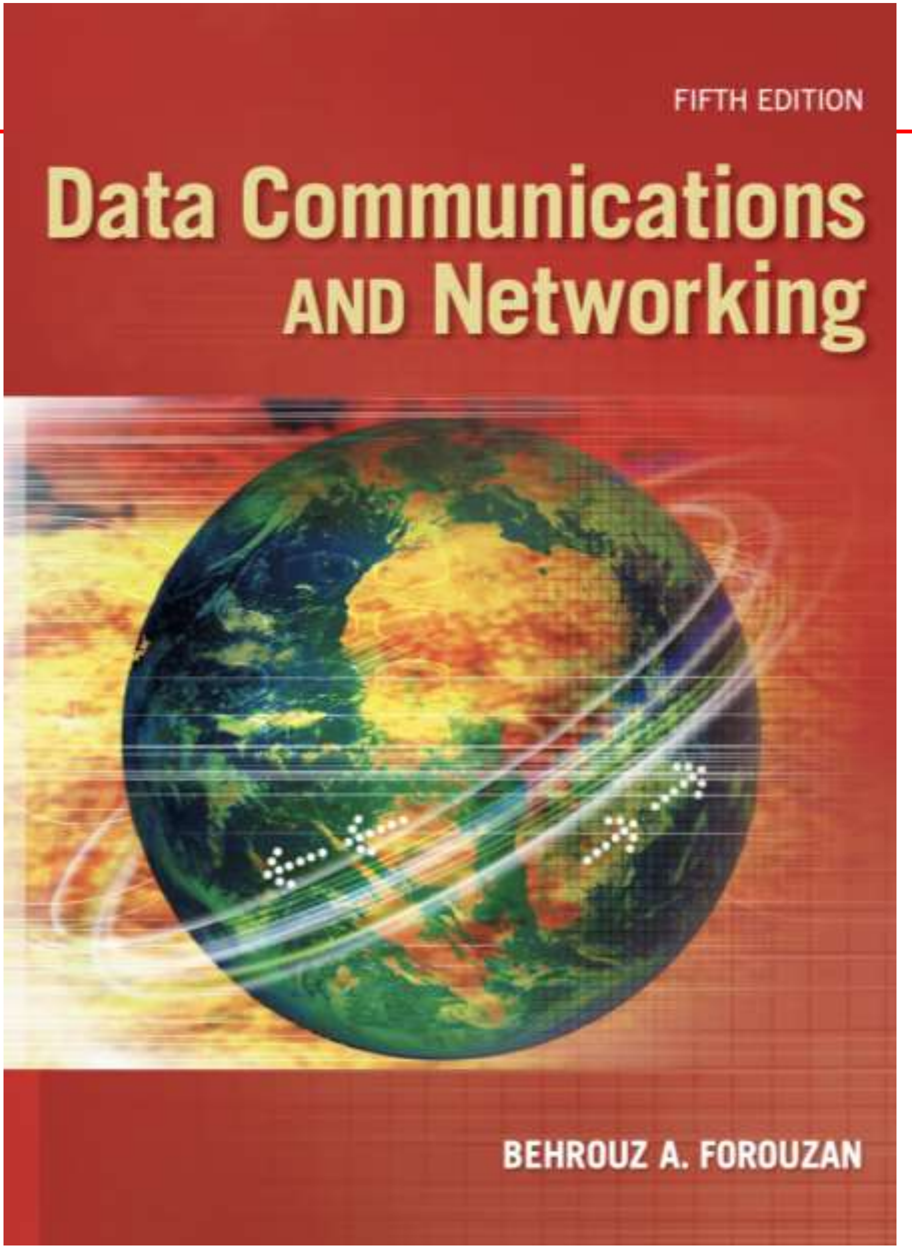


Chapter 4

Digital Transmission

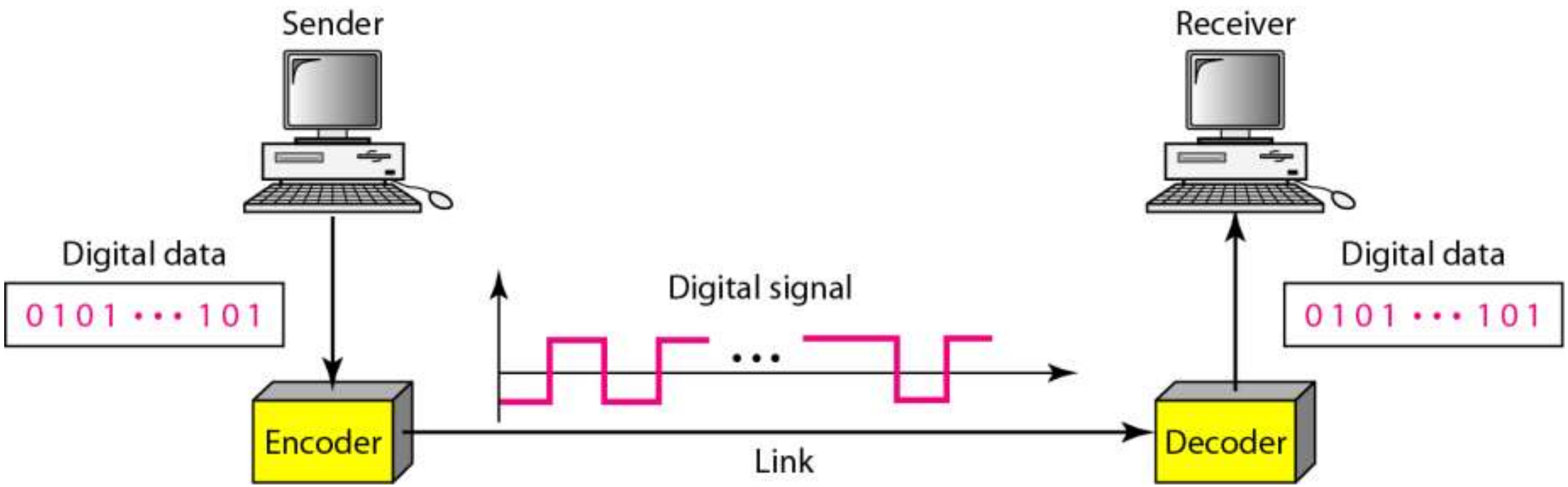
4-1 DIGITAL-TO-DIGITAL CONVERSION

In this section, we see how we can represent digital data by using digital signals. The conversion involves three techniques: line coding, block coding, and scrambling. Line coding is always needed; block coding and scrambling may or may not be needed.

Topics discussed in this section:

Line Coding
Line Coding Schemes
Block Coding
Scrambling

Figure 4.1 *Line coding and decoding*



Line coding is the process of converting digital data to digital signals.

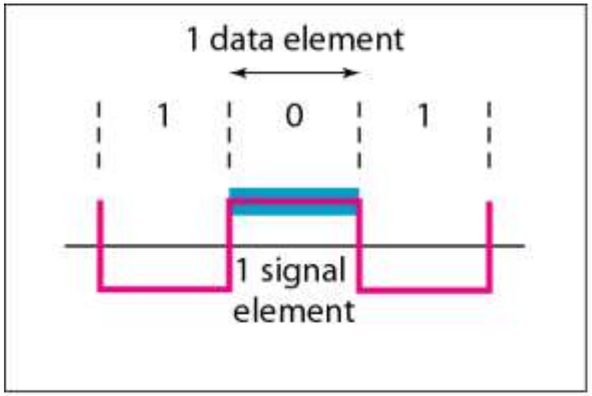
Signal Element versus Data Element

- Data element
  - The smallest entity that can represent a piece of information: this is **bit**.
- Signal element
  - The shortest unit (timewise) of a **digital signal**.

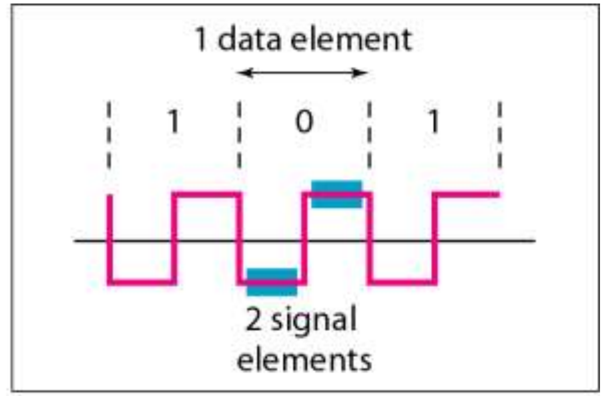
* Data rate and Signal Rate ?

- In other words
  - Data element are what we need to send.
  - Signal elements are what we can send.

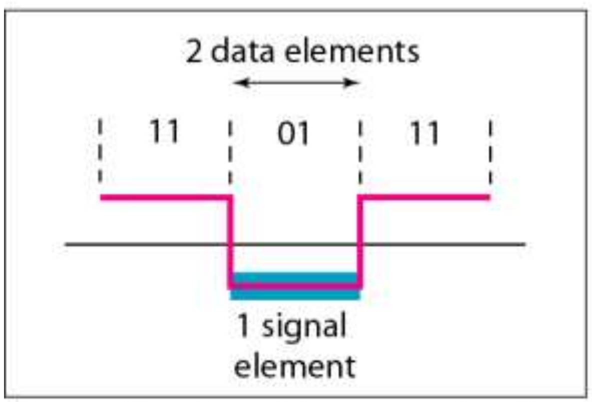
Figure 4.2 *Signal element versus data element*



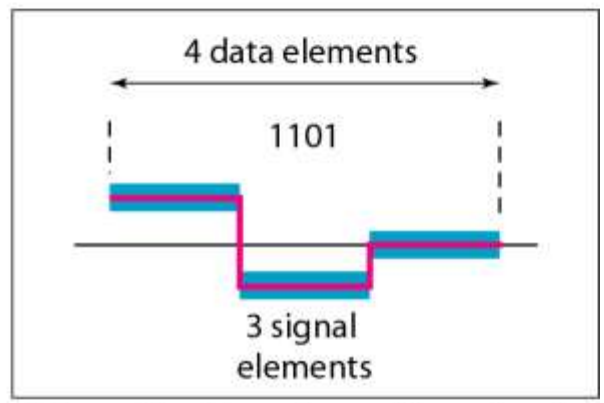
a. One data element per one signal element ($r = 1$)



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



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



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

Data Rate versus Signal Rate

- **Data rate**
  - The number of data elements (bits) sent in 1s
  - The unit is bits per second (bps)
  - Called **bit rate**
- **Signal rate**
  - The number of signal elements sent in 1s
  - The unit is the **baud**
  - Signal rate is sometimes called the **pulse rate**, the **modulation rate**, or the **baud rate**
- Relationship between data rate and signal rate

$$S_{avg} = c \times N \times \frac{1}{r} \qquad \textbf{baud}$$

- *S*: number of signal elements, *c*: the case factor, *N*: data rate (bps), *r*: data elements per signal elements

Example 4.1

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

Solution

We assume that the average value of c is 1/2 . The baud rate is then

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

Bandwidth Requirement

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

The baud rate, not the bit rate, determines the required bandwidth for a digital signal.

More changes in the signal mean injecting more frequencies into the signal.

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

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

Example 4.2

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

Solution

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

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

Baseline Wandering

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

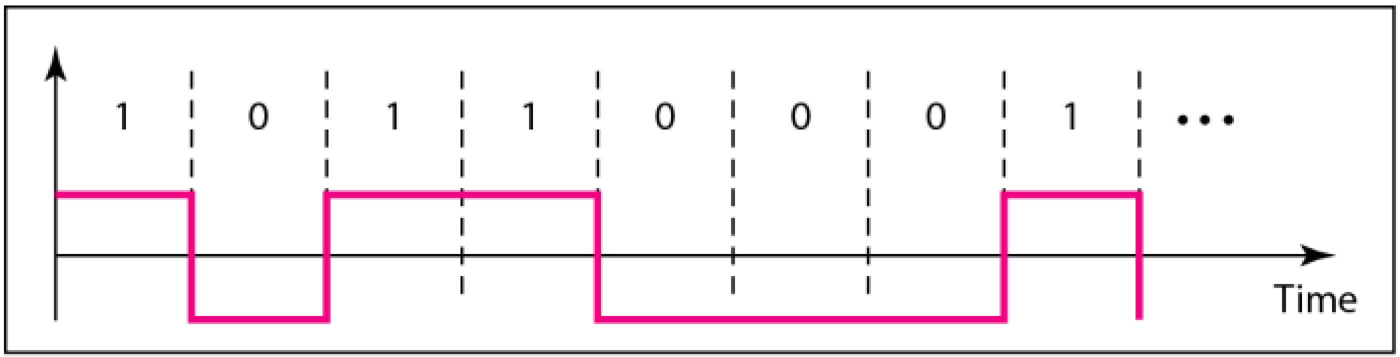
DC Components

- DC Components
  - When the voltage level in a digital signal is constant for a while, the spectrum creates very low frequencies (results of Fourier analysis).
  - These frequencies around zero, call *DC (direct-current) components*, present problems for a system that cannot pass low frequencies or a system that uses electrical coupling (via a transformer).
  - For example, a telephone line cannot pass frequencies below 200 Hz.

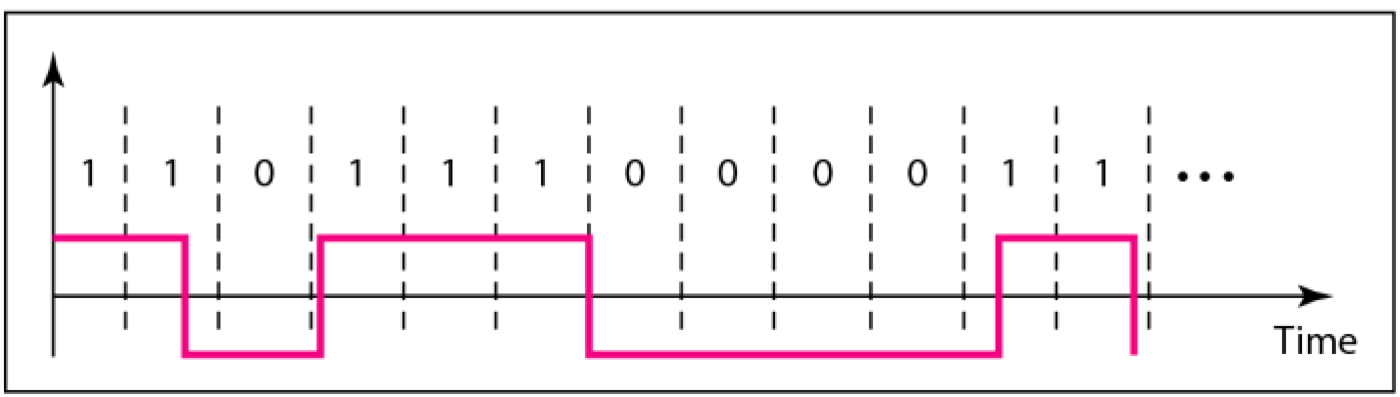
Self-synchronization

- To correctly interpret the signals received from the sender, the receiver's bit intervals must correspond exactly to the sender's bit intervals. If the receiver clock is faster or slower, the bit intervals are not matched and the receiver might misinterpret the signals.
- Self-synchronization
  - Digital signal includes timing information in the data being transmitted.
  - This can be achieved if there are transitions in the signal that alert the receiver to the beginning, middle, or end of the pulse.

