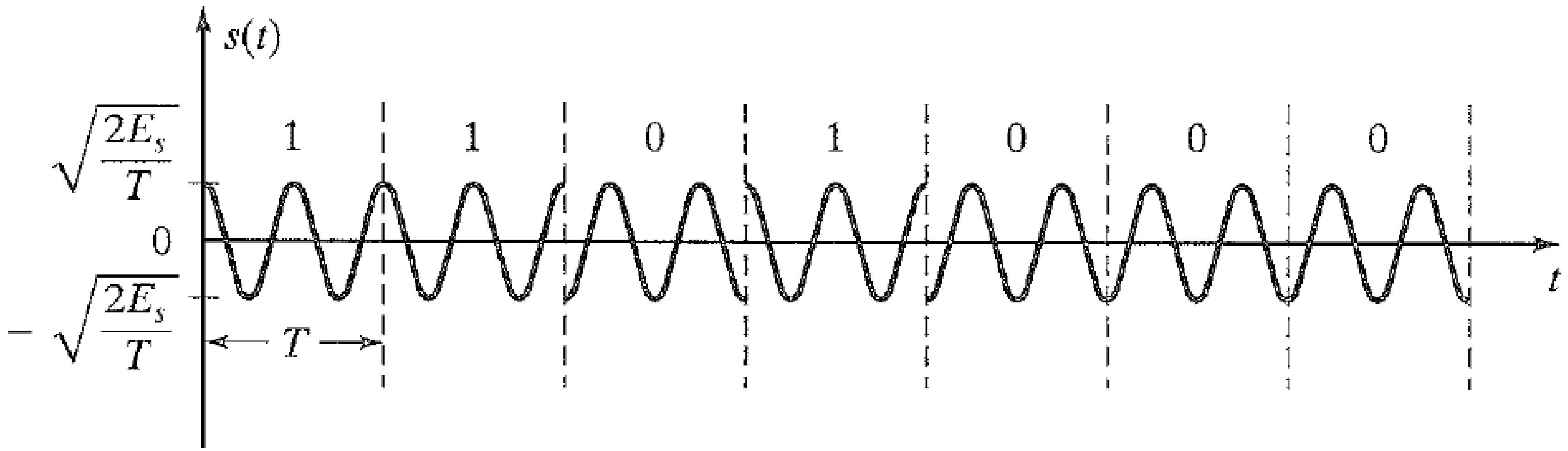



FIGURE 1.4: BPSK-modulated waveform corresponding to the codeword $\mathbf{v} = (1101000)$.

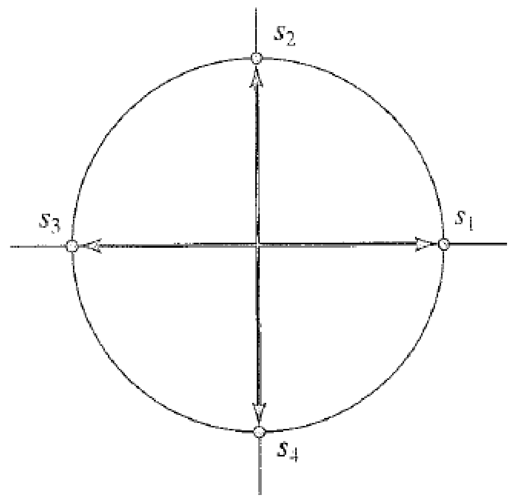
The demodulator must produce an output corresponding to the received signal in each $T$-second interval. This output may be a real number or one of a discrete set of preselected symbols depending on the demodulator design. An optimum demodulator always includes a matched filter or correlation detector followed by a switch that samples the output once every $T$ seconds. For BPSK modulation with coherent detection the sampled output is a real number

$$y = \int_0^T r(t)\sqrt{\frac{2E_s}{T}} \cos 2\pi f_0 t \; dt . \tag{1.3}$$

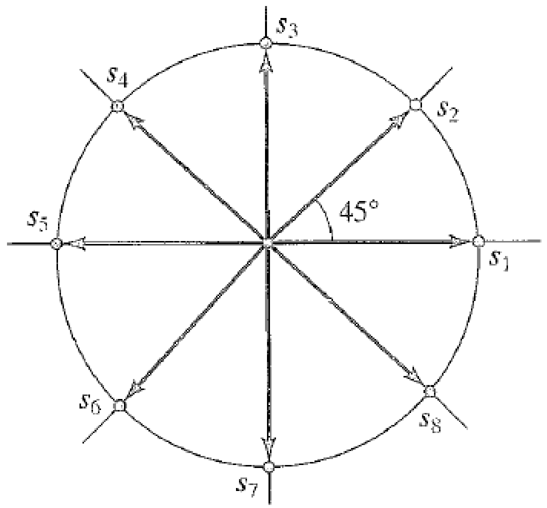
To transmit information with $M = 2^l$ channel signals, the output sequence of the binary encoder is first segmented into a sequence of $l$-bit bytes. Each byte is called a symbol, and there are $M$ distinct symbols. Each symbol is then mapped into one of the $M$ signals in a signal set $S$ for transmission. Each signal is a waveform pulse of duration $T$, which results in $M$-ary modulation. One example of $M$-ary modulation is *M-ary phase-shift-keying* (MPSK), for which the signal set consists of $M$ sinusoidal pulses. These signals have the same energy $E_s$ but $M$ different equally spaced phases. Such a signal set is given by

$$s_i(t) = \sqrt{\frac{2E_s}{T}} \cos(2\pi f_0 t + \phi_i), \quad 0 \le t \le T,$$


(a) BPSK




(b) QPSK




(c) 8-PSK

FIGURE 1.5: BPSK, QPSK, and 8-PSK signal constellations.

