Chapter 4

Digital Transmission

Partially Edited and Presented by

Dr. Md. Abir Hossain

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

Line coding and decoding

- **Line coding** is the process of converting digital data to digital signals.
- At the sender, digital data are encoded into a digital signal; at the receiver, the digital data are recreated by decoding the digital signal.

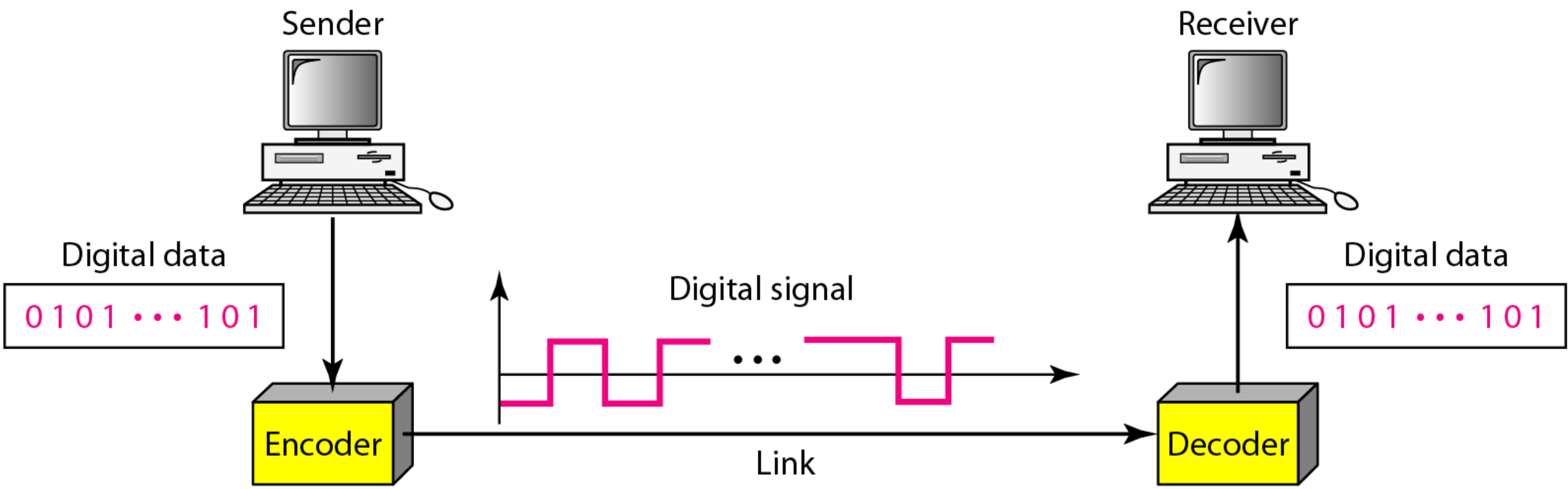
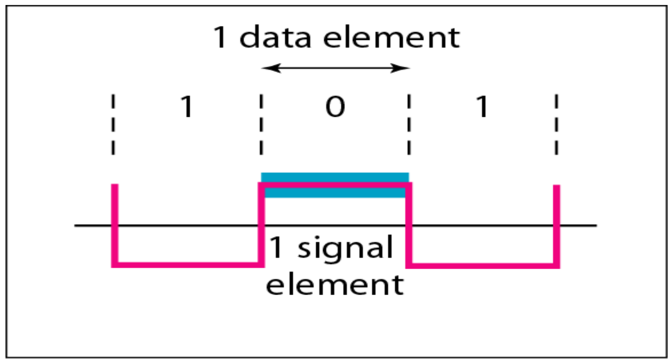


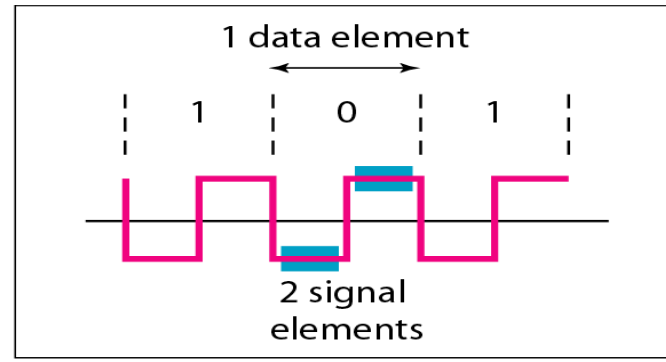
Fig 4.1: Line coding and decoding

Signal element versus data element

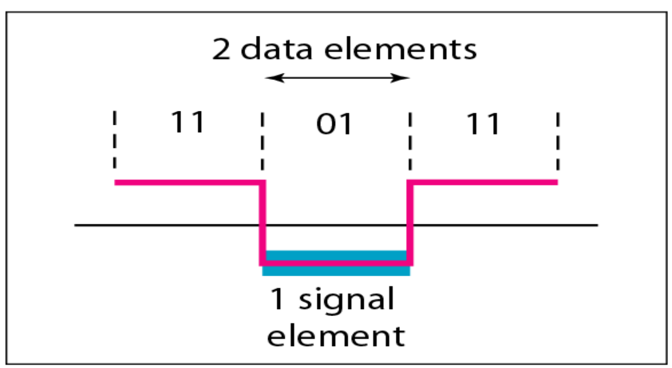
- A data element is the smallest entity that can represent a piece of information
- r is the number of data elements carried by each signal element