Figure 4.3 *Effect of lack of synchronization*



a. Sent



b. Received

Example 4.3

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

Solution

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

1000 bits sent	1001 bits received	1 extra bps

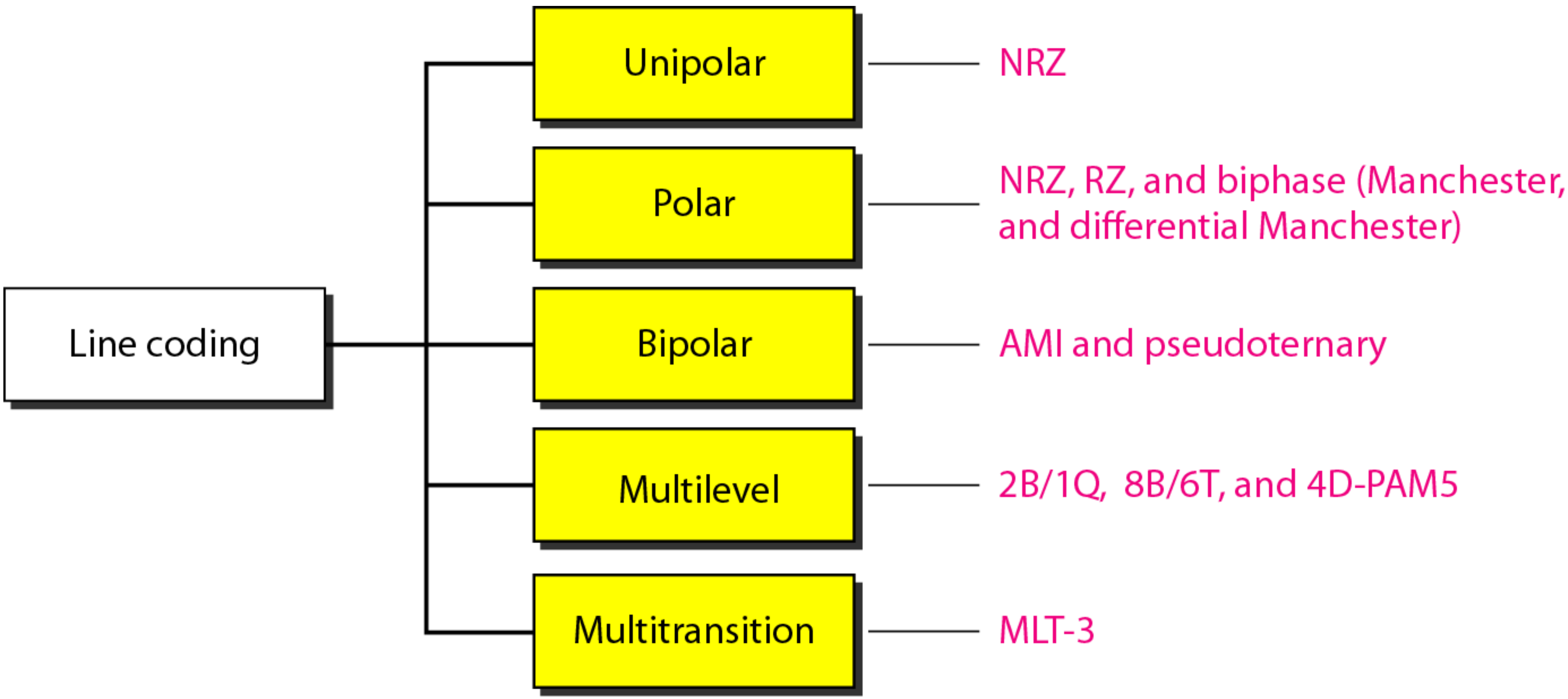
At 1 Mbps, the receiver receives 1,001,000 bps instead of 1,000,000 bps.

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

Other Useful Property

- Built in Error Detection
  - Some coding scheme can detect some errors during transmission.
- Immunity to Noise and Interference
  - Very small effect of noise and interference.
- Complexity
  - Simpler scheme is cheap compared to complex scheme

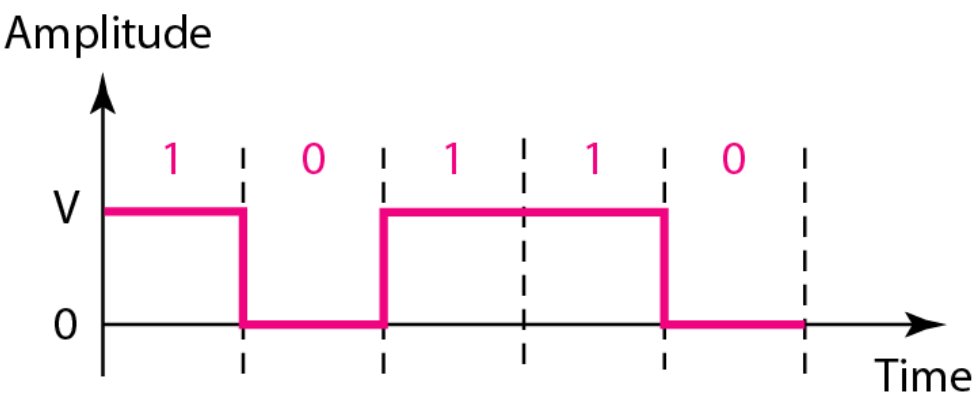
Figure 4.4 *Line coding schemes*




Unipolar

Figure 4.5 *Unipolar NRZ scheme*

Non-Return-to-Zero (NRZ)



$$\frac{1}{2}V^2 + \frac{1}{2}(0)^2 = \frac{1}{2}V^2$$

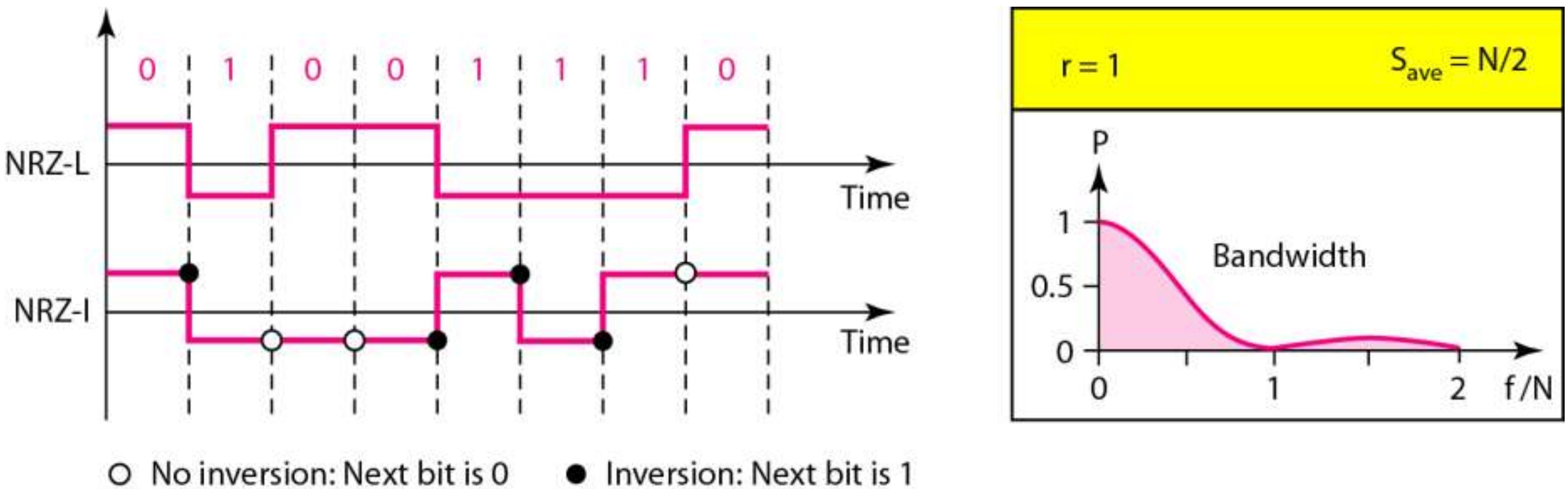
Normalized power

It is called NRZ because the signal does not return to zero at the middle of the bit.

Polar

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

NRZ-L (NRZ-Level), NRZ-I (NRZ-Invert) **(voltages are on the both Sides of the zero line)**



Note

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

Note

NRZ-L and NRZ-I both have an average signal rate of $N/2$ Bd.

Note

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

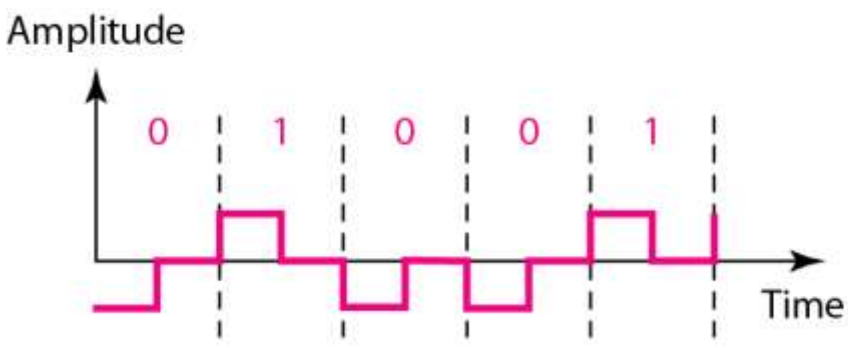
Example 4.4

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

Solution

The average signal rate is S = N/2 = 500 kbaud. The minimum bandwidth for this average baud rate is B_{min} = S = 500 kHz.

RZ: Return-to-Zero



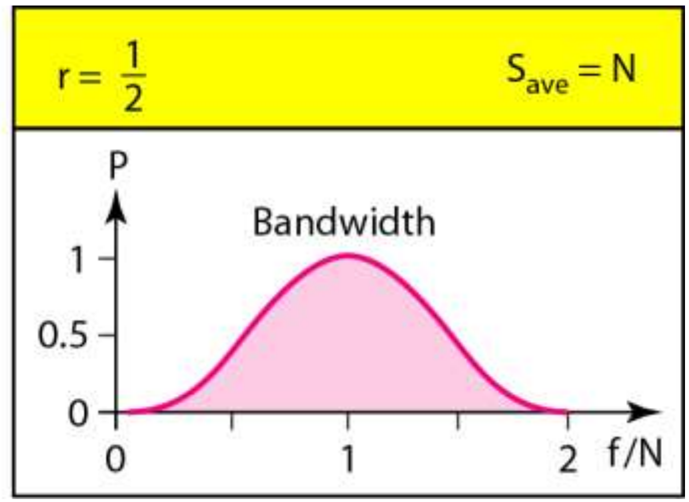
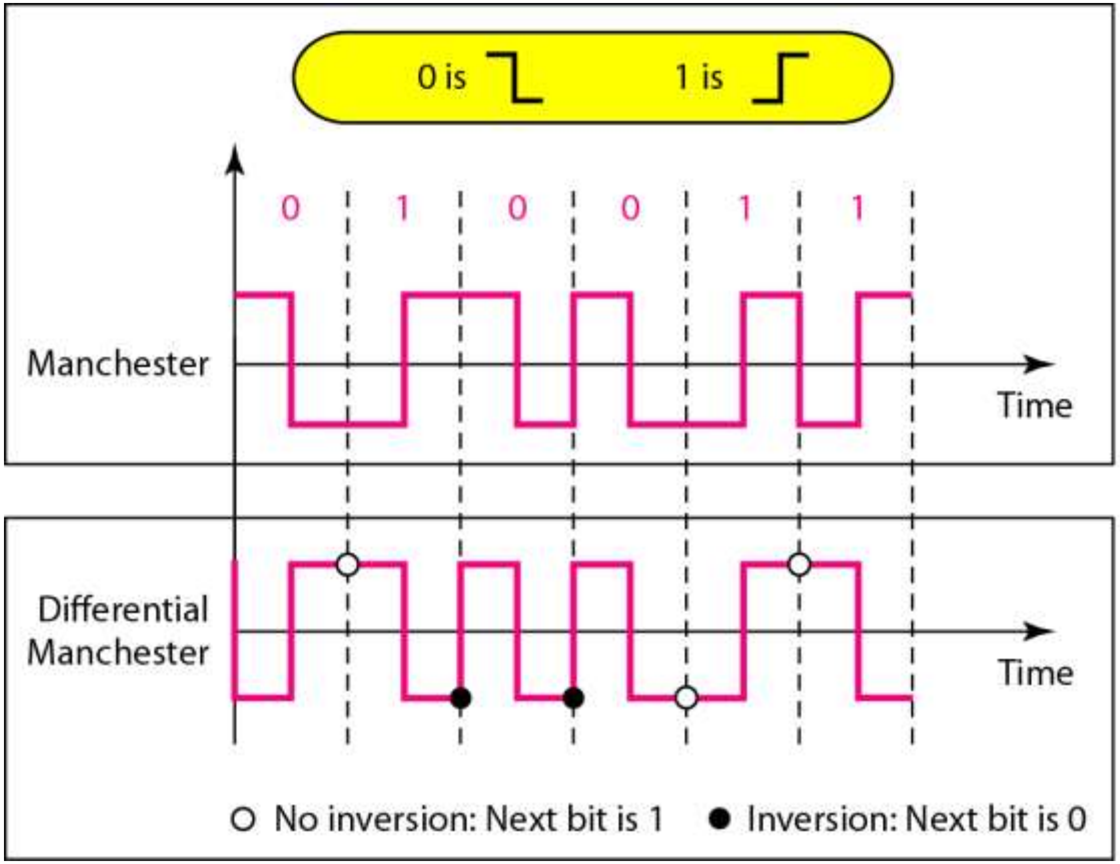
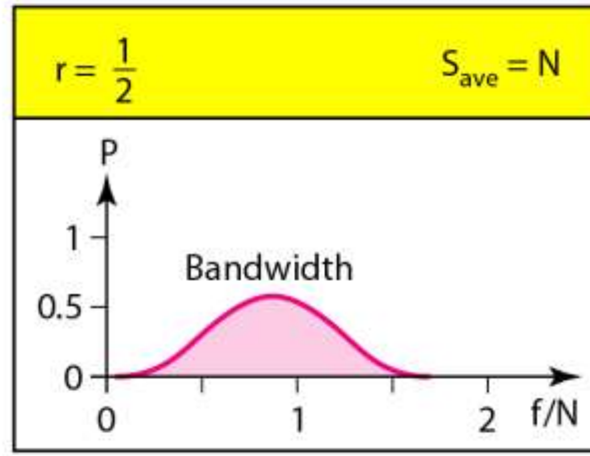