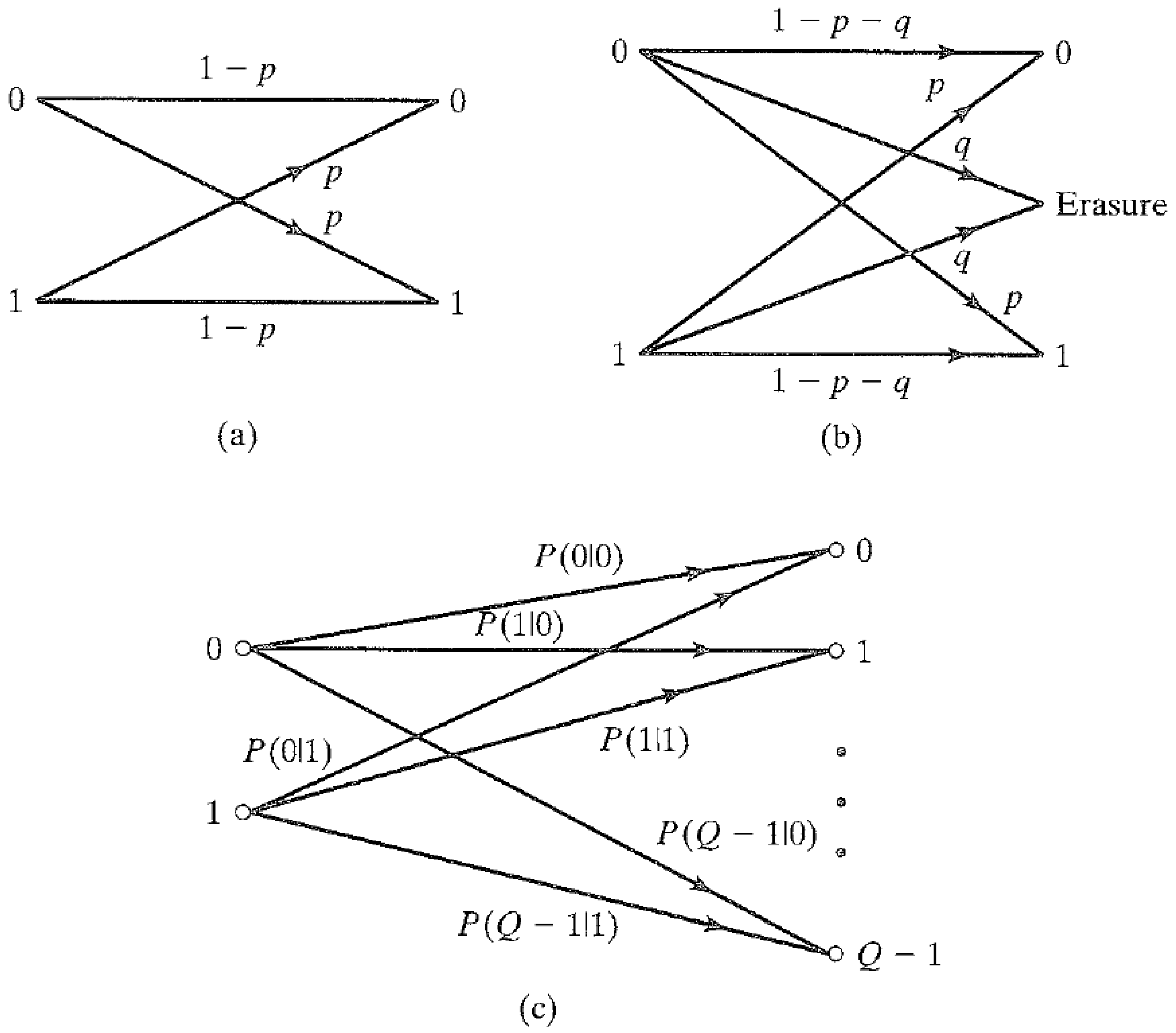



FIGURE 1.6: Transition probability diagrams for (a) binary symmetric channel (BSC); (b) binary symmetric erasure channel; and (c) binary-input, $Q$-ary-output discrete memoryless channel.

The transition probability $p$ can be calculated from a knowledge of the signals used, the probability distribution of the noise, and the output quantization threshold of the demodulator. When BPSK modulation is used on an AWGN channel with optimum coherent detection and binary output quantization, the BSC transition probability is just the uncoded BPSK bit error probability for equally likely signals given by

$$p = Q(\sqrt{2E_s/N_0}), \tag{1.4}$$

where $Q(x) \triangleq \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-y^2/2} dy$ is the *complementary error function* (or simply $Q$-function) of Gaussian statistics. An upper bound on $Q(x)$ that will be used later in evaluating the error performance of codes on a BSC is

$$Q(x) \leq \tfrac{1}{2} e^{-x^2/2}, \qquad x \geq 0. \tag{1.5}$$

Two important and related parameters in any digital communication system are the speed of information transmission and the bandwidth of the channel. Because one encoded symbol is transmitted every $T$ seconds, the *symbol transmission rate*

(baud rate) is $1/T$. In a coded system, if the code rate is $R = k/n$, $k$ information bits correspond to the transmission of $n$ symbols, and the *information transmission rate* (data rate) is $R/T$ bits per second (bps). In addition to signal modification due to the effects of noise, most communication channels are subject to signal distortion owing to bandwidth limitations. To minimize the effect of this distortion on the detection process, the channel should have a *bandwidth* $W$ of at least $\frac{1}{2}T$ hertz (Hz).[1] In an uncoded system ($R = 1$), the data rate $1/T = 2W$ is therefore limited by the channel bandwidth. In a binary coded system, with a code rate $R < 1$, the data rate $R/T = 2RW$ is reduced by the factor $R$ compared with an uncoded system. Hence, to maintain the same data rate as an uncoded system, the coded system requires *bandwidth expansion* by a factor of $1/R$. It is characteristic of binary coded systems to require some bandwidth expansion to maintain a constant data rate. If no additional bandwidth can be used without causing severe signal distortion, binary coding is not feasible, and bandwidth-efficient means of reliable communication