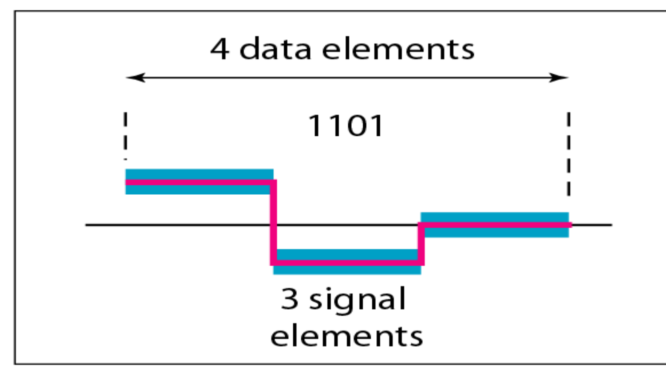
a. One data element per one signal element (r = 1)



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

- The **data rate** defines the number of data elements (bits) sent in 1s. The unit is bits per second (bps).
- The **signal rate** is the number of signal elements sent in 1s. The unit is the baud.
- There are several common terminologies used in the literature.
- The data rate is sometimes called the **bit rate;** the signal rate is sometimes called the **pulse rate,** the **modulation rate,** or the **baud rate.**
- One goal: To increase the data rate while decreasing the signal rate.
- Increasing the data rate increases the speed of transmission; decreasing the signal rate decreases the bandwidth requirement.

Relationship between data rate (N) and signal rate (S)

$$S = N/r \qquad\qquad S_{\text{average}} = c \times N \times (1/r) \text{ baud}$$

- Where, a ratio r which is the number of data elements carried by each signal element.
- where N is the data rate (bps);
- c is the case factor, which varies for each case;
- S is the number of signal elements per second

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

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

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

The minimum bandwidth can be

$$B_{min}=C \times N \times (1/r)$$

The Maximum data rate can be

$$N_{max}=r \times B \times (1/C)$$

Example 4.2

The maximum data rate of a channel 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

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

Maximum data rate if the bandwidth of the channel is given

Definitions

- In decoding a digital signal, the receiver calculates a running average of the received signal power. This average is called the ***baseline***.
- 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.