Figure 4.8 *Polar biphase: Manchester and differential Manchester schemes*





Note

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

Note

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

Bipolar

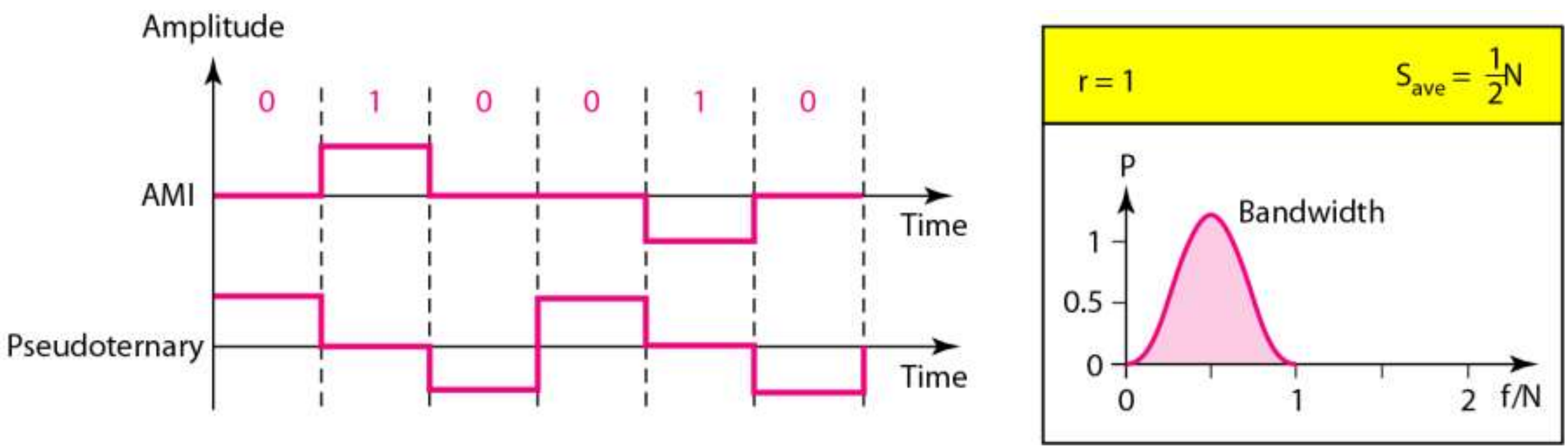
Bipolar Schemes

- Bipolar encoding (sometimes called multilevel binary)
  - Three voltage levels: positive, negative, and zero
- Two variations of bipolar encoding
  - AMI (alternate mark inversion)
    - 0: neutral zero voltage
    - 1: alternating positive and negative voltages
  - Pseudoternary
    - 1: neutral zero voltage
    - 0: alternating positive and negative voltages
- Bipolar schemes have **no** *DC component problem*

AMI and Pseudoternary

- **AMI** (alternate mark inversion)
  - The work ***mark*** comes from telegraphy and means **1**.
  - AMI means alternate 1 inversion
  - The neutral zero voltage represents binary 0.
  - Binary 1s are represented by alternating positive and negative voltages.
- **Pesudotenary**
  - Same as AMI, but 1 bit is encoded as a zero voltage and the 0 bit is encoded as alternating positive and negative voltages.

Figure 4.9 *Bipolar schemes: AMI and pseudoternary*




Multilevel

Multilevel Schemes

- The desire to *increase the data speed* or *decrease the required bandwidth* has resulted in the creation of many schemes.
- The goal is to increase the number of bits per baud by encoding a pattern of ***m*** data elements into a pattern of ***n*** signal elements.
- Different types of signal elements can be allowing different signal levels.
- If we have L different levels, then we can produce L^n combinations of signal patterns.
- The data element and signal element relation is $2^m \leq L^n$
- mBnL coding, where *m* is the length of the binary pattern, *B* means binary data, *n* is the length of the signal pattern, and *L* is the number of levels in the signaling.
- B (binary, L=2), T (tenary, L=3), and Q (quaternary, L=4).

Note

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

2B1Q

- 2B1Q (two binary, one quaternary)
  - m=2, n=1, and L=4
  - The signal rate (baud rate)

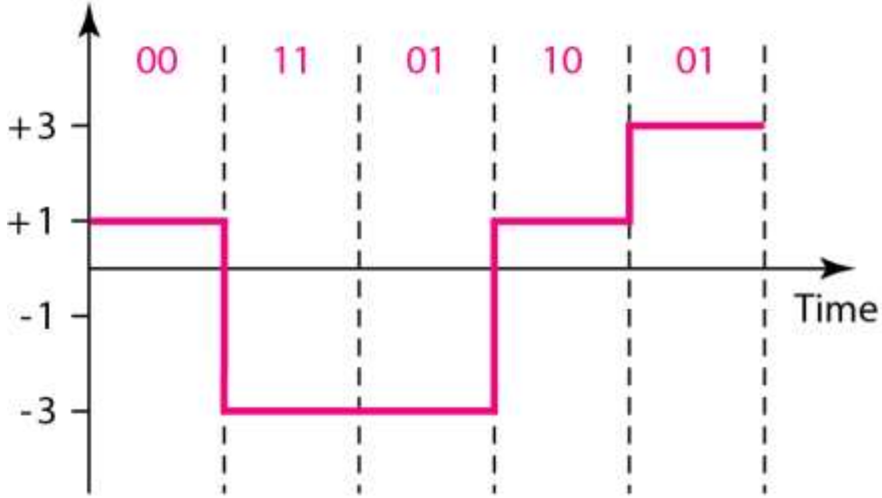
$$S = cN\frac{1}{r} = \frac{1}{2} \times N \times \frac{1}{2} = \frac{N}{4}$$

- 2B1Q is used in DSL (digital subscriber line) technology to provide a high-speed connection to the Internet by using subscriber telephone lines.

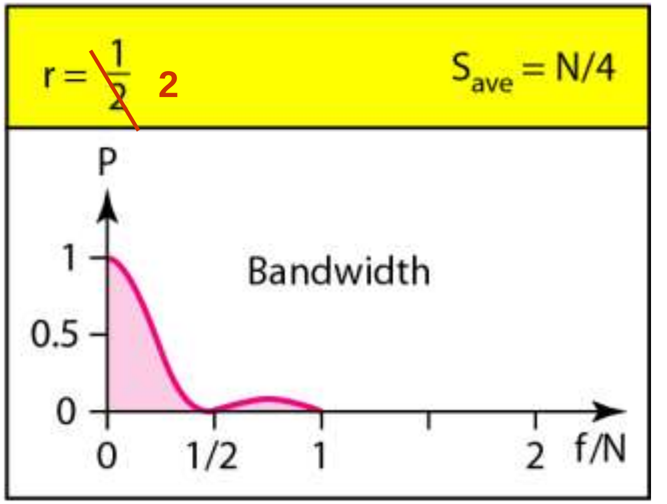
Figure 4.10 *Multilevel: 2B1Q scheme*

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

Transition table



Assuming positive original level



8B6T

- Stands for 8 Binary / 6 Ternary encoding
- Used in 100BASE-4T Ethernet cabling.
- Maps 8-bit binary data to 6-symbol ternary signals (−1, 0, +1).
- Data Pattern $2^8 = 256$. Signal Pattern: $3^6 = 729$.
  $729 - 256 = 473$ redundant unused patterns enabling synchronization and error detection.
- Each signal pattern has a DC weight of 0 or +1. No -1 weight.

8B6T

- Conversation rule:
  - Find the DC weight of the signal patter for next 8bit data from the table
  - To make the whole stream DC-balanced, the sender keeps track of the weight.
  - DC weight is neutral => keep the signal as it is.
  - DC weight is positive and running weight neutral => keep the signal as it is.
  - DC weight is positive and running weight positive => invert the signal.

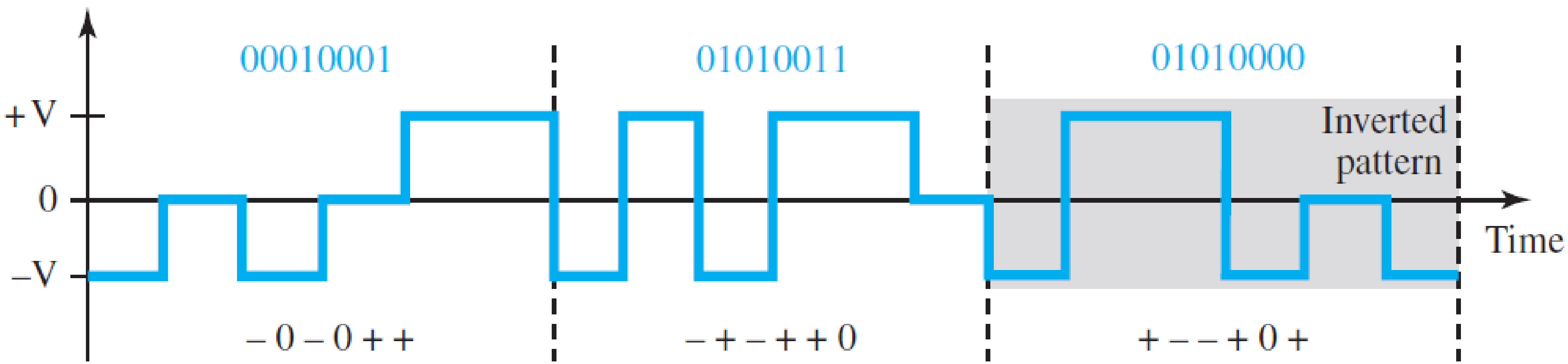
Table F.1 *8B/6T code*

Data	*Code*	*Data*	*Code*	*Data*	*Code*	*Data*	*Code*
00	−+00−+	**20**	−++−00	**40**	−00+0+	**60**	0++0−0
01	0−+−+0	**21**	+00+−−	**41**	0−00++	**61**	+0+−00
02	0−+0−+	**22**	−+0−++	**42**	0−0+0+	**62**	+0+0−0
03	0−++0−	**23**	+−0−++	**43**	0−0++0	**63**	+0+00−
04	−+0+0−	**24**	+−0+00	**44**	−00++0	**64**	0++00−
05	+0−−+0	**25**	−+0+00	**45**	00−0++	**65**	++0−00
06	+0−0−+	**26**	+00−00	**46**	00−+0+	**66**	++00−0
07	+0−+0−	**27**	−+++−−	**47**	00−++0	**67**	++000−
08	−+00+−	**28**	0++−0−	**48**	00+000	**68**	0++−+−
09	0−++−0	**29**	+0+0−−	**49**	++−000	**69**	+0++−−
0A	0−+0+−	**2A**	+0+−0−	**4A**	+−+000	**6A**	+0+−+−
0B	0−+−0+	**2B**	+0+−−0	**4B**	−++000	**6B**	+0+−−+
0C	−+0−0+	**2C**	0++−−0	**4C**	0+−000	**6C**	0++−−+
0D	+0−+−0	**2D**	++00−−	**4D**	+0−000	**6D**	++0+−−
0E	+0−0+−	**2E**	++0−0−	**4E**	0−+000	**6E**	++0−+−
0F	+0−−0+	**2F**	++0−−−0	**4F**	−0+000	**6F**	++0−−+
10	0−−+0+	**30**	+−00−+	**50**	+−−+0+	**70**	000++−
11	−0−0++	**31**	0+−−+0	**51**	−+−0++	**71**	000+−+
12	−0−+0+	**32**	0+−0−+	**52**	−+−+0+	**72**	000−++
13	−0−++0	**33**	0+−+0−	**53**	−+−++0	**73**	000+00
14	0−−++0	**34**	+−0+0−	**54**	+−−++0	**74**	000+0−
15	−−00++	**35**	−0+−+0	**55**	−−+0++	**75**	000+−0
16	−−0+0+	**36**	−0+0−+	**56**	−−++0+	**76**	000−0+
17	−−0++0	**37**	−0++0−	**57**	−−++0	**77**	000−+0
18	−+0−+0	**38**	+−00+−	**58**	−+−0++	**78**	+++−−0