# MAXIMUM LIKELIHOOD DECODING

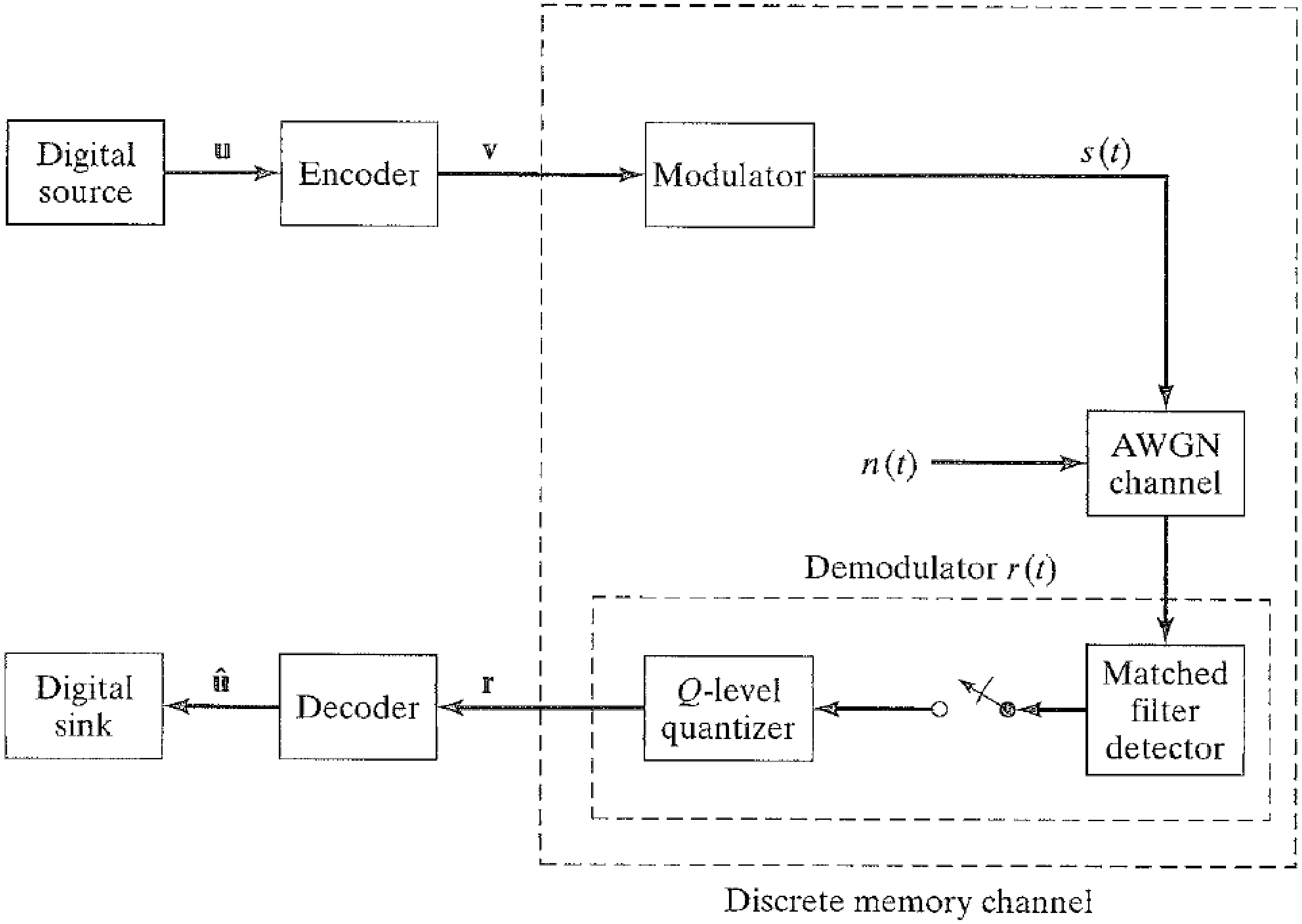



FIGURE 1.7: A coded system on an additive white Gaussian noise channel.

The decoder must produce an estimate $\hat{\mathbf{u}}$ of the information sequence $\mathbf{u}$ based on the received sequence $\mathbf{r}$. Equivalently, since there is a one-to-one correspondence between the information sequence $\mathbf{u}$ and the codeword $\mathbf{v}$, the decoder can produce an estimate $\hat{\mathbf{v}}$ of the codeword $\mathbf{v}$. Clearly, $\hat{\mathbf{u}} = \mathbf{u}$ if and only if $\hat{\mathbf{v}} = \mathbf{v}$. A *decoding rule* is a strategy for choosing an estimated codeword $\hat{\mathbf{v}}$ for each possible received sequence $\mathbf{r}$. If the codeword $\mathbf{v}$ was transmitted, a *decoding error* occurs if and only if $\hat{\mathbf{v}} \neq \mathbf{v}$. Given that $\mathbf{r}$ is received, the *conditional error probability of the decoder* is defined as

$$P(E|\mathbf{r}) \triangleq P(\hat{\mathbf{v}} \neq \mathbf{v}|\mathbf{r}). \tag{1.7}$$

The *error probability of the decoder* is then given by

$$P(E) = \sum_{\mathbf{r}} P(E|\mathbf{r})P(\mathbf{r}), \tag{1.8}$$

where $P(\mathbf{r})$ is the probability of the received sequence $\mathbf{r}$. $P(\mathbf{r})$ is independent of the decoding rule used, since $\mathbf{r}$ is produced prior to decoding. Hence, an optimum decoding rule, that is, one that minimizes $P(E)$, must minimize $P(E|\mathbf{r}) = P(\hat{\mathbf{v}} \neq \mathbf{v}|\mathbf{r})$ for all $\mathbf{r}$. Because minimizing $P(\hat{\mathbf{v}} \neq \mathbf{v}|\mathbf{r})$ is equivalent to maximizing $P(\hat{\mathbf{v}} = \mathbf{v}|\mathbf{r})$, $P(E|\mathbf{r})$ is minimized for a given $\mathbf{r}$ by choosing $\hat{\mathbf{v}}$ as the codeword $\mathbf{v}$ that maximizes

$$P(\mathbf{v}|\mathbf{r}) = \frac{P(\mathbf{r}|\mathbf{v})P(\mathbf{v})}{P(\mathbf{r})}; \tag{1.9}$$

that is, $\hat{\mathbf{v}}$ is chosen as the most likely codeword given that $\mathbf{r}$ is received. If all information sequences, and hence all codewords, are equally likely; that is, $P(\mathbf{v})$ is the same for all $\mathbf{v}$, maximizing (1.9) is equivalent to maximizing $P(\mathbf{r}|\mathbf{v})$. For a DMC,

$$P(\mathbf{r}|\mathbf{v}) = \prod_{i} P(r_i|v_i), \tag{1.10}$$

$$\log P(\mathbf{r}|\mathbf{v}) = \sum_i \log P(r_i|v_i).$$

Now, consider specializing the MLD decoding rule to the BSC. In this case $\mathbf{r}$ is a binary sequence that may differ from the transmitted codeword $\mathbf{v}$ in some positions owing to the channel noise. When $r_i \neq v_i$, $P(r_i|v_i) = p$, and when $r_i = v_i$, $P(r_i|v_i) = 1 - p$. Let $d(\mathbf{r}, \mathbf{v})$, be the distance between $\mathbf{r}$ and $\mathbf{v}$, that is, the number of positions in which $\mathbf{r}$ and $\mathbf{v}$ differ. This distance is called the *Hamming distance*. For a block code of length $n$, (1.11) becomes

$$\begin{aligned} \log P(\mathbf{r}|\mathbf{v}) &= d(\mathbf{r}, \mathbf{v}) \log p + [n - d(\mathbf{r}, \mathbf{v})] \log(1 - p) \\ &= d(\mathbf{r}, \mathbf{v}) \log \frac{p}{1-p} + n \log(1 - p). \end{aligned} \tag{1.12}$$

(For a convolutional code, $n$ in (1.12) is replaced by $N$.) Since $\log \frac{p}{1-p} < 0$ for $p < \frac{1}{2}$, and $n \log(1-p)$ is a constant for all $\mathbf{v}$, the MLD decoding rule for the BSC *chooses* $\hat{\mathbf{v}}$ *as the codeword* $\mathbf{v}$ *that minimizes the distance* $d(\mathbf{r}, \mathbf{v})$ *between* $\mathbf{r}$ *and* $\mathbf{v}$; that is, *it chooses the codeword that differs from the received sequence in the fewest number of positions*. Hence, an MLD for the BSC is sometimes called a *minimum distance decoder*.