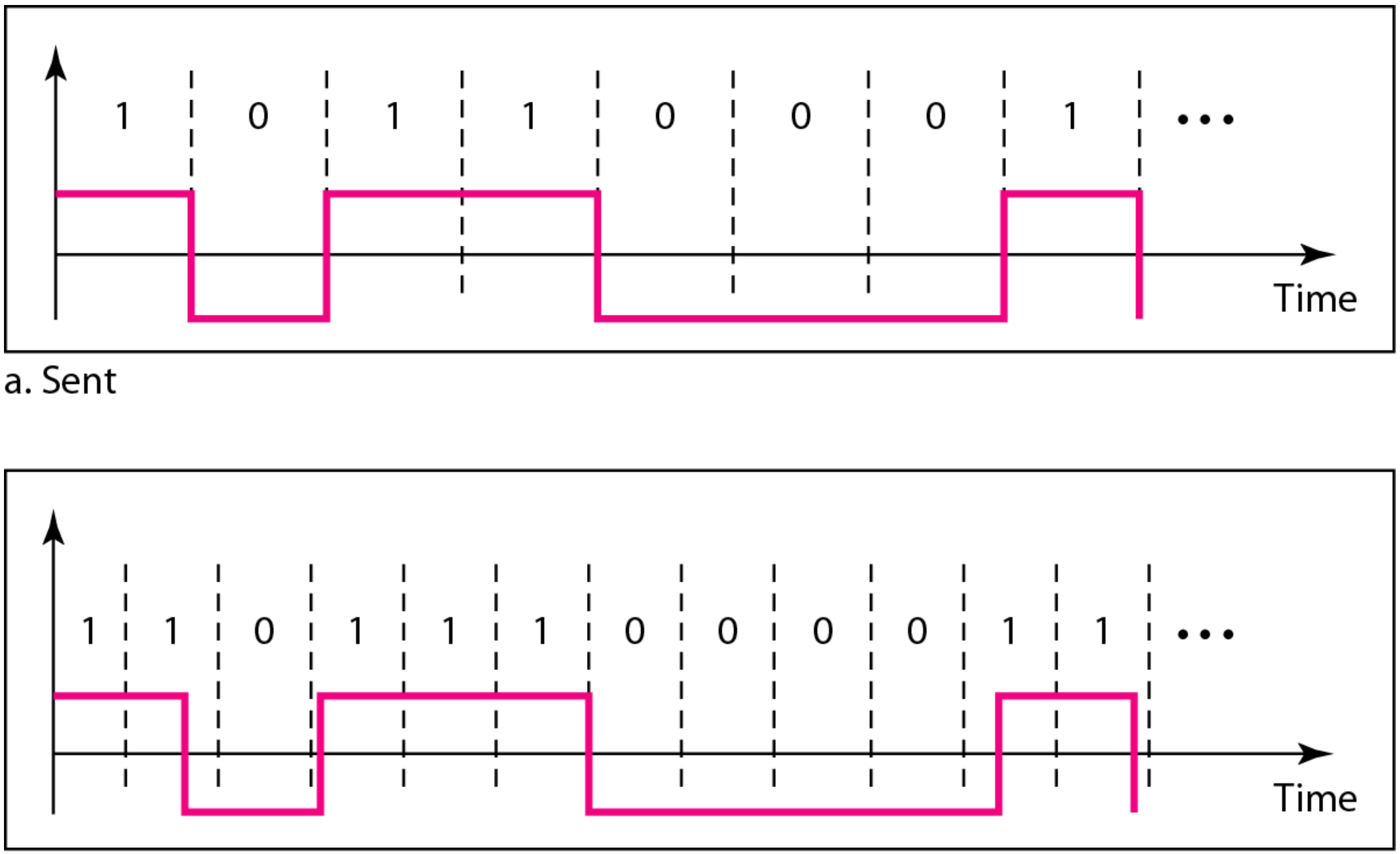
Definitions

- When the voltage level in a digital signal is constant for a while, the spectrum creates very low frequencies.
- These frequencies are around zero, called DC (direct-current) *components,* present problems for a system that cannot pass low frequencies or a system that uses electrical coupling (via a transformer).
- DC component means 0/1 parity that can cause base-line wondering.
- For example, a telephone line cannot pass frequencies below 200 Hz. Also a long-distance link may use one or more transformers to isolate different parts of the line electrically. For these systems, we need a scheme with no **DC component.**

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.
- Figure 4.3 (next slide) shows a situation in which the receiver has a shorter bit duration.
- The sender sends 10110001, while the receiver receives 110111000011.
- A **self-synchronizing** 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.
- If the receiver's clock is out of synchronization, these points can reset the clock.

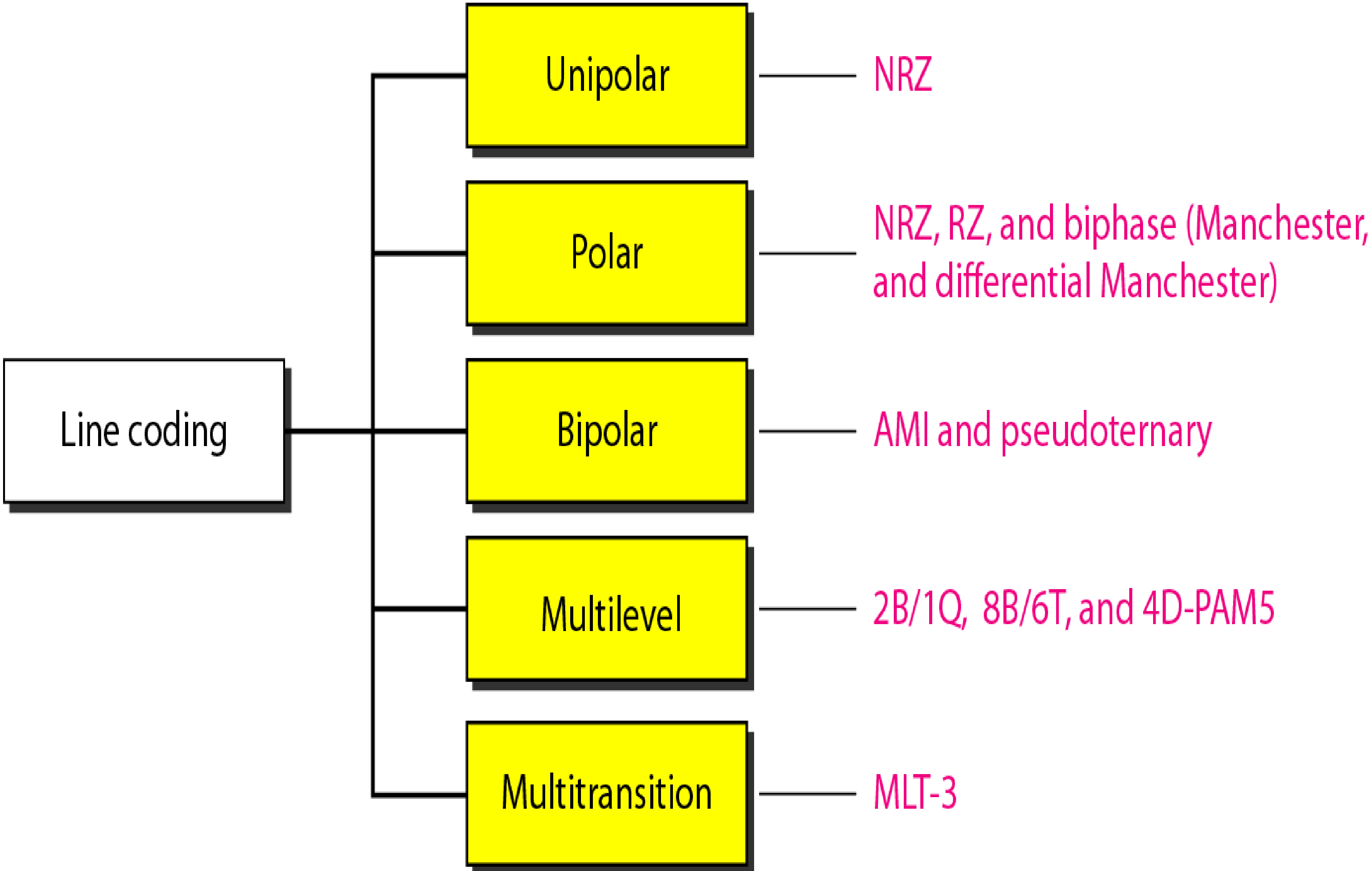
Figure 4.3 *Effect of lack of synchronization*