Table F.1 *8B/6T code (continued)*

Data	*Code*	*Data*	*Code*	*Data*	*Code*	*Data*	*Code*
19	+−0−+0	39	0+−+−0	59	−0−+++	79	+++−0−
1A	−++−+0	3A	0+−0+−	5A	0−−+++	7A	+++0−−
1B	+00−+0	3B	0+−−0+	5B	0−−0++	7B	0++0−−
1C	+00+−0	3C	+−0−0+	5C	+−−0++	7C	−00−++
1D	−+++−0	3D	−0++−0	5D	−000++	7D	−00+00
1E	+−0+−0	3E	−0+0+−	5E	0+++−−	7E	+−−−++
1F	−+0+−0	3F	−0+−0+	5F	0++−00	7F	+−−+00
80	−00+−+	A0	−++0−0	C0	−+0+−+	E0	−++0−+
81	0−0−++	A1	+−+−00	C1	0−+−++	E1	+−−++0
82	0−0+−+	A2	+−+0−0	C2	0−++−+	E2	+−+0−+
83	0−0++−	A3	+−+00−	C3	0−+++−	E3	+−++0−
84	−00++−	A4	−++00−	C4	−+0++−	E4	−+++0−
85	00−−++	A5	++−−00	C5	+0−−++	E5	++−−+0
86	00−+−+	A6	++−0−0	C6	+0−+−+	E6	++−0−+
87	00−++−	A7	++−00−	C7	+0−++−	E7	++−+0−
88	−000+0	A8	−++−+−	C8	−+00+0	E8	−++0+−
89	0−0+00	A9	+−++−−	C9	0−++00	E9	+−++−0
8A	0−00+0	AA	+−+−+−	CA	0−+0+0	EA	+−+0+−
8B	0−000+	AB	+−+−−+	CB	0−+00+	EB	+−+−0+
8C	−0000+	AC	−++−−+	CC	−+000+	EC	−++−0+
8D	00−+00	AD	++−+−−	CD	+0−+00	ED	++−+−0
8E	00−0+0	AE	++−−+−	CE	+0−0+0	EE	++−0+−
8F	00−00+	AF	++−−−+	CF	+0−00+	EF	++−−0+
90	+−−+−+	B0	+000−0	D0	+−0+−+	F0	+000−+
91	−+−−++	B1	0+0−00	D1	0+−−++	F1	0+0−+0
92	−+−+−+	B2	0+00−0	D2	0+−+−+	F2	0+00−+
93	−+−++−	B3	0+000−	D3	0+−++−	F3	0+0+0−
94	+−−++−	B4	−0000−	D4	+−0++−	F4	+00+0−
95	−−+−++	B5	00+−00	D5	−0+−++	F5	00+−+0
96	−−++−+	B6	00+0−0	D6	−0++−+	F6	00+0−+
97	−−+++−	B7	00+00−	D7	−0+++−	F7	00++0−
98	+−−0+0	B8	+00−+−	D8	+−00+0	F8	+000+−
99	−+−+00	B9	0+0+−−	D9	0+−+00	F9	0+0+−0
9A	−+−0+0	BA	0+0−+−	DA	0+−0+0	FA	0+00+−
9B	−+−00+	BB	0+0−−+	DB	0+−00+	FB	0+0−0+
9C	+−−00+	BC	+00−−+	DC	+−000+	FC	+00−0+
9D	−−++00	BD	00++−−	DD	−0++00	FD	00++−0
9E	−−+0+0	BE	00+−+−	DE	−0+0+0	FE	00+0+−
9F	−−+00+	BF	00+−−+	DF	−0+00+	FF	00+−0+

Figure 4.11 Multilevel: 8B6T scheme



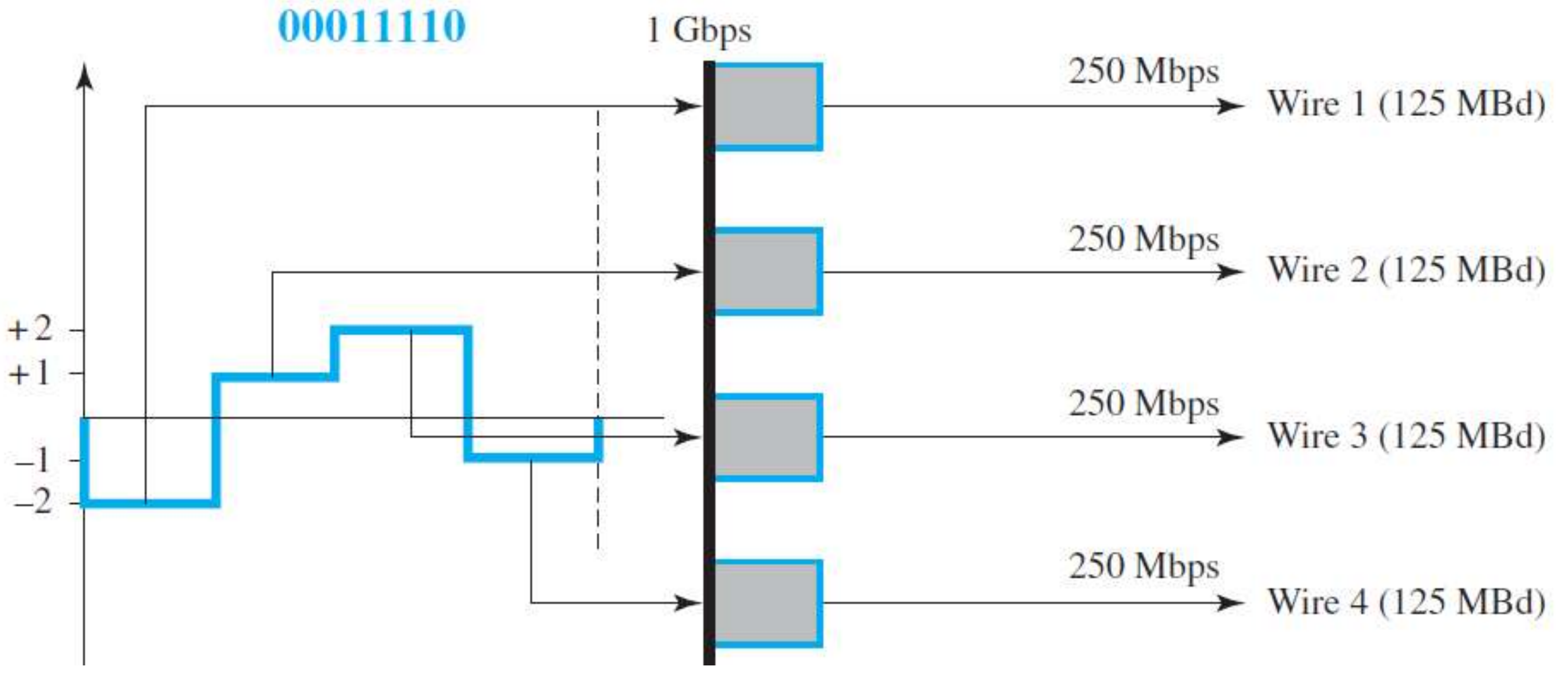
4D-PAM5

- Four-Dimensional, Five-Level Pulse Amplitude Modulation
- Used in: Gigabit Ethernet (1 Gbps over copper cables).

- Uses 4 wires (channels) to transmit data simultaneously.
- Voltage levels are −2, −1, 0, +1, +2.Level 0 is reserved for error detection only.

- 256 signal pattern (4 wires × 4 levels = 4^4).
- Only 256 data patterns (2^8),
- Redundant signal patterns used in error detection and Synchronization.

4D-PAM5



- Conversion Rule
  - Each 8-bit word mapped to a group of 4 signal elements, one per wire.
  - In a 1D view, it resembles 8B4Q (8 bits to 4-level signals).
  - In a 4D setting, signal rate is significantly reduced.
  - From N/2 (1D) to N/8 (4D), improving bandwidth efficiency.

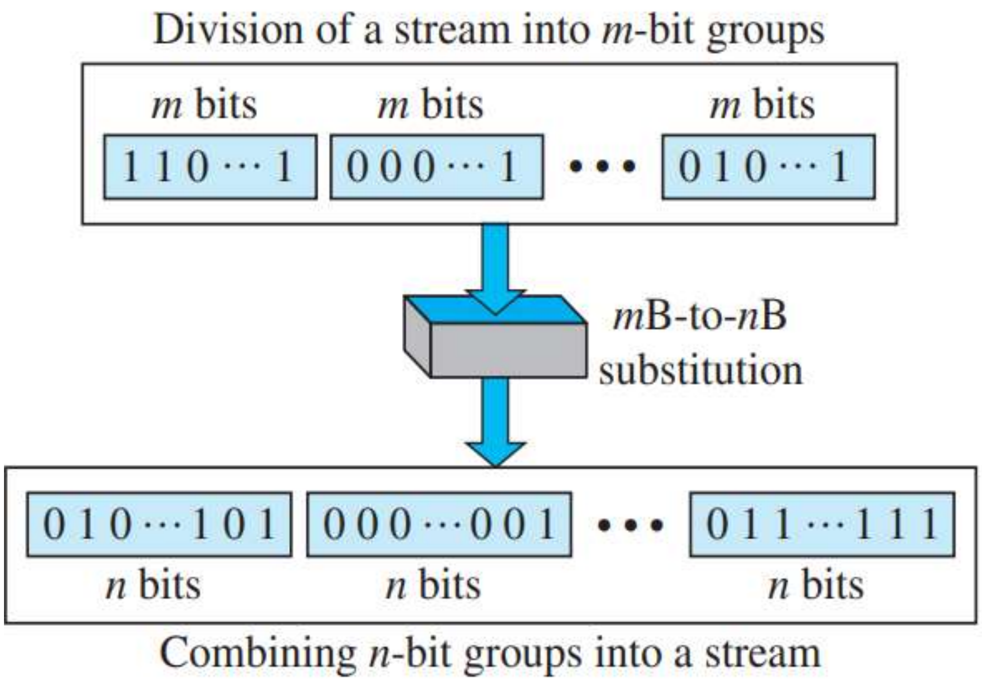
Summary

Category	*Scheme*	*Bandwidth (average)*	*Characteristics*
Unipolar	NRZ	$B = N/2$	Costly, no self-synchronization if long 0s or 1s, DC
Polar	NRZ-L	$B = N/2$	No self-synchronization if long 0s or 1s, DC
	NRZ-I	$B = N/2$	No self-synchronization for long 0s, DC
	Biphase	$B = N$	Self-synchronization, no DC, high bandwidth
Bipolar	AMI	$B = N/2$	No self-synchronization for long 0s, DC
Multilevel	2B1Q	$B = N/4$	No self-synchronization for long same double bits
	8B6T	$B = 3N/4$	Self-synchronization, no DC
	4D-PAM5	$B = N/8$	Self-synchronization, no DC
Multitransition	MLT-3	$B = N/3$	No self-synchronization for long 0s

Block Coding

- Adds redundancy to ensure synchronization and error detection.
- Enhances performance of line coding.

Figure 4.14 Block coding concept

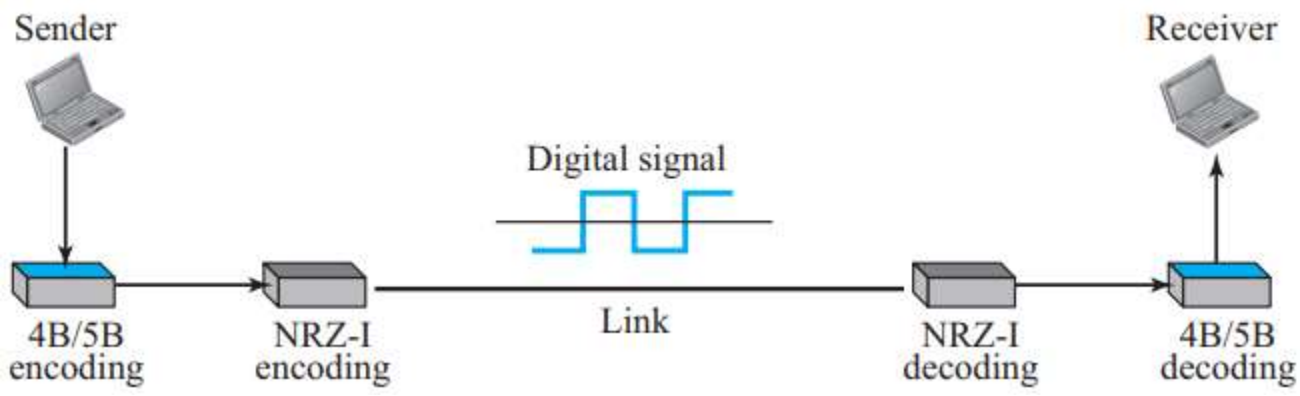


mB/nB Encoding

- Converts m-bit blocks into n-bit blocks (where n > m).

4B/5B Block Coding

- It is designed to work with NRZ-I.
- NRZ-I offers a good signal rate but
  suffers from synchronization issues due to long sequences of 0s.
- 4B/5B coding is used before NRZ-I to eliminate long runs of 0s.



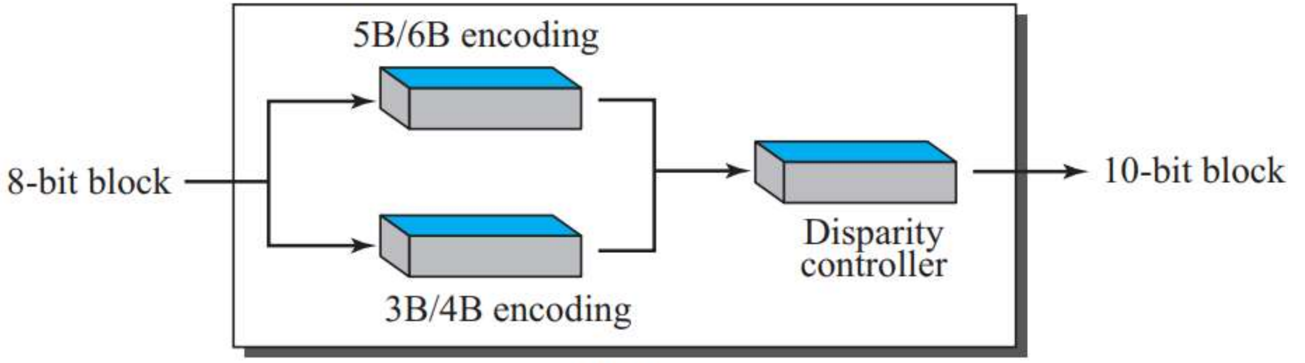
- 4-bit input → 16 combinations.
  5-bit output → 32 combinations.
  Unused 5-bit groups:
  - Some reserved for control.
  - Others left unused → help in error detection.
  - Arrival of an invalid 5-bit group indicates transmission error.

4B/5B mapping codes

Data Sequence	*Encoded Sequence*	*Control Sequence*	*Encoded Sequence*
0000	11110	Q (Quiet)	00000
0001	01001	I (Idle)	11111
0010	10100	H (Halt)	00100
0011	10101	J (Start delimiter)	11000
0100	01010	K (Start delimiter)	10001
0101	01011	T (End delimiter)	01101
0110	01110	S (Set)	11001
0111	01111	R (Reset)	00111
1000	10010		
1001	10011		
1010	10110		
1011	10111		
1100	11010		
1101	11011		
1110	11100		
1111	11101		

8B/10B Block Coding

- Similar to 4B/5B, but uses: 8-bit input → 10-bit output
- Offers better error detection and synchronization than 4B/5B.



8B/10B encoder

- 8B/10B is split into: 5B/6B for the 5 most significant bits 3B/4B for the 3 least significant bits
- Splitting simplifies the mapping process.

Disparity Control

- Prevents long runs of 0s or 1s (bad for synchronization).
- Uses a disparity controller:
    - Tracks the balance between 0s and 1s over multiple blocks.
    - If imbalance occurs in current block (creates disparity), complements the bits (0 → 1, 1 → 0).
    - Maintains DC balance and improves signal quality.