The capability of a noisy channel to transmit information reliably was determined by Shannon [1] in his original work. This result, called the *noisy channel coding theorem*, states that every channel has a *capacity* $C$, and that for any rate $R < C$, there exist codes of rate $R$ that, with maximum likelihood decoding, have an arbitrarily small decoding error probability $P(E)$. In particular, for any $R < C$, there exist block codes with sufficiently large block length $n$ such that

$$P(E) \leq 2^{-nE_b(R)}, \tag{1.13}$$

and there exist convolutional codes with sufficiently large memory order $m$ such that

$$P(E) \leq 2^{-(m+1)nE_c(R)}. \tag{1.14}$$

$E_b(R)$ and $E_c(R)$ are positive functions of $R$ for $R < C$ and are completely determined by the channel characteristics. The bound of (1.13) implies that arbitrarily small error probabilities are achievable with block coding for any fixed $R < C$ by increasing the block length $n$ while holding the ratio $k/n$ constant. The bound of (1.14) implies that arbitrarily small error probabilities are also achievable with convolutional coding for any fixed $R < C$ by increasing the memory order $m$ while holding $k$ and $n$ constant.

# TYPES OF ERRORS

*random-error channels* *random-error correcting codes.*

*burst- error channels* *burst-error correcting codes*

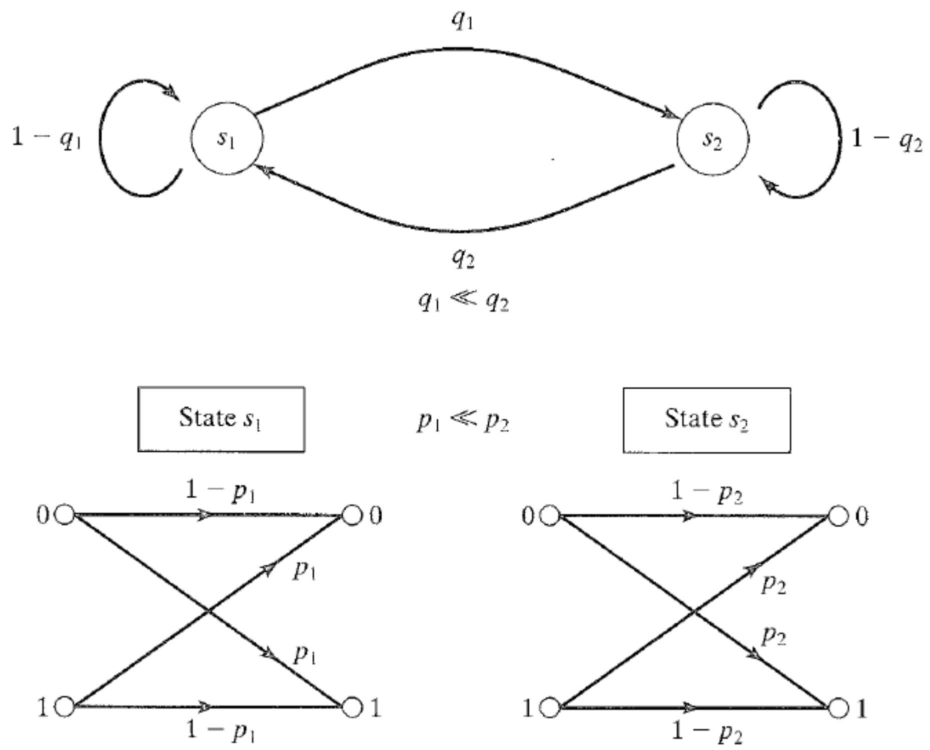



FIGURE 1.8: A simplified model of a channel with memory.

ERROR CONTROL STRATEGIES

- FEC
- ARQ

# FEC

The block diagram shown in Figure 1.1 represents a one-way communication system. The transmission (or recording) is strictly in one direction, from transmitter to receiver. Error control strategies for a one-way system must use *forward error correction* (FEC); that is, they employ error-correcting codes that automatically correct errors detected at the receiver. Examples are digital storage systems, in which the information recorded on a storage medium may be replayed weeks or even months after it is recorded; and deep-space communication systems, where the relatively simple encoding equipment can be placed aboard the spacecraft, but the much more complex decoding procedure must be performed on Earth. Most of the coded systems in use today employ some form of FEC, even if the channel is not strictly one-way. This book is devoted mostly to the analysis and design of FEC systems.

# ARQ

In some cases, though, a transmission system can be two-way; that is, information can be sent in both directions, and the transmitter also acts as a receiver (a transceiver), and vice versa. Examples of two-way systems are some data networks and satellite communication systems. Error control strategies for a two-way system use error detection and retransmission, called *automatic repeat request* (ARQ). In an ARQ system, when errors are detected at the receiver, a request is sent for the transmitter to repeat the message, and repeat requests continue to be sent until the message is correctly received.

PERFORMANCE MEASURES

decoding error (called the *error probability*)

*coding gain* and *asymptotic coding gain*

decoding error

*bit-error rate* (BER)

*word-error rate* (WER)

*block-error rate* (BLER)

In a coded communication system with code rate $R = k/n$, the number of transmitted symbols (or signals) per information bit is $1/R$. If the energy per transmitted symbol is $E_s$, then the energy-per-information bit is