a. Sent

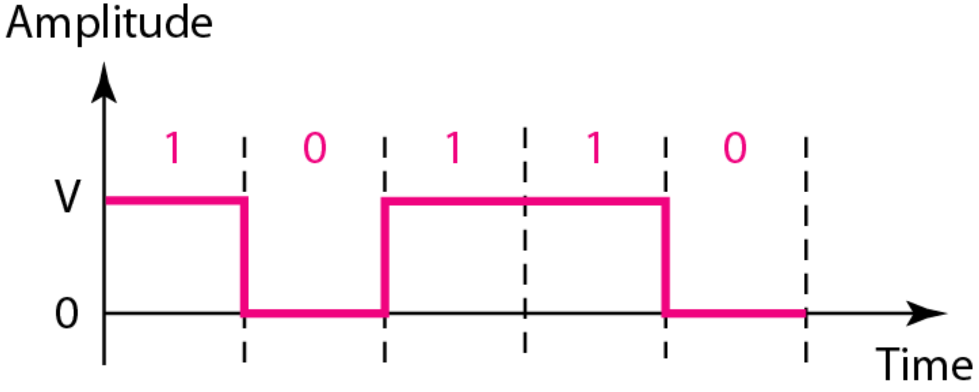
b. Received

Line coding schemes



Unipolar NRZ scheme

- In a unipolar scheme, all the signal levels are on one side of the time axis, either above or below.
- In Non-Return-to-Zero, positive voltage defines bit 1 and the zero voltage defines bit 0.
- It is called NRZ because the signal does not return to zero at the middle of the bit.
- The scheme is costly.
- The normalized power (the power needed to send 1 bit per unit line resistance) is double that for polar NRZ.
- Disadvantage: DC Component and Synchronization.