Scrambling

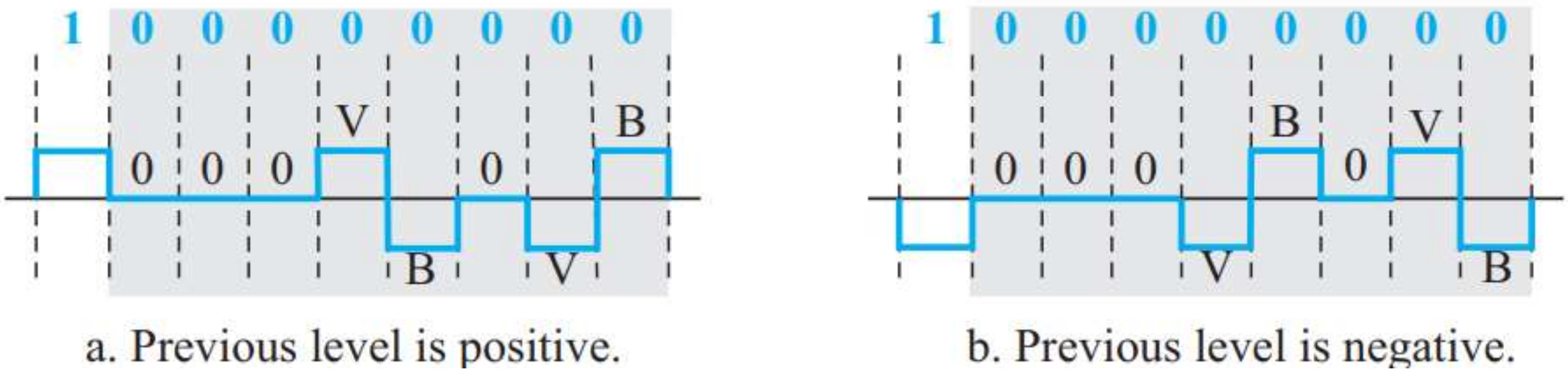
- A solution that substitutes long zero-level pulses with a combination of other levels to provide synchronization.
- Done during encoding, not as a separate block coding stage.

- **Common Scrambling Techniques**
  - *B8ZS (Bipolar with 8-Zero Substitution)*
  - *HDB3 (High-Density Bipolar 3-Zeros)*

B8ZS

- Eight consecutive zero-level voltages are replaced by the sequence 000VB0VB.
- The V in the sequence denotes violation; this is a nonzero voltage that breaks an AMI rule of encoding

Figure 4.19 Two cases of B8ZS scrambling technique

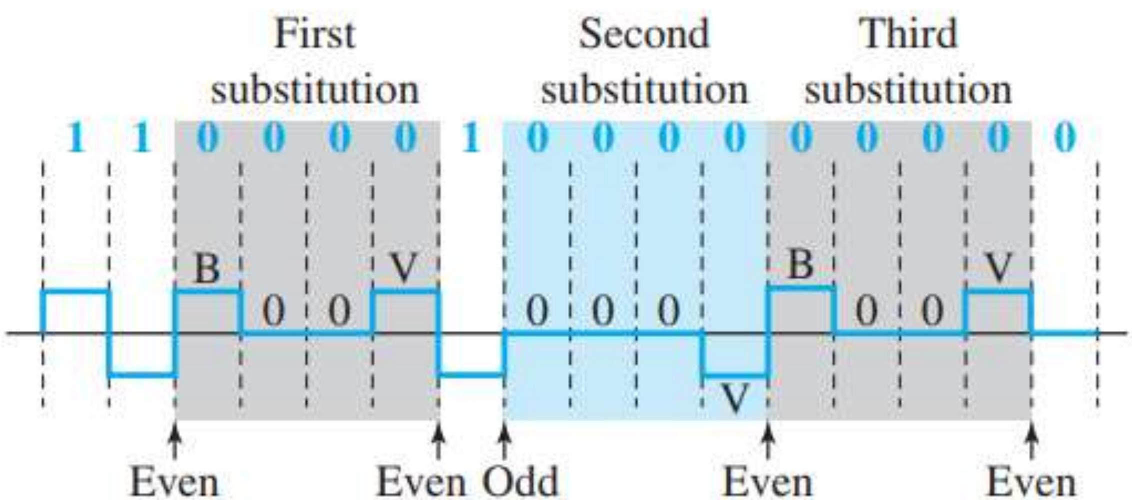


- The letter V (violation) or B (bipolar) here is relative. The V means the same polarity as the polarity of the previous nonzero pulse; B means the polarity opposite to the polarity of the previous nonzero pulse.

HDB3

- Four consecutive zero-level voltages are replaced with a sequence of 000V or B00V.
- Two rules can be stated as follows:
  1. If the number of nonzero pulses after the last substitution is odd, the substitution pattern will be 000V, which makes the total number of nonzero pulses even.
  2. If the number of nonzero pulses after the last substitution is even, the substitution pattern will be B00V, which makes the total number of nonzero pulses even.

Figure 4.20 Different situations in HDB3 scrambling technique



4-2 ANALOG-TO-DIGITAL CONVERSION

We have seen in Chapter 3 that a digital signal is superior to an analog signal. The tendency today is to change an analog signal to digital data. In this section we describe two techniques, pulse code modulation and delta modulation.

Topics discussed in this section:

Pulse Code Modulation (PCM)
Delta Modulation (DM)

Figure 4.21 *Components of PCM encoder*

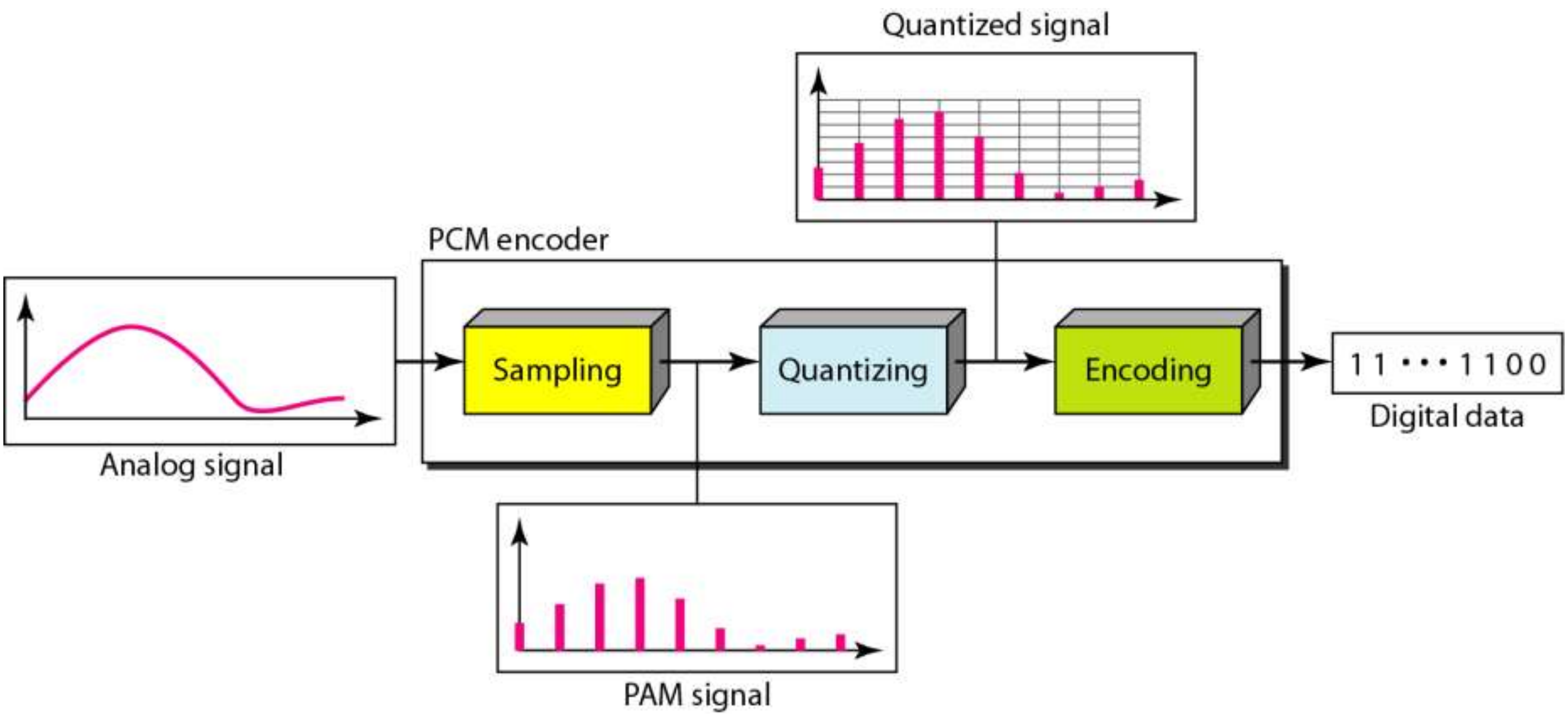
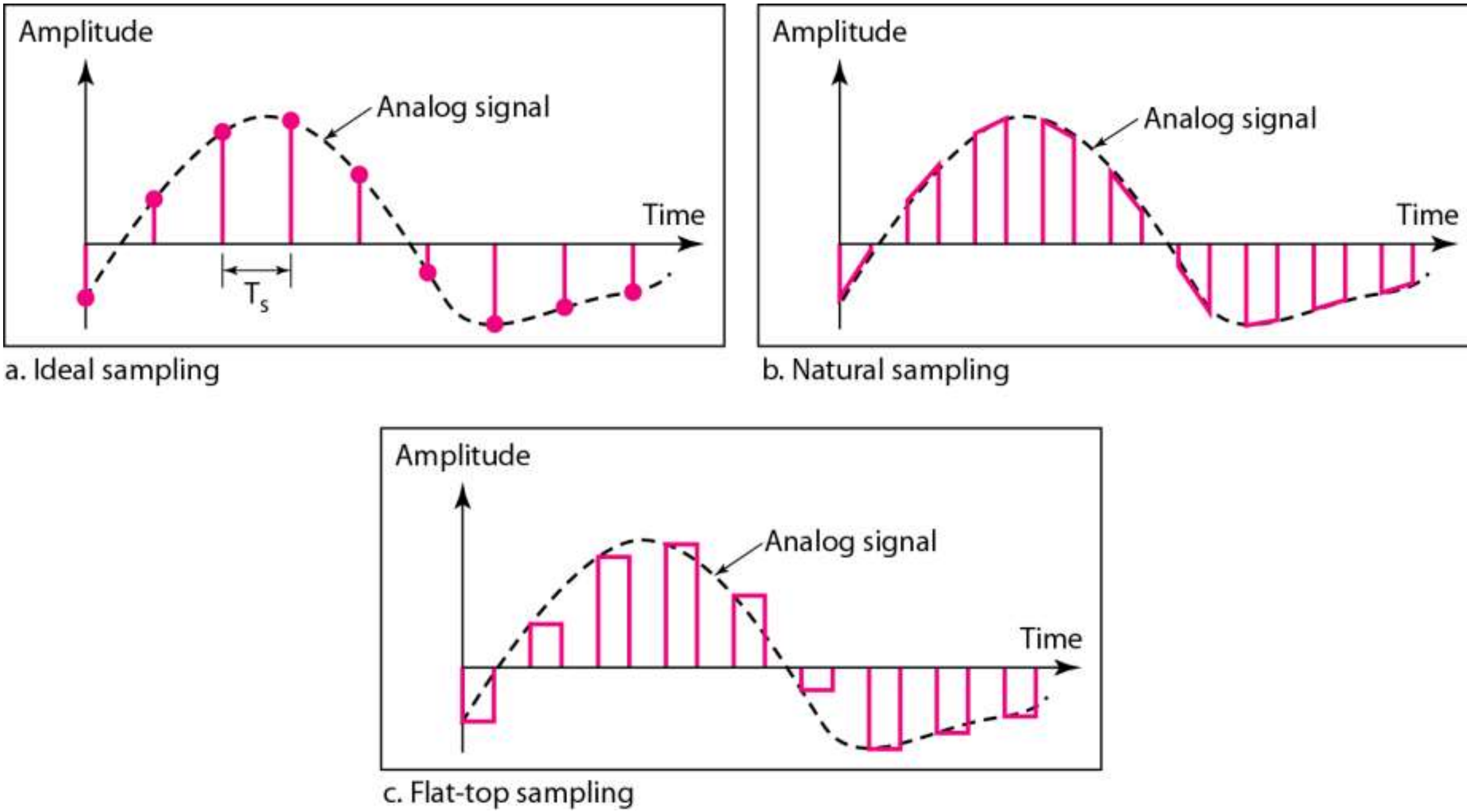


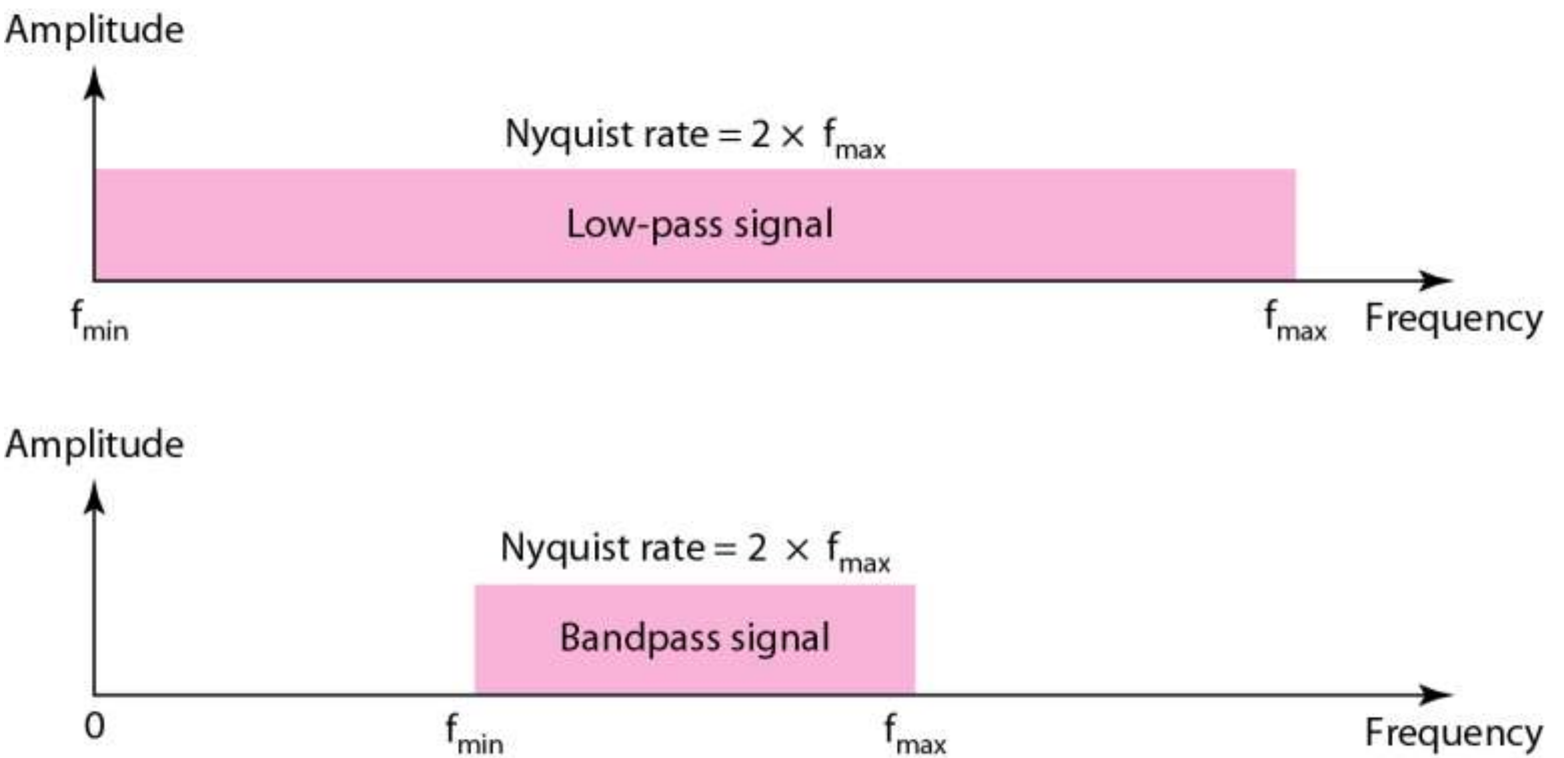
Figure 4.22 *Three different sampling methods for PCM*



Note

According to the Nyquist theorem, the sampling rate must be at least 2 times the highest frequency contained in the signal.