$$E_b = E_s/R. \tag{1.15}$$

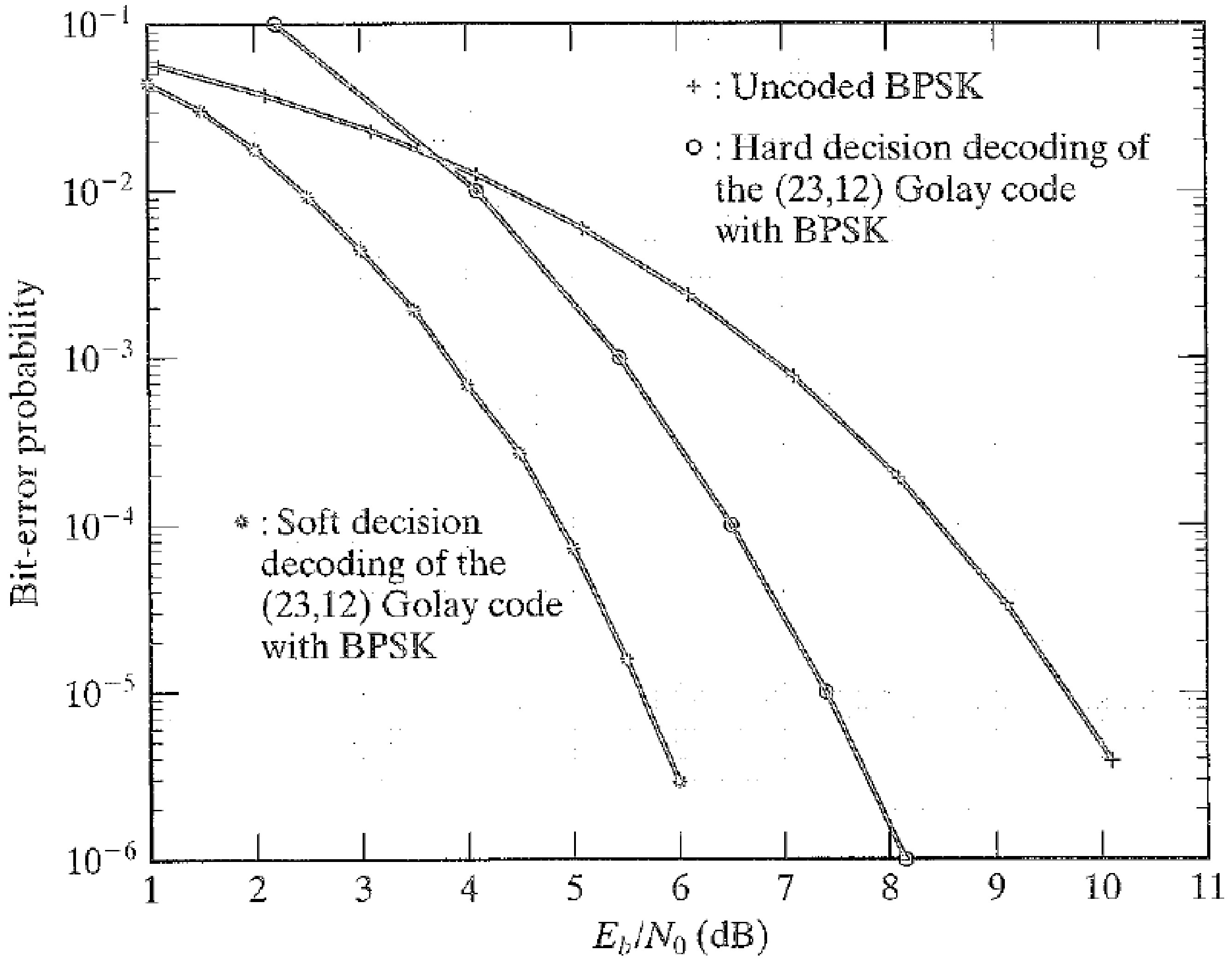



FIGURE 1.9: Bit-error performance of a coded communication system with the (23,12) Golay code.

From Figure 1.9, we observe that at sufficiently low values of $E_b/N_0$, the coding gain actually becomes negative. This threshold phenomenon is common to all coding schemes. There always exists an $E_b/N_0$ below which the code loses its effectiveness and actually makes the situation worse. This SNR is called the *coding threshold*. It is important to keep this threshold low and to maintain a coded communication system operating at an SNR well above its coding threshold. For the Golay-coded system with hard-decision decoding, the coding threshold is 3.7 dB.

Another quantity that is sometimes used as a performance measure for a particular code is the *asymptotic coding gain* (the coding gain for large SNR). This coding gain depends only on the code rate and the minimum distance $d_{\min}$ of the code, which is defined as the minimum Hamming distance between any two codewords. For an AWGN channel, it follows from (1.4) and (1.5) that for high SNR (large $E_b/N_0$), the bit-error probability of an uncoded BPSK system with coherent detection can be approximated by

$$(P_b)_{uncoded} \cong 1/2e^{-(E_b/N_0)_{uncoded}}. \tag{1.16}$$

For high SNR with soft-decision (unquantized) MLD, the bit error probability of an $(n,k)$ block code with minimum distance $d_{\min}$ can be approximated by (see Chapter 10)

$$(P_b)_{coded} \cong Ke^{-d_{\min}R(E_b/N_0)_{coded}}, \tag{1.17}$$

where $R = k/n$ is the code rate, and $K$ is a (typically small) constant that depends only on the code. One may compute the reduction in SNR by equating (1.16) and (1.17), which gives

$$Ke^{-d_{\min}R(E_b/N_0)_{coded}} = \frac{1}{2}e^{-(E_b/N_0)_{uncoded}}. \tag{1.18}$$

Taking the logarithm (base $e$) of both sides and noting that $\log K$ and $\log \frac{1}{2}$ are negligible for large $E_b/N_0$, we obtain

$$\frac{(E_b/N_0)_{uncoded}}{(E_b/N_0)_{coded}} = Rd_{min}. \tag{1.19}$$

Thus, for soft-decision MLD, the asymptotic coding gain of an $(n, k)$ code with minimum distance $d_{min}$ over uncoded BPSK is given by