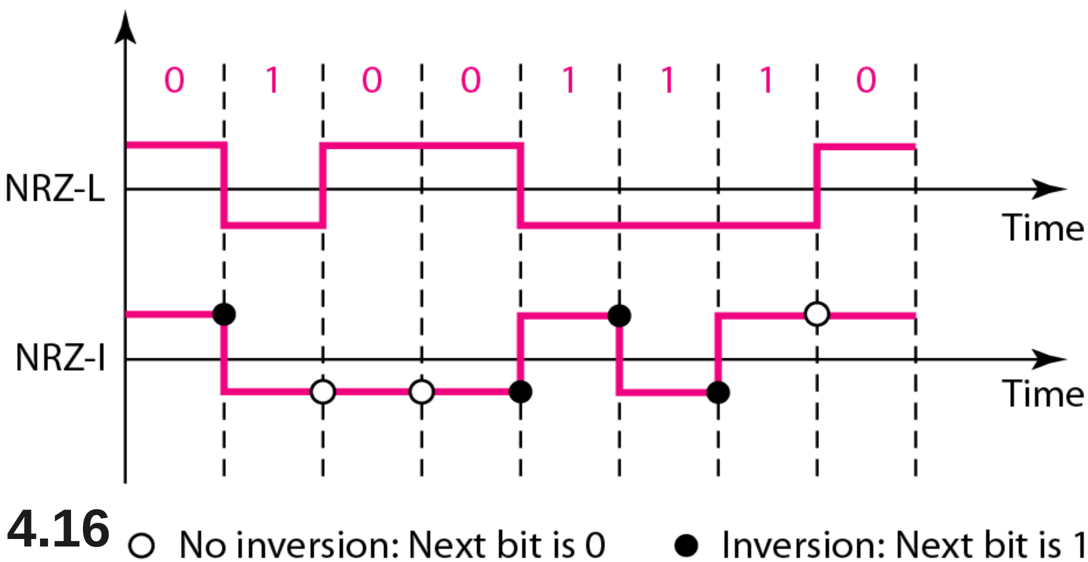


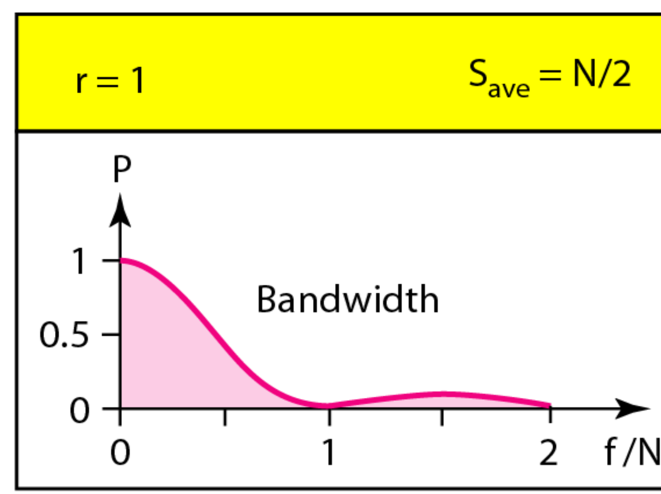
$$\frac{1}{2}V^2 + \frac{1}{2}(0)^2 = \frac{1}{2}V^2$$

Normalized power

Polar NRZ-L and NRZ-I scheme

- In polar schemes, the voltages are on both sides of the time axis.
- Non-Return-to-Zero (NRZ) with L (Level) and I (Invert).
- 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.
- If there is a long sequence of 0s or 1s in NRZ-L, the average signal power becomes skewed. In NRZ-I this problem occurs only for a long sequence of 0s.
- The synchronization problem. Another problem with NRZ-L occurs when there is a sudden change of polarity in the system. NRZ-L and NRZ-I both have an average signal rate of N/2 Bd. NRZ-L and NRZ-I both have a DC component problem.





Example 4.4

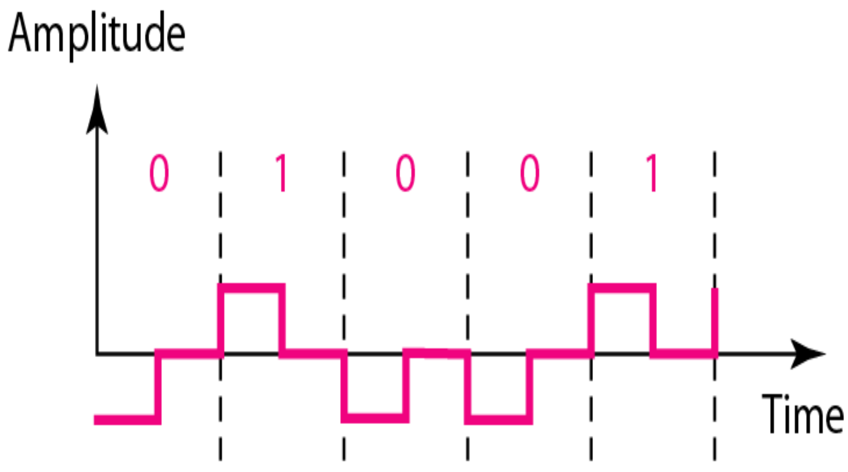
A system is using NRZ-I to transfer 1Mbps data. What are the average signal rate and minimum bandwidth?