Figure 4.23 *Nyquist sampling rate for low-pass and bandpass signals*

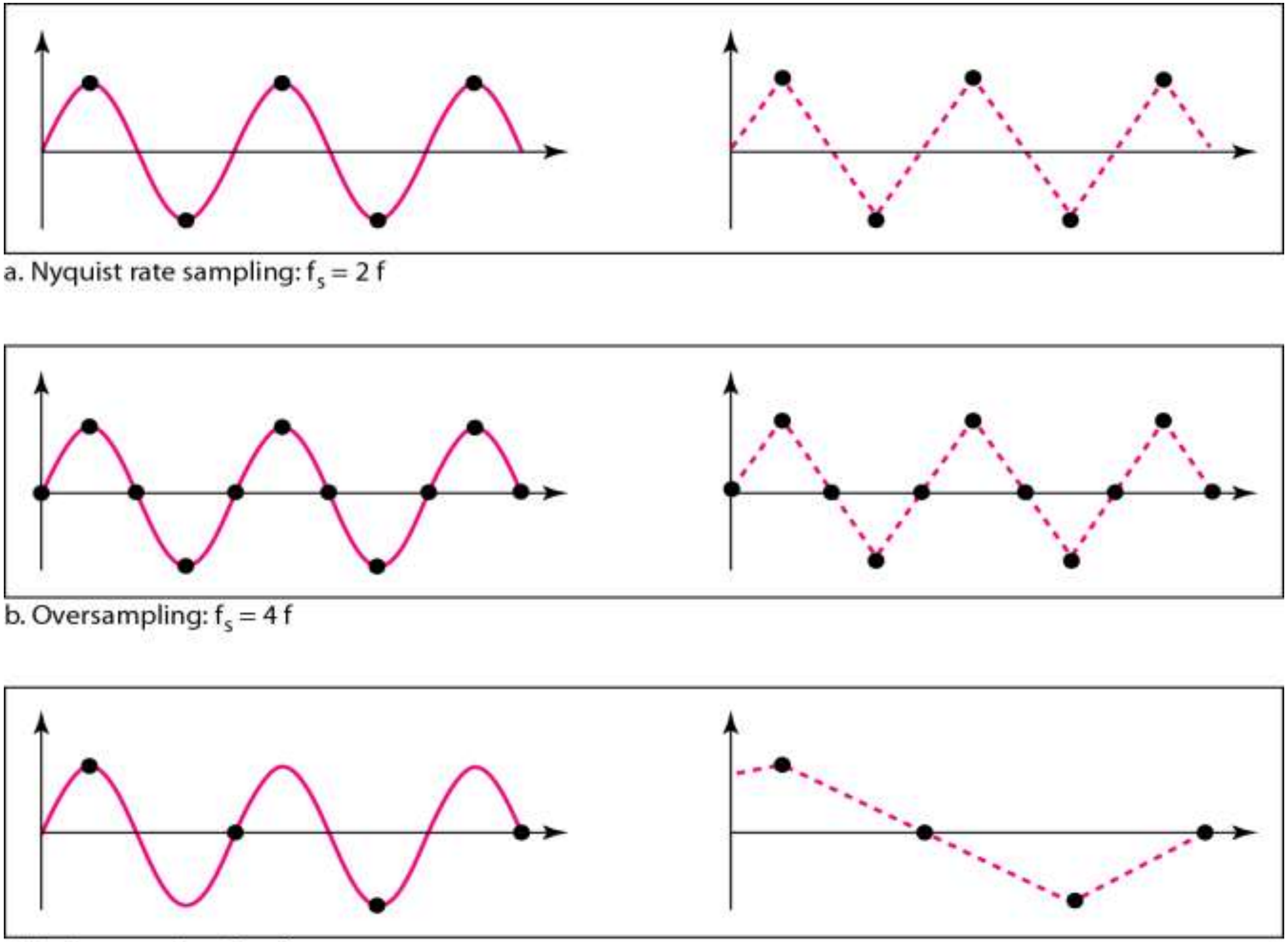


Example 4.6

For an intuitive example of the Nyquist theorem, let us sample a simple sine wave at three sampling rates: $f_s = 4f$ (2 times the Nyquist rate), $f_s = 2f$ (Nyquist rate), and $f_s = f$ (one-half the Nyquist rate). Figure 4.24 shows the sampling and the subsequent recovery of the signal.

It can be seen that sampling at the Nyquist rate can create a good approximation of the original sine wave (part a). Oversampling in part b can also create the same approximation, but it is redundant and unnecessary. Sampling below the Nyquist rate (part c) does not produce a signal that looks like the original sine wave.

Figure 4.24 *Recovery of a sampled sine wave for different sampling rates*



a. Nyquist rate sampling: $f_s = 2f$

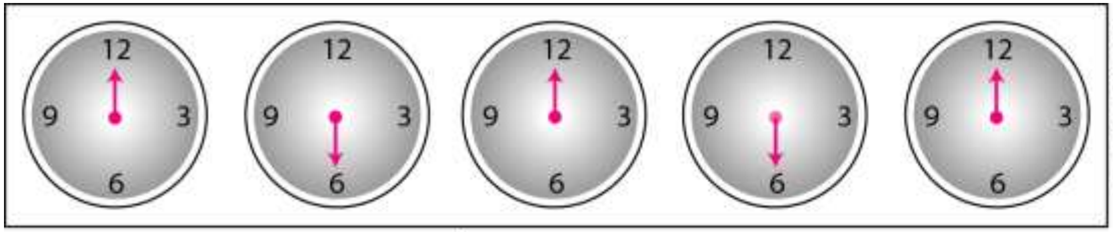
b. Oversampling: $f_s = 4f$

c. Undersampling: $f_s = f$

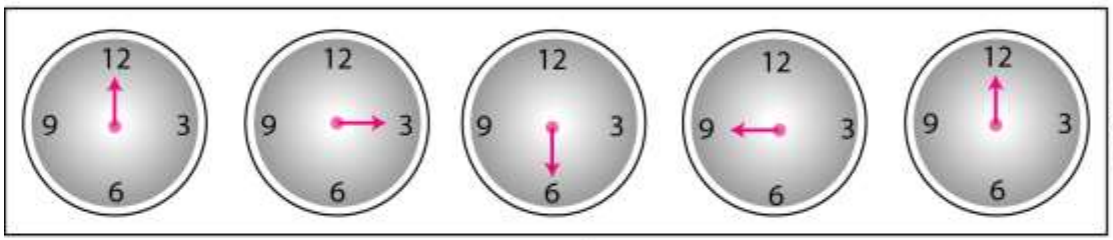
Example 4.7

Consider the revolution of a hand of a clock. The second hand of a clock has a period of 60 s. According to the Nyquist theorem, we need to sample the hand every 30 s ($T_s = T$ or $f_s = 2f$). In Figure 4.25a, the sample points, in order, are 12, 6, 12, 6, 12, and 6. The receiver of the samples cannot tell if the clock is moving forward or backward. In part b, we sample at double the Nyquist rate (every 15 s). The sample points are 12, 3, 6, 9, and 12. The clock is moving forward. In part c, we sample below the Nyquist rate ($T_s = T$ or $f_s = f$). The sample points are 12, 9, 6, 3, and 12. Although the clock is moving forward, the receiver thinks that the clock is moving backward.

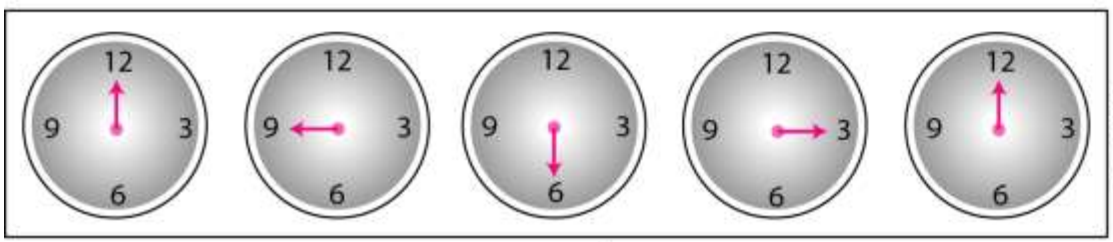
Figure 4.25 *Sampling of a clock with only one hand*



a. Sampling at Nyquist rate: $T_s = T\frac{1}{2}$



b. Oversampling (above Nyquist rate): $T_s = T\frac{1}{4}$



c. Undersampling (below Nyquist rate): $T_s = T\frac{3}{4}$

Example 4.8

An example related to Example 4.7 is the seemingly backward rotation of the wheels of a forward-moving car in a movie. This can be explained by under-sampling. A movie is filmed at 24 frames per second. If a wheel is rotating more than 12 times per second, the under-sampling creates the impression of a backward rotation.

Example 4.9

Telephone companies digitize voice by assuming a maximum frequency of 4000 Hz. The sampling rate therefore is 8000 samples per second.

Example 4.10

A complex low-pass signal has a bandwidth of 200 kHz. What is the minimum sampling rate for this signal?

Solution

The bandwidth of a low-pass signal is between 0 and f, where f is the maximum frequency in the signal. Therefore, we can sample this signal at 2 times the highest frequency (200 kHz). The sampling rate is therefore 400,000 samples per second.

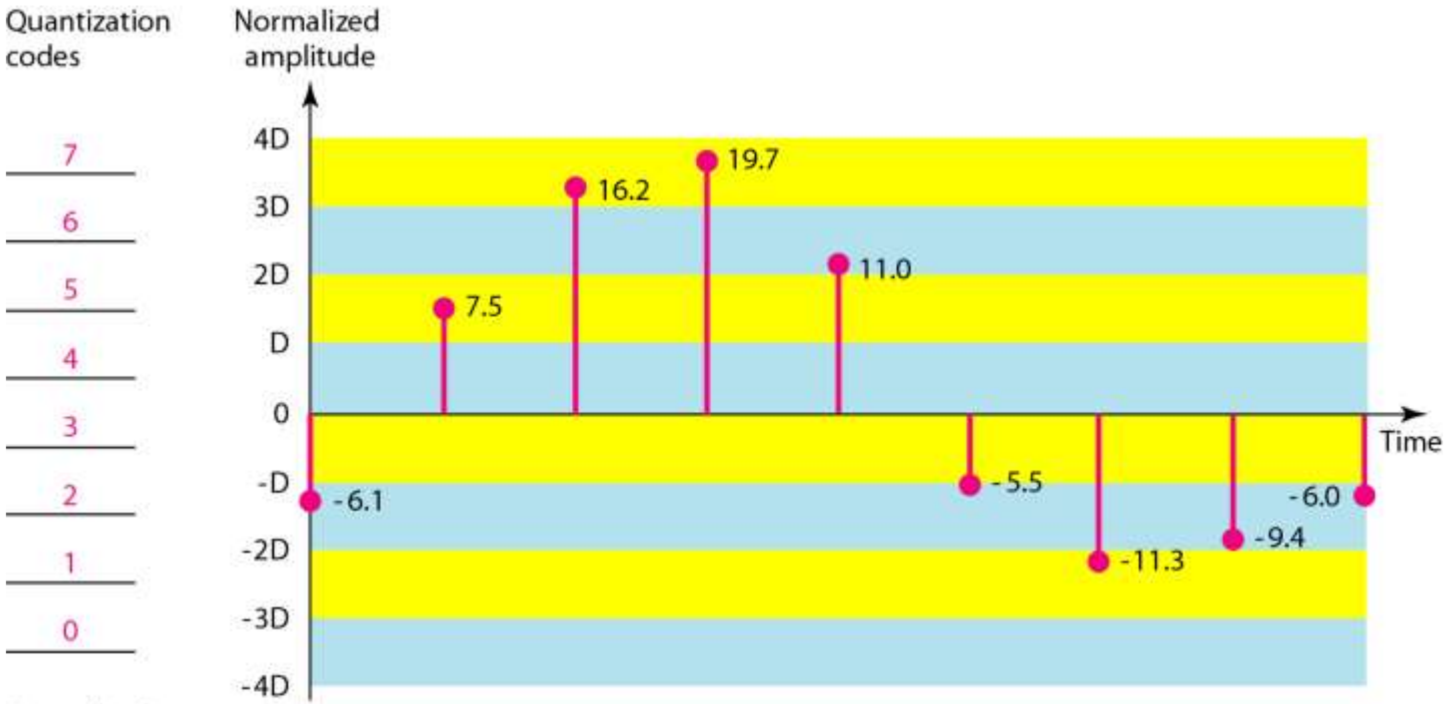
Example 4.11

A complex bandpass signal has a bandwidth of 200 kHz. What is the minimum sampling rate for this signal?