$$\gamma_{asymp} = 10 \log_{10} R d_{min}. \tag{1.20}$$

For hard-decision MLD, it can be shown [7] that the asymptotic coding gain is

$$\gamma_{asymp} = 10 \log_{10} \left(\frac{1}{2}\right) R d_{min}. \tag{1.21}$$

From (1.20) and (1.21), we see that soft-decision MLD has a

$$10 \log_{10} 2 = 3 \text{ dB}$$

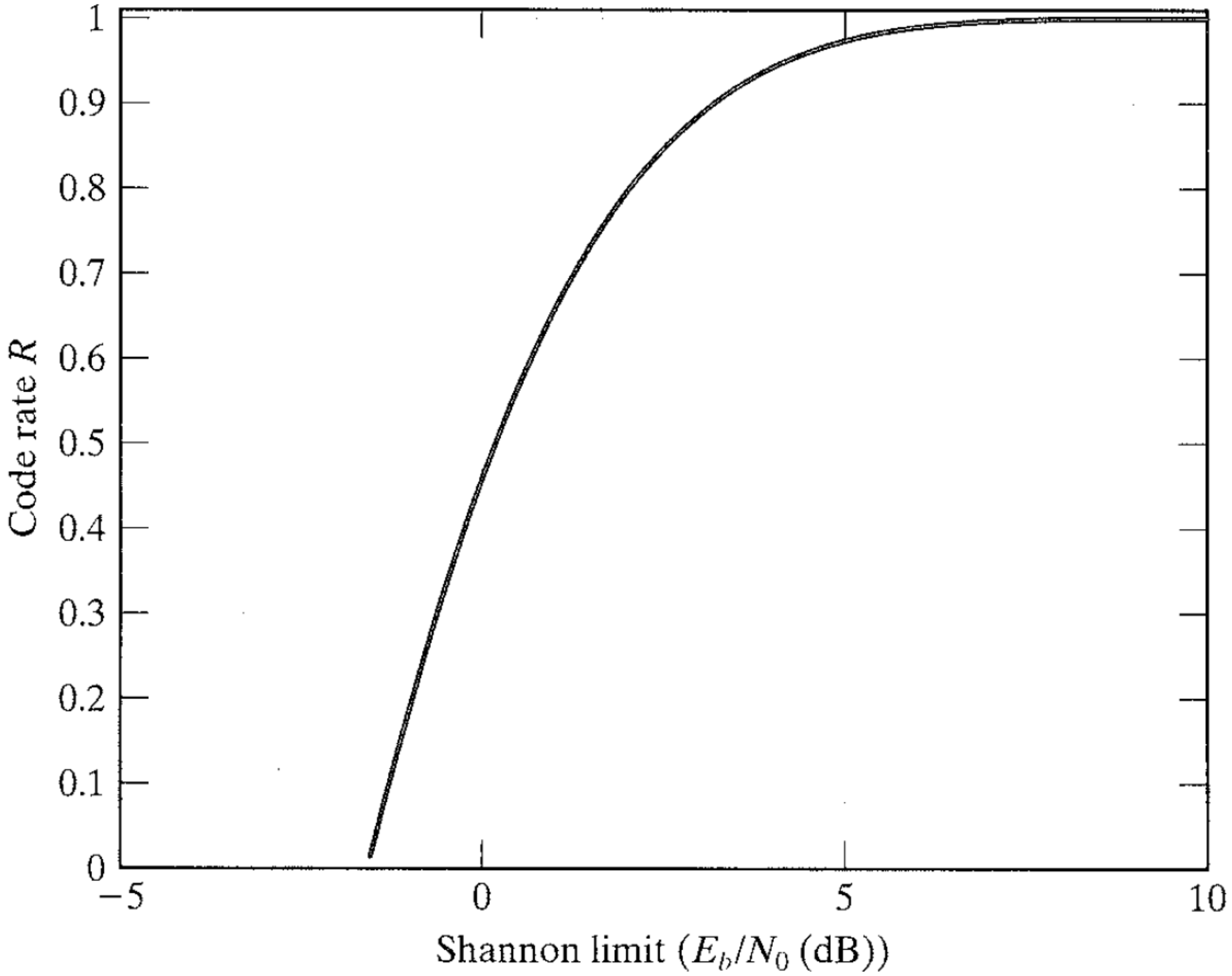



FIGURE 1.10: Shannon limit as a function of code rate $R$.

TABLE 1.2: Shannon limit of a continuous-output AWGN channel with BPSK signaling for various code rates $R$.

| $R$ | $E_b/N_0$ (dB) | $R$ | $E_b/N_0$ (dB) | $R$ | $E_b/N_0$ (dB) | $R$ | $E_b/N_0$ (dB) |
|---|---|---|---|---|---|---|---|
| 0.01 | −1.548 | 0.35 | −0.432 | 0.69 | 1.208 | 0.954 | 4.304 |
| 0.02 | −1.531 | 0.36 | −0.394 | 0.70 | 1.275 | 0.958 | 4.425 |
| 0.03 | −1.500 | 0.37 | −0.355 | 0.71 | 1.343 | 0.961 | 4.521 |
| 0.04 | −1.470 | 0.38 | −0.314 | 0.72 | 1.412 | 0.964 | 4.618 |
| 0.05 | −1.440 | 0.39 | −0.276 | 0.73 | 1.483 | 0.967 | 4.725 |
| 0.06 | −1.409 | 0.40 | −0.236 | 0.74 | 1.554 | 0.970 | 4.841 |
| 0.07 | −1.378 | 0.41 | −0.198 | 0.75 | 1.628 | 0.972 | 4.922 |
| 0.08 | −1.347 | 0.42 | −0.156 | 0.76 | 1.708 | 0.974 | 5.004 |
| 0.09 | −1.316 | 0.42 | −0.118 | 0.77 | 1.784 | 0.976 | 5.104 |
| 0.10 | −1.285 | 0.44 | −0.074 | 0.78 | 1.867 | 0.978 | 5.196 |
| 0.11 | −1.254 | 0.45 | −0.032 | 0.79 | 1.952 | 0.980 | 5.307 |
| 0.12 | −1.222 | 0.46 | 0.010 | 0.800 | 2.045 | 0.982 | 5.418 |
| 0.13 | −1.190 | 0.47 | 0.055 | 0.807 | 2.108 | 0.983 | 5.484 |
| 0.14 | −1.158 | 0.48 | 0.097 | 0.817 | 2.204 | 0.984 | 5.549 |
| 0.15 | −1.126 | 0.49 | 0.144 | 0.827 | 2.302 | 0.985 | 5.615 |
| 0.16 | −1.094 | 0.50 | 0.188 | 0.837 | 2.402 | 0.986 | 5.681 |
| 0.17 | −1.061 | 0.51 | 0.233 | 0.846 | 2.503 | 0.987 | 5.756 |
| 0.18 | −1.028 | 0.52 | 0.279 | 0.855 | 2.600 | 0.988 | 5.842 |
| 0.19 | −0.995 | 0.53 | 0.326 | 0.864 | 2.712 | 0.989 | 5.927 |
| 0.20 | −0.963 | 0.54 | 0.374 | 0.872 | 2.812 | 0.990 | 6.023 |
| 0.21 | −0.928 | 0.55 | 0.424 | 0.880 | 2.913 | 0.991 | 6.119 |
| 0.22 | −0.896 | 0.56 | 0.474 | 0.887 | 3.009 | 0.992 | 6.234 |
| 0.23 | −0.861 | 0.57 | 0.526 | 0.894 | 3.114 | 0.993 | 6.360 |
| 0.24 | −0.827 | 0.58 | 0.574 | 0.900 | 3.205 | 0.994 | 6.495 |
| 0.25 | −0.793 | 0.59 | 0.628 | 0.907 | 3.312 | 0.995 | 6.651 |
| 0.26 | −0.757 | 0.60 | 0.682 | 0.913 | 3.414 | 0.996 | 6.837 |
| 0.27 | −0.724 | 0.61 | 0.734 | 0.918 | 3.500 | 0.997 | 7.072 |
| 0.28 | −0.687 | 0.62 | 0.791 | 0.924 | 3.612 | 0.998 | 7.378 |
| 0.29 | −0.651 | 0.63 | 0.844 | 0.929 | 3.709 | 0.999 | 7.864 |
| 0.30 | −0.616 | 0.64 | 0.904 | 0.934 | 3.815 | | |
| 0.31 | −0.579 | 0.65 | 0.960 | 0.938 | 3.906 | | |
| 0.32 | −0.544 | 0.66 | 1.021 | 0.943 | 4.014 | | |
| 0.33 | −0.507 | 0.67 | 1.084 | 0.947 | 4.115 | | |
| 0.34 | −0.469 | 0.68 | 1.143 | 0.951 | 4.218 | | |