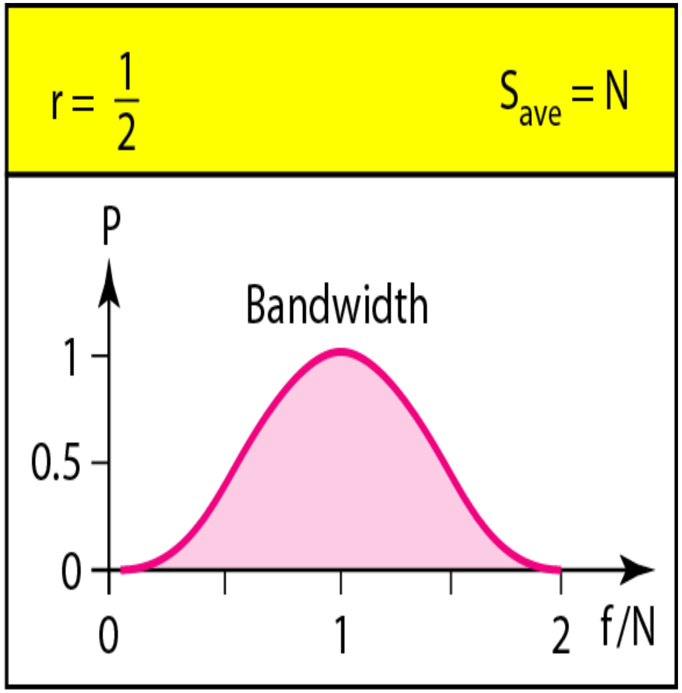
Solution

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

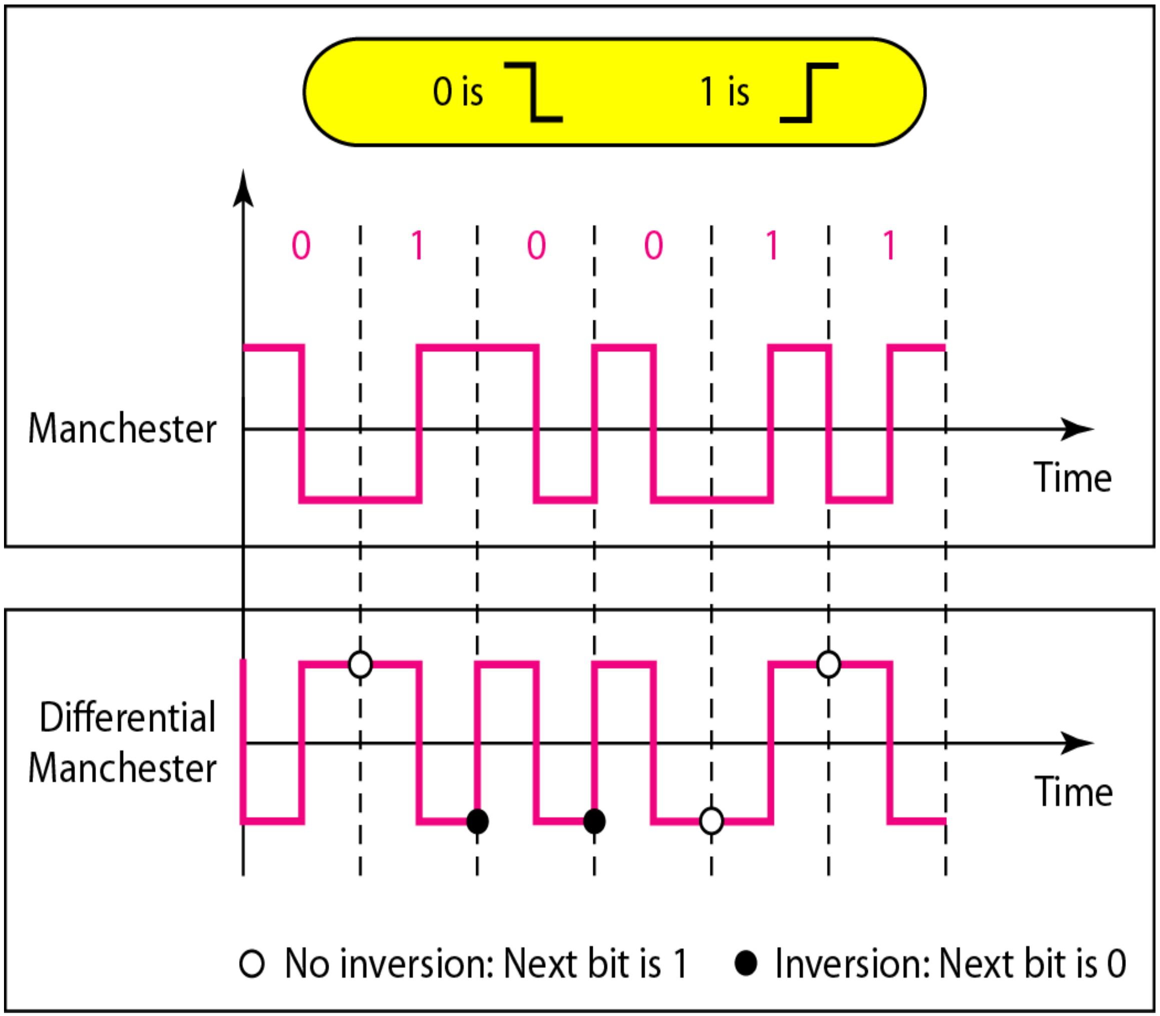
Polar Return to Zero scheme

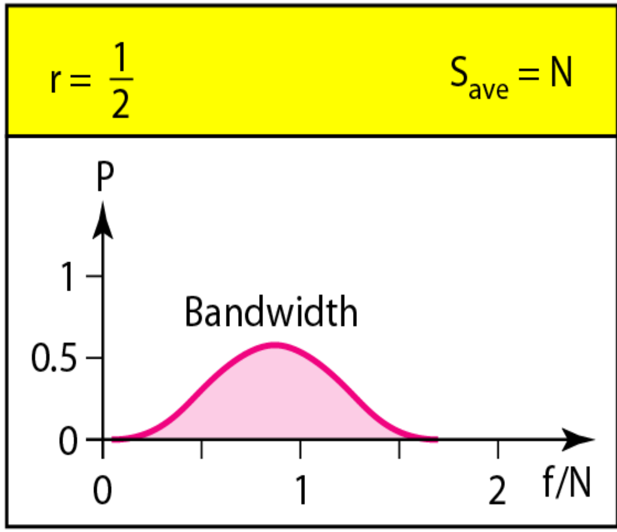
- ***Return-to-Zero (RZ)*** uses three values: positive, negative, and zero.
- Signal changes not between bits but during the bit.
- Occupy greater bandwidth as needs change during the bits.
- No DC component problem.
- Another problem is the complexity due to 3 signals.
- Not in use.