Solution

We cannot find the minimum sampling rate in this case because we do not know where the bandwidth starts or ends. We do not know the maximum frequency in the signal.

Figure 4.26 *Quantization and encoding of a sampled signal*



Normalized PAM values	-1.22	1.50	3.24	3.94	2.20	-1.10	-2.26	-1.88	-1.20
Normalized quantized values	-1.50	1.50	3.50	3.50	2.50	-1.50	-2.50	-1.50	-1.50
Normalized error	-0.38	0	+0.26	-0.44	+0.30	-0.40	-0.24	+0.38	-0.30
Quantization code	2	5	7	7	6	2	1	2	2
Encoded words	010	101	111	111	110	010	001	010	010

Quantization

- Quantization Levels
  - Choosing lower values of L increases the quantization error if there is a lot of fluctuation in the signal.
- Quantization Error
  - If the input value is also at the middle of the zone, there is no quantization error; otherwise, there is an error.
  - It can be proven that the contribution of the quantization error to the SNR_{dB} of the signal depends on the number of quantization levels L, or the bits per sample n_b, as shown in the following formula:

$$SNR_{dB} = 6.02n_b + 1.76 \quad dB$$

Example 4.12

What is the SNR_{dB} in the example of Figure 4.26?

Solution

We can use the formula to find the quantization. We have eight levels and 3 bits per sample, so

$$SNR_{dB} = 6.02(3) + 1.76 = 19.82 \text{ dB}$$

Increasing the number of levels increases the SNR.

Example 4.13

A telephone subscriber line must have an SNR_{dB} above 40. What is the minimum number of bits per sample?

Solution

We can calculate the number of bits as

$$\mathrm{SNR}_{\mathrm{dB}} = 6.02 n_b + 1.76 = 40 \longrightarrow n = 6.35$$

Telephone companies usually assign 7 or 8 bits per sample.

Example 4.14

We want to digitize the human voice. What is the bit rate, assuming 8 bits per sample?

Solution

The human voice normally contains frequencies from 0 to 4000 Hz. So the sampling rate and bit rate are calculated as follows:

$$\text{Sampling rate} = 4000 \times 2 = 8000 \text{ samples/s}$$
$$\text{Bit rate} = 8000 \times 8 = 64{,}000 \text{ bps} = 64 \text{ kbps}$$

Figure 4.27 *Components of a PCM decoder*

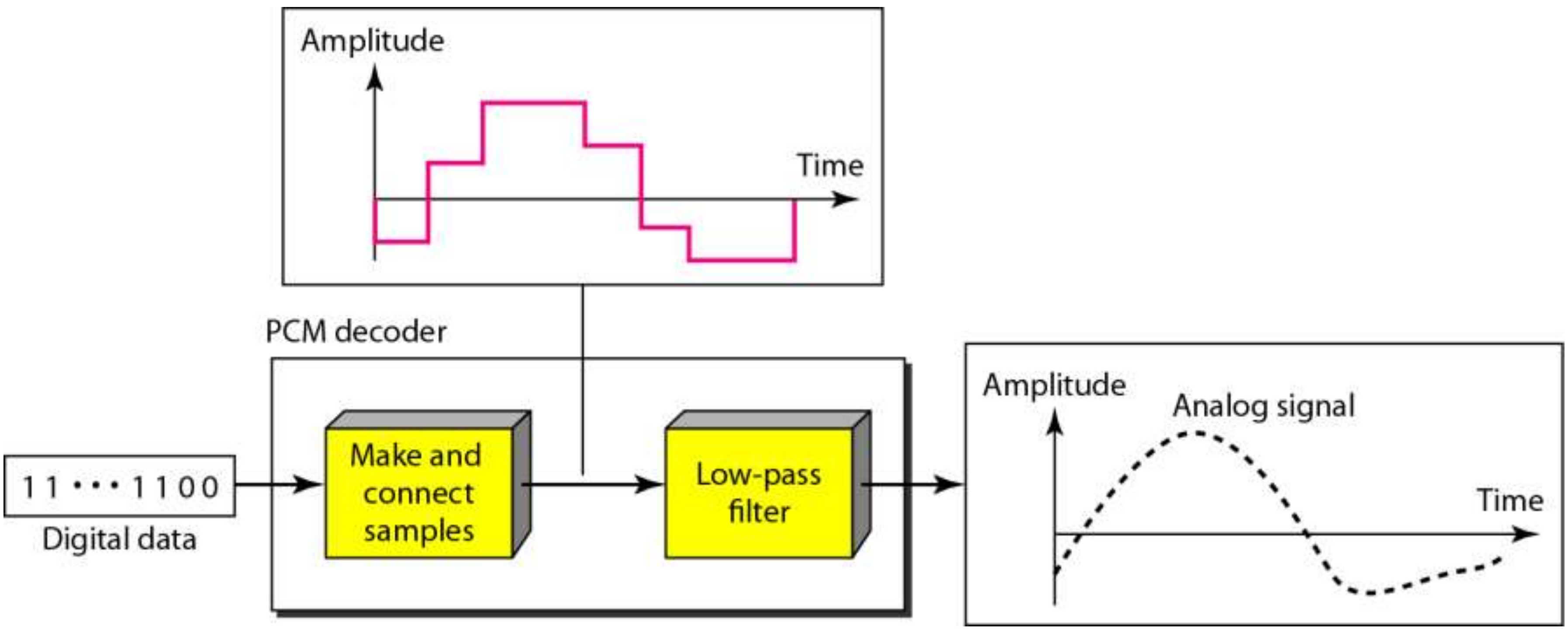


Figure 4.28 *The process of delta modulation*

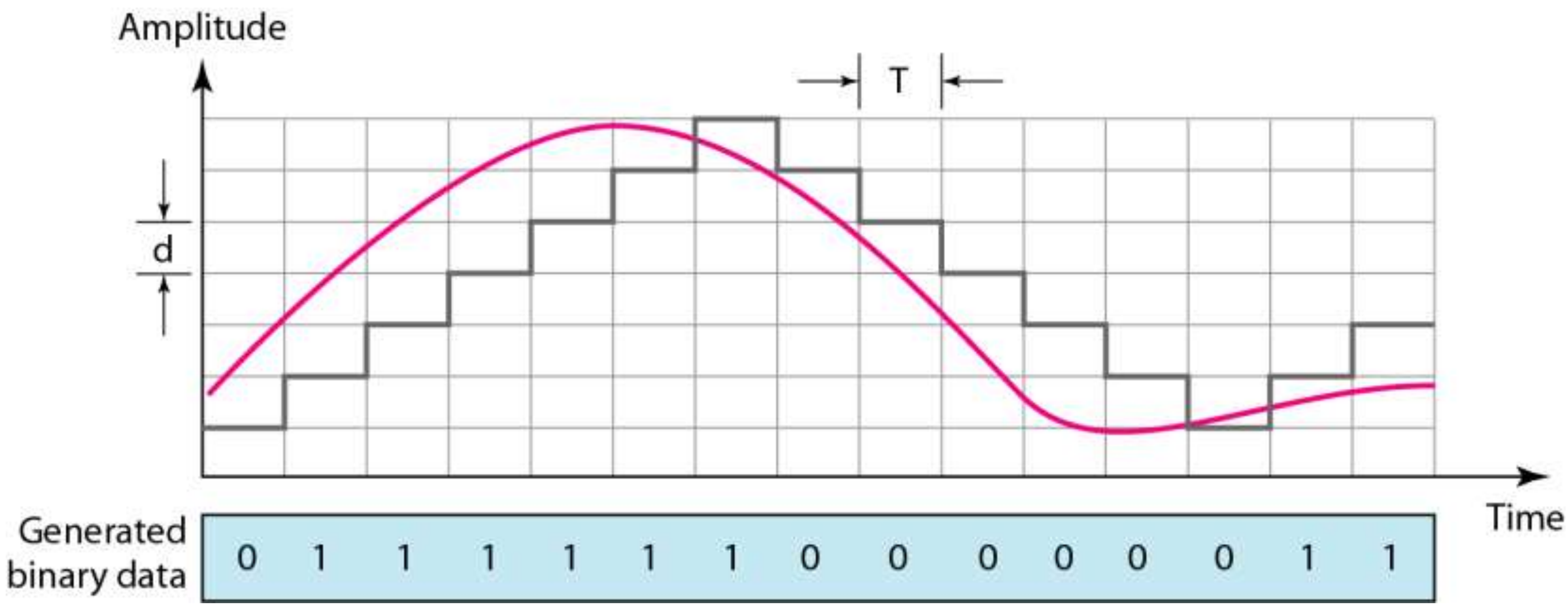


Figure 4.29 *Delta modulation components*

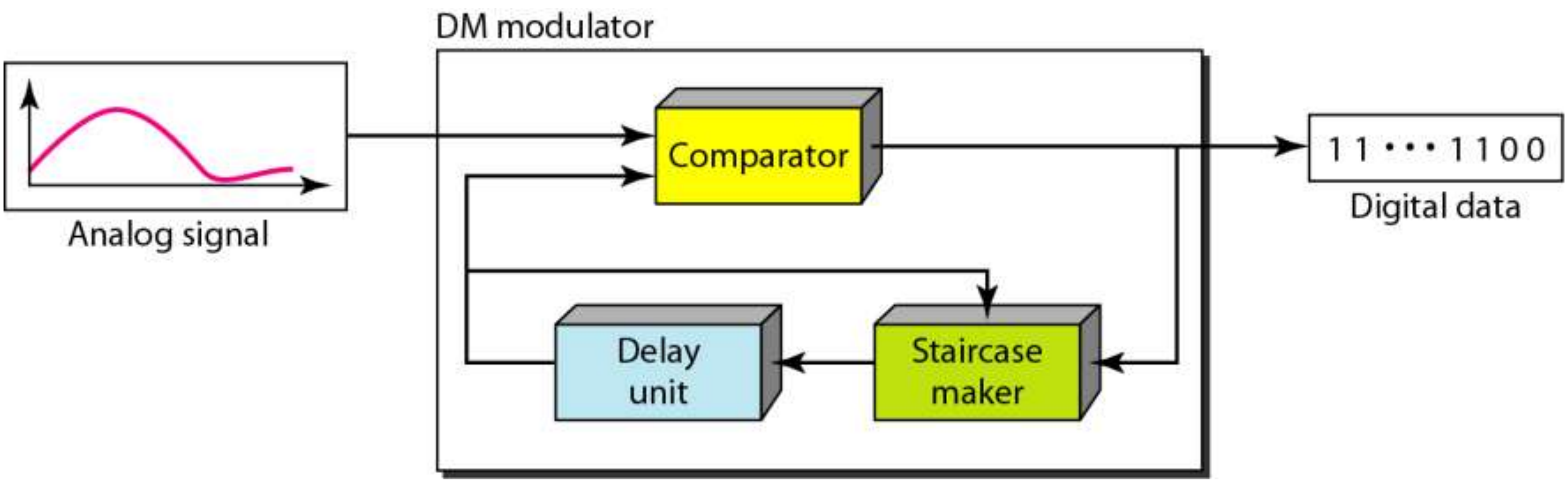
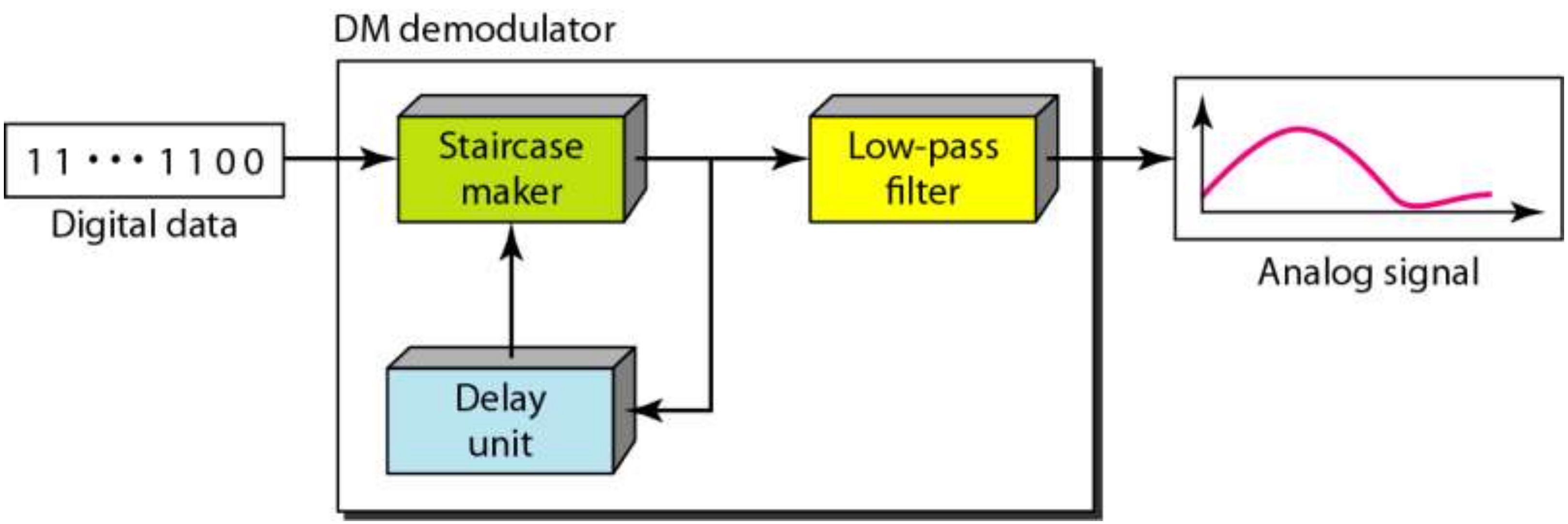


Figure 4.30 *Delta demodulation components*



Adaptive DM

- Adaptive delta modulation
  - A better performance can be achieved if the value of δ is not fixed.
  - The value of δ changes according to the amplitude of the analog signal.
- Quantization Error
  - DM is not perfect.
  - Quantization error is always introduced in the process.
  - Much less than that for PCM.

4-3 TRANSMISSION MODES

The transmission of binary data across a link can be accomplished in either parallel or serial mode. In parallel mode, multiple bits are sent with each clock tick. In serial mode, 1 bit is sent with each clock tick. While there is only one way to send parallel data, there are three subclasses of serial transmission: asynchronous, synchronous, and isochronous.