Figure 1.10 shows the Shannon limit as a function of the code rate $R$ for BPSK signaling on a continuous-output AWGN channel. From this figure we can determine the Shannon limit for a given code rate. For example, to achieve error-free communication with a coded system of rate $R = \frac{1}{2}$ over a continuous-output AWGN channel using BPSK signaling, the minimum required SNR (the Shannon limit) is 0.188 dB. Table 1.2 gives the values of the Shannon limit for various code rates from 0 to 0.999. The Shannon limit can be used as a yardstick to measure the maximum achievable coding gain for a coded system with a given rate $R$ over an uncoded system with the same modulation signal set. For example, to achieve a BER of $10^{-5}$, an uncoded BPSK system requires an SNR of 9.65 dB. For a coded BPSK system with code rate $R = \frac{1}{2}$, the Shannon limit is 0.188. Therefore, the maximum achievable (or potential) coding gain for a coded BPSK system with a code rate $R = \frac{1}{2}$ is 9.462 dB. For example, Figure 1.11 shows the bit-error performance of a rate $R = \frac{1}{2}$ convolutional code with memory order $m = 6$ decoded with soft-decision MLD (see Chapter 12). To achieve a BER of $10^{-5}$, this code requires an SNR of 4.15 dB and achieves a 5.35 dB coding gain compared to uncoded BPSK; however, it is 3.962 dB away from the Shannon limit. This gap can be reduced by using a longer and more powerful code, block or convolutional, decoded with an efficient soft-decision MLD or near-MLD algorithm.

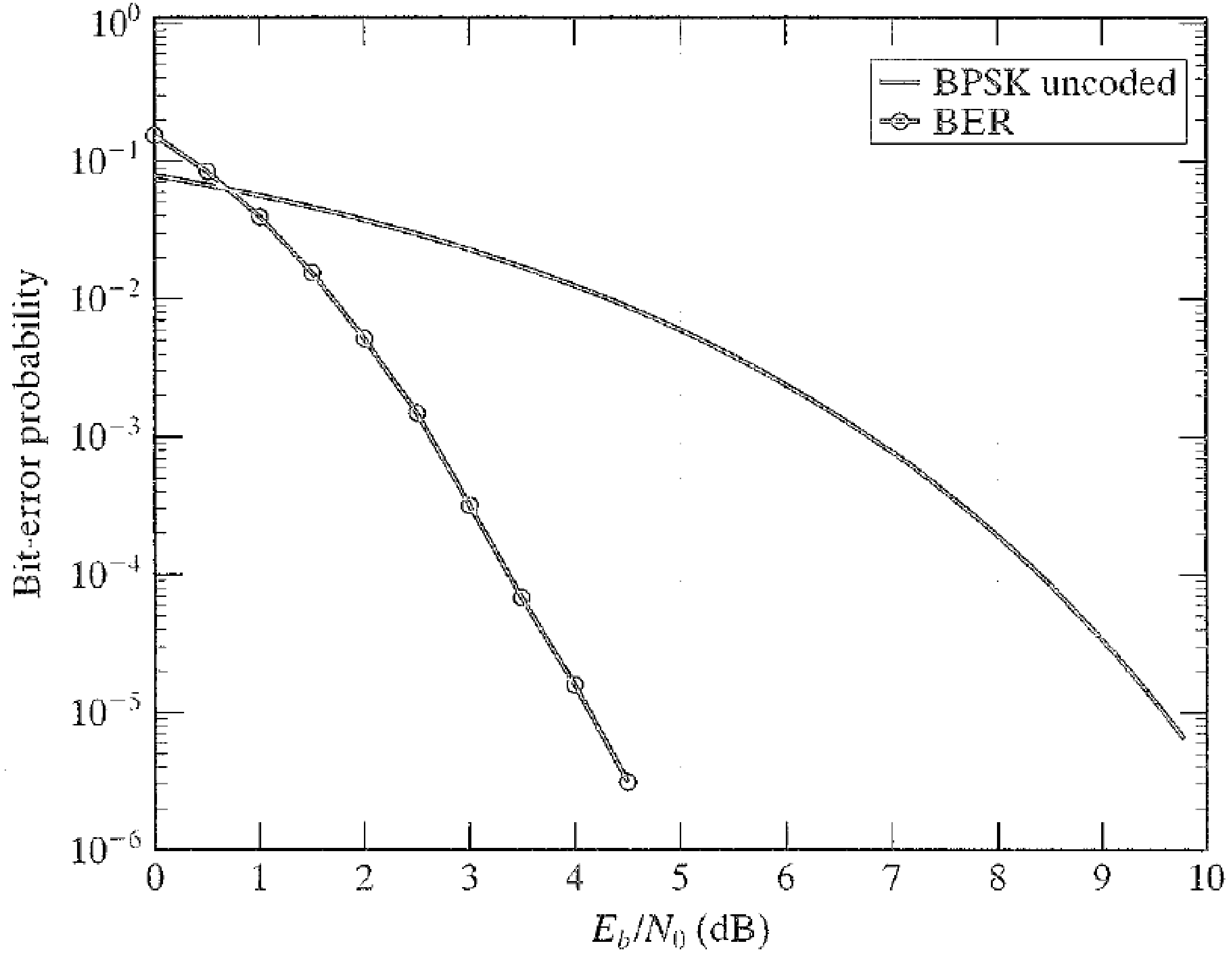



FIGURE 1.11: Bit-error performance of a rate $R = \frac{1}{2}$ convolutional code with memory order $m = 6$ decoded with soft-decision MLD.