Polar Bi-phase (Manchester and differential Manchester schemes)





Note

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

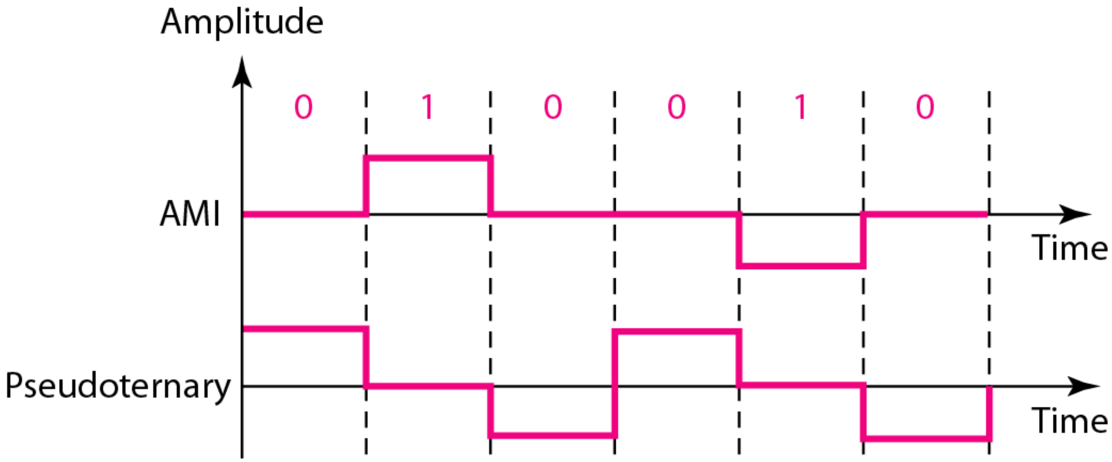
The minimum bandwidth of Manchester and differential Manchester is 2 times that of NRZ.

Note

In bipolar encoding, we use three levels: positive, zero, and negative.

Bipolar schemes: AMI and pseudoternary

- In bipolar encoding, we use three levels: positive, zero, and negative.
- **Alternate Mark Inversion (AMI)** and **Pseudoternary.**
- Mark means 1. So AMI means alternate 1 inversion.
- A neutral zero voltage represents binary 0. Binary 1s are represented by alternating positive and negative voltages.
- A variation of AMI encoding is called pseudoternary in which the 1 bit is encoded as a zero voltage and the 0 bit is encoded as alternating positive and negative voltages.
- Same signal rate as NRZ, but there is no DC component.



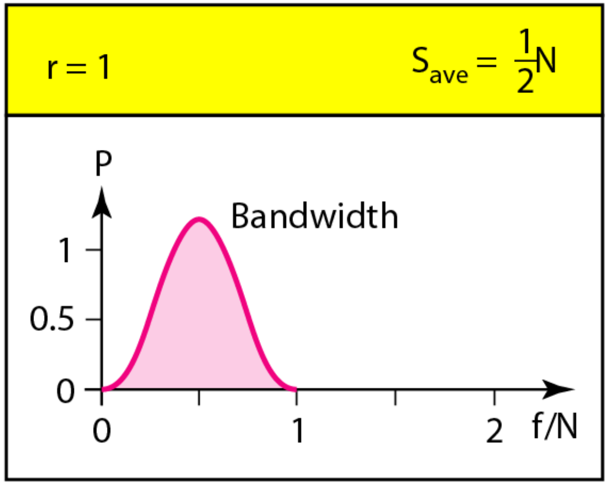
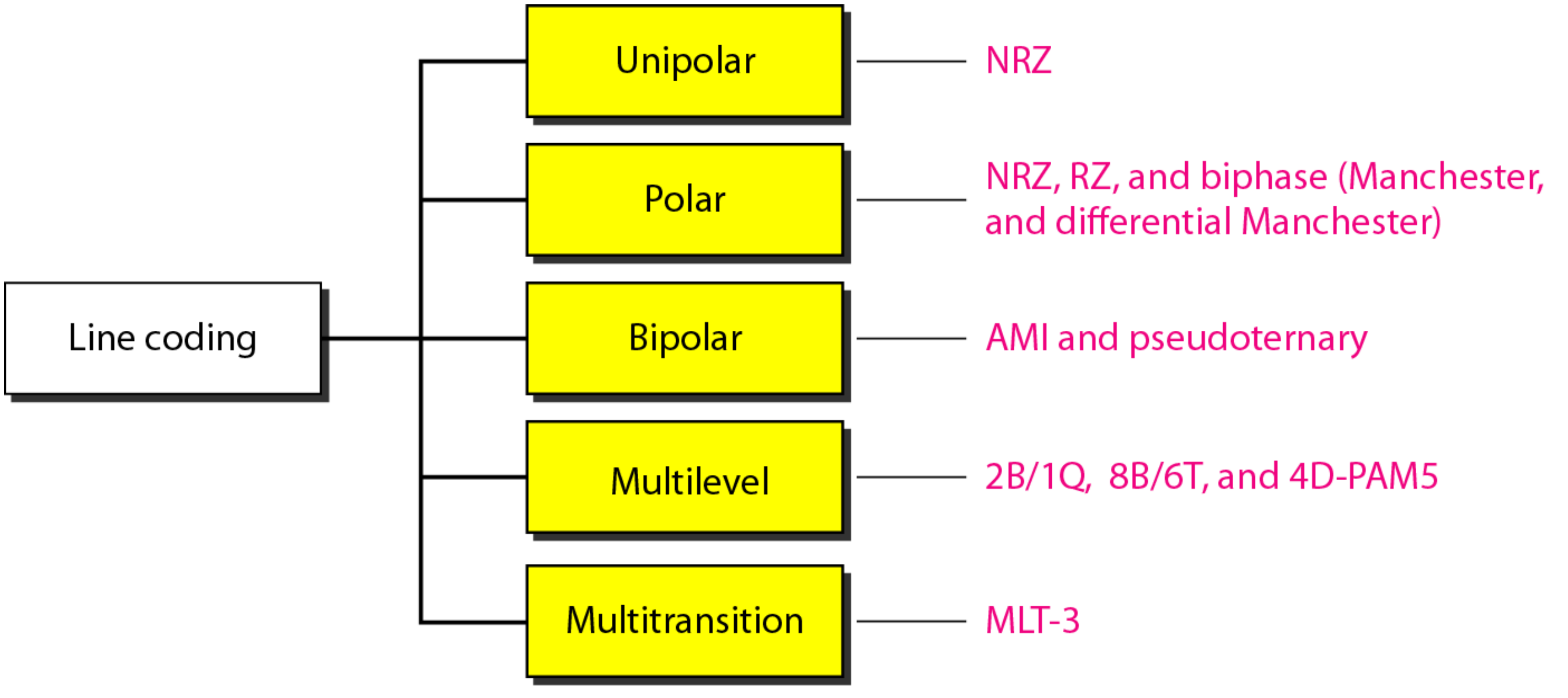