Topics discussed in this section:

Parallel Transmission
Serial Transmission

Figure 4.31 *Data transmission and modes*

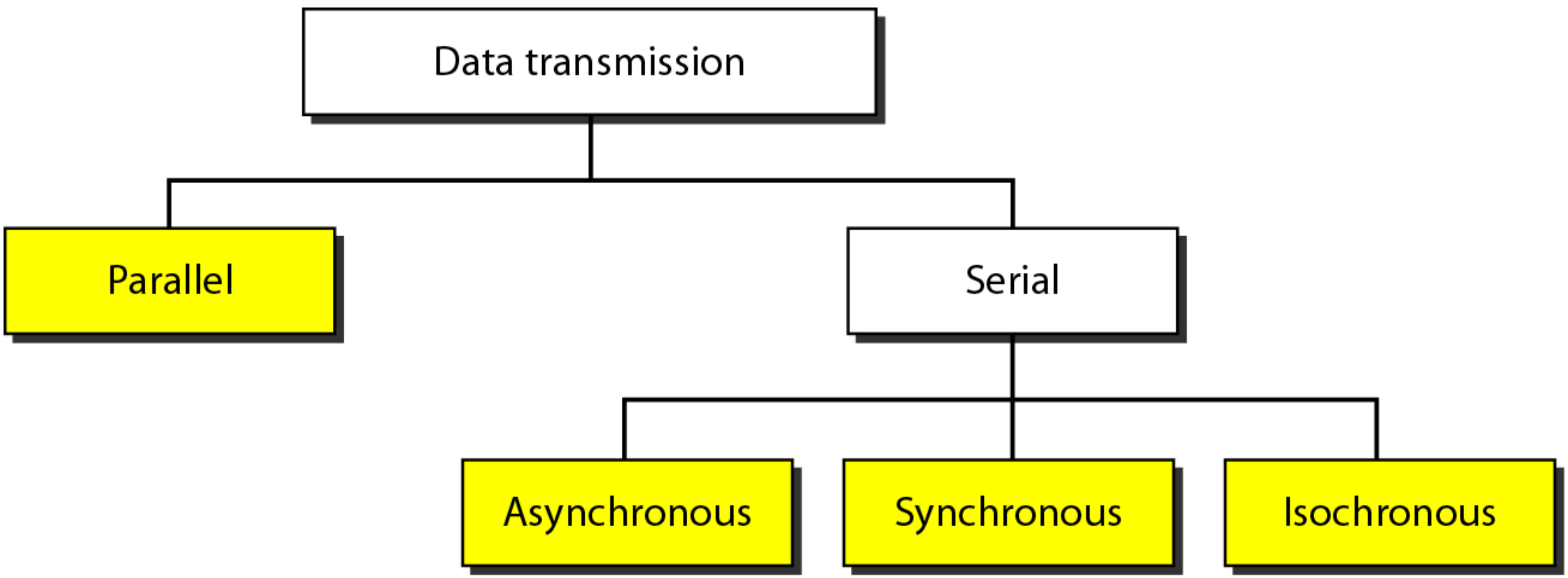


Figure 4.32 *Parallel transmission*

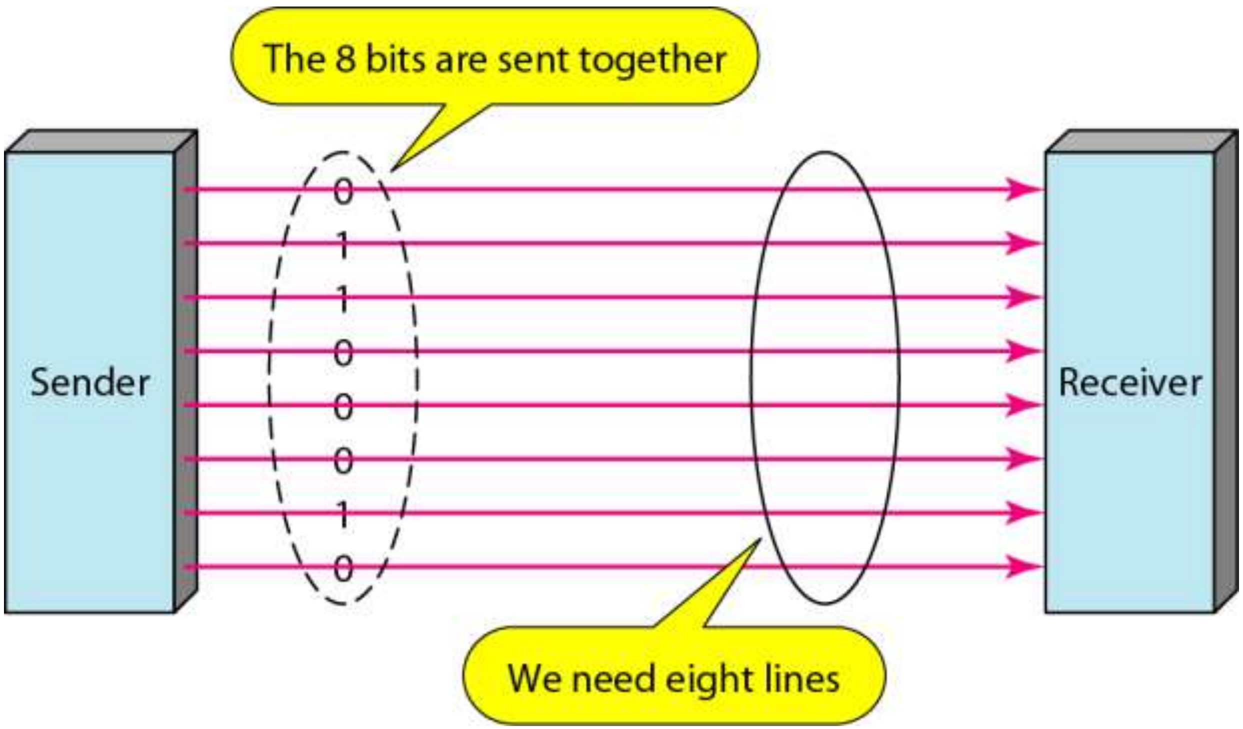
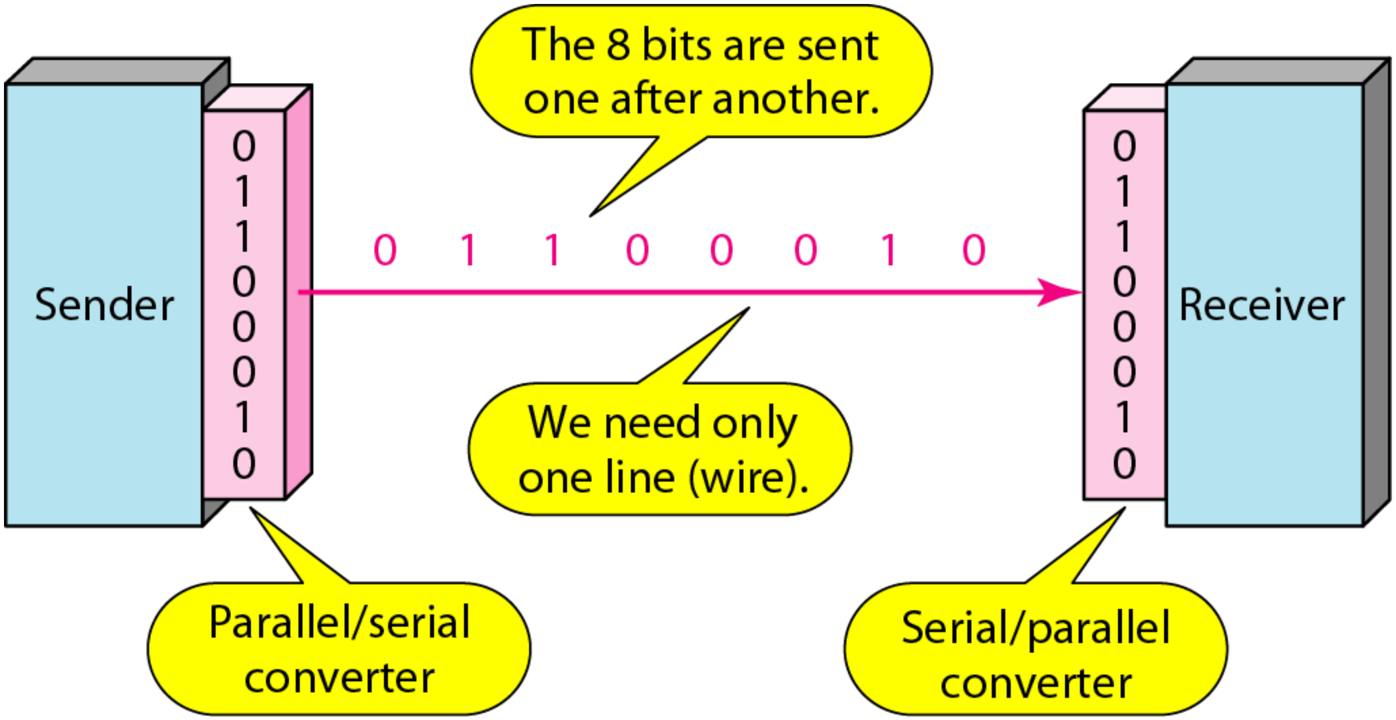


Figure 4.33 *Serial transmission*



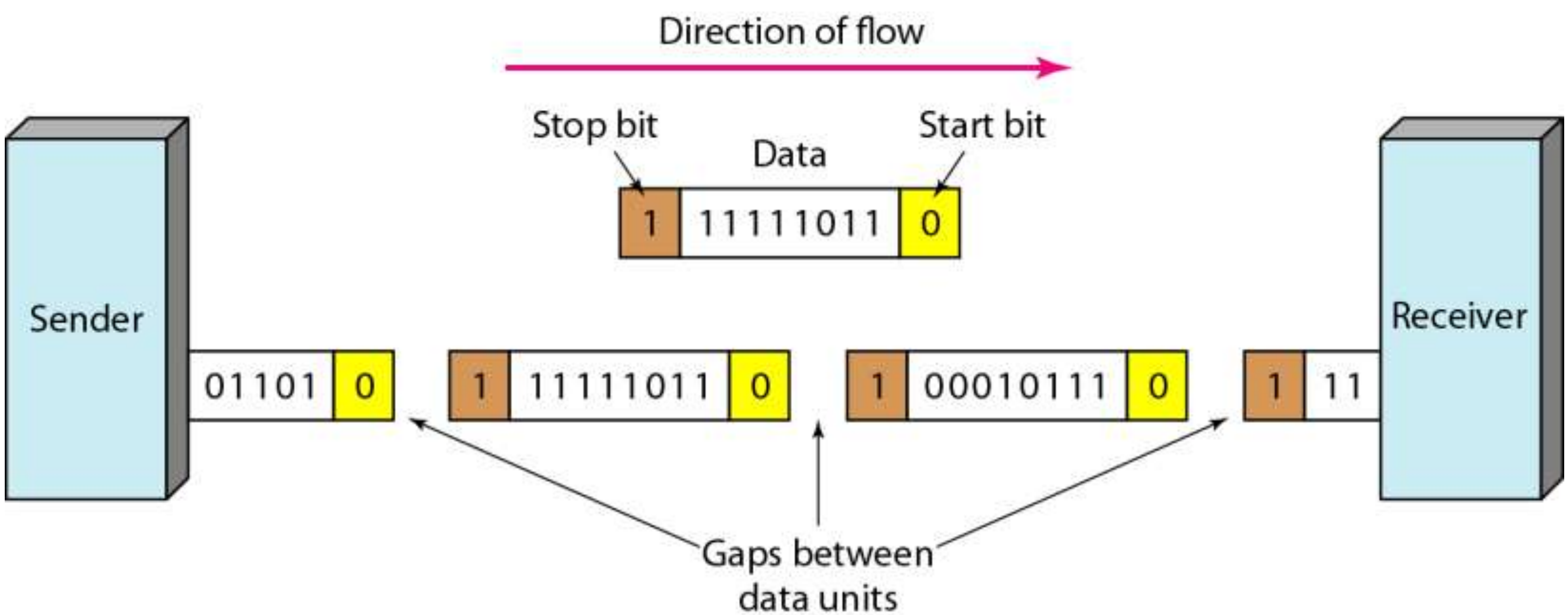
Note

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

Note

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

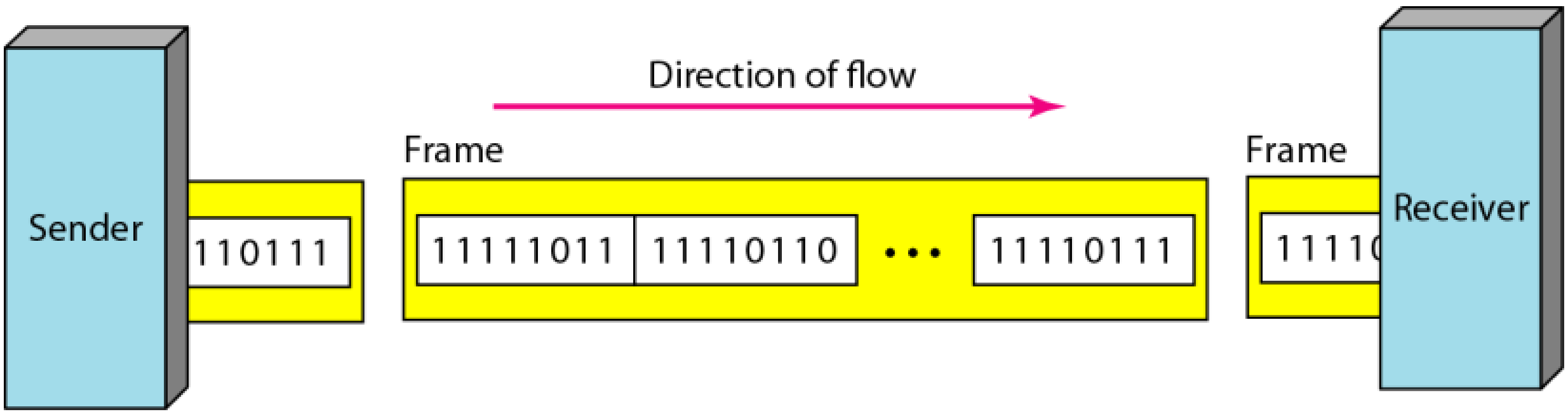
Figure 4.34 *Asynchronous transmission*



Note

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

Figure 4.35 *Synchronous transmission*



Isochronous Transmission

Note

In isochronous mode of transmission, byte level synchronization is not enough, the entire stream of bits must be synchronized.