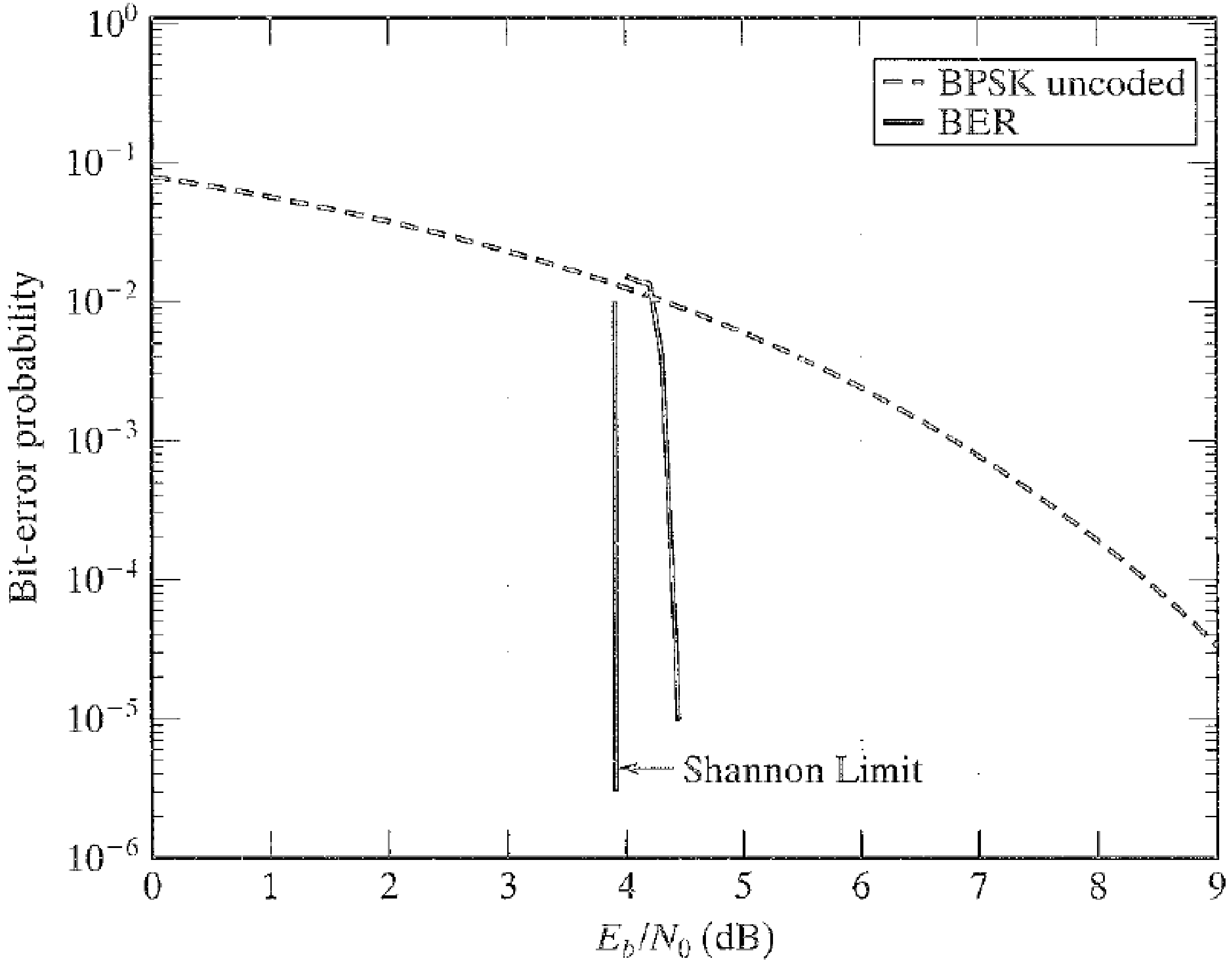



FIGURE 1.12: Bit-error performance of a (65520,61425) low-density parity check code decoded with a soft-decision near-MLD algorithm.

# CODED MODULATION

Combining coding with binary modulation to achieve coding gain (or to improve error performance) results in channel bandwidth expansion; that is, the coded system requires a larger channel bandwidth for signal transmission than the uncoded system. This is because redundant symbols must be added to the transmitted information sequence to combat the channel noise (i.e., for error control).

To overcome this problem, coding must be used in conjunction with bandwidth-efficient multilevel modulation ($M$-ary modulation with $M > 2$). If coding and modulation are combined properly and designed as a single entity, coding gain can be achieved without bandwidth expansion. Such a combination of coding and modulation is called *coded modulation*. The first coded modulation technique, known as *trellis coded modulation*, was devised by Ungerboeck

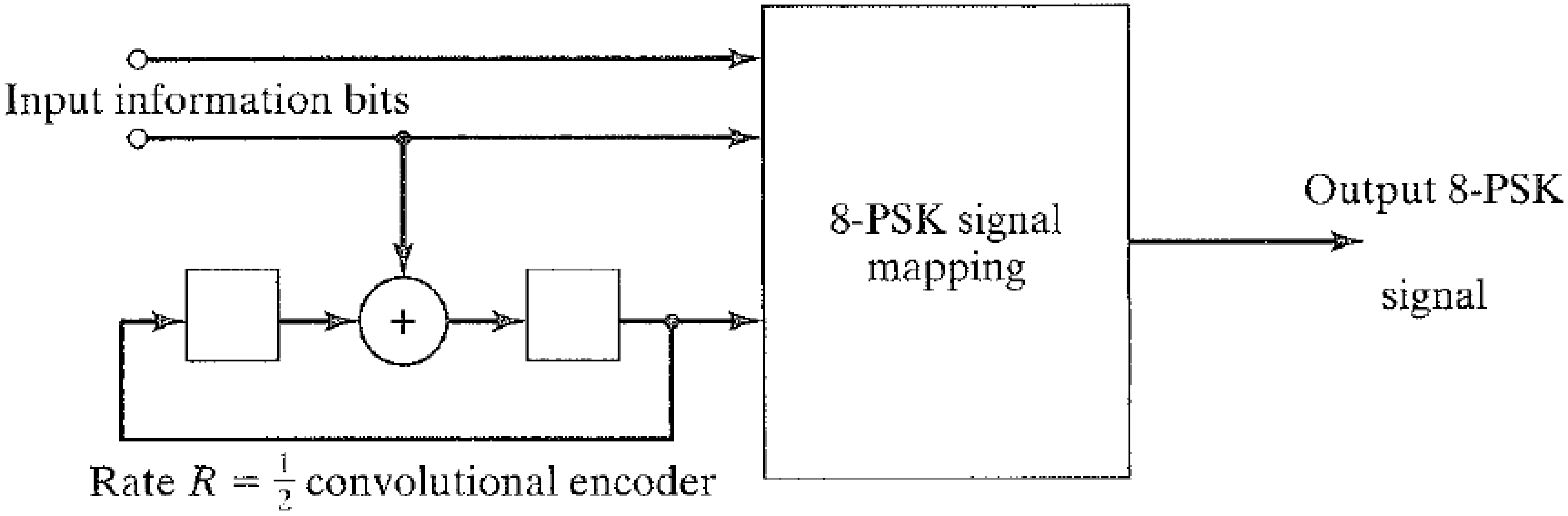



FIGURE 1.13: A trellis coded 8-PSK modulation system.

Coded modulation can be classified into two categories based on the code structure:

1. Trellis coded modulation (TCM), in which convolutional (or trellis) codes are combined with multilevel modulation signal sets, and

2. Block coded modulation (BCM) [13], in which block codes are combined with multilevel modulation signal sets.