Figure 4.4 *Line coding schemes*

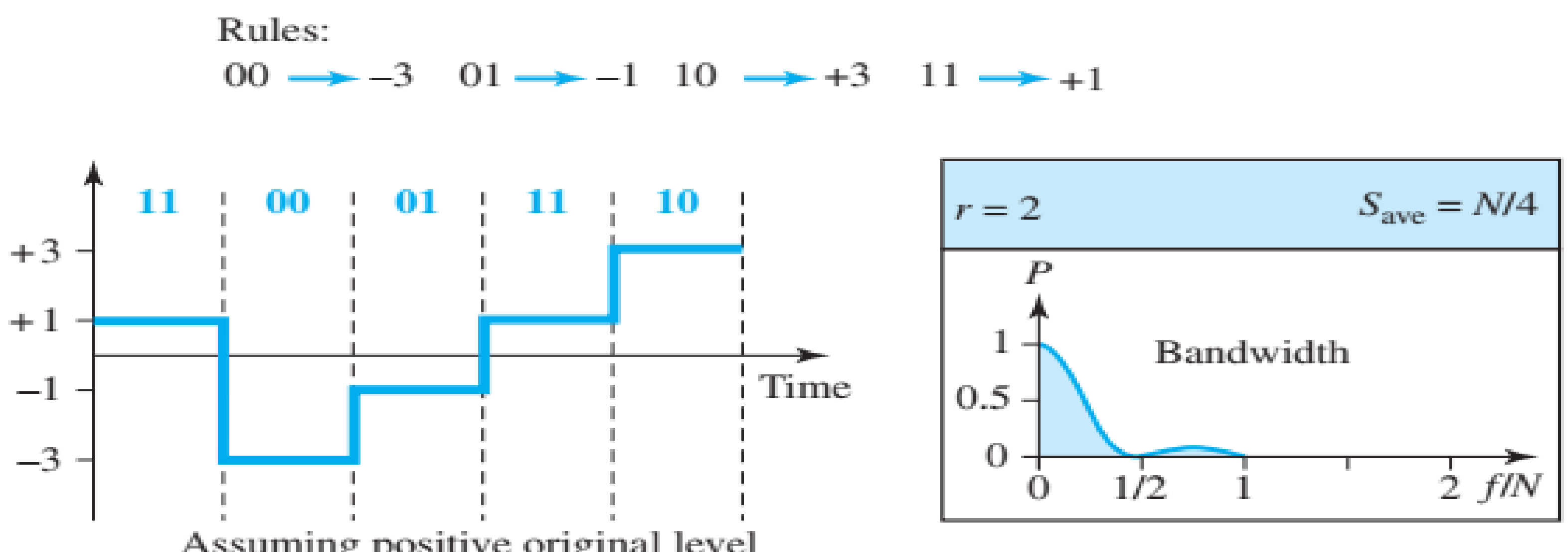


Note

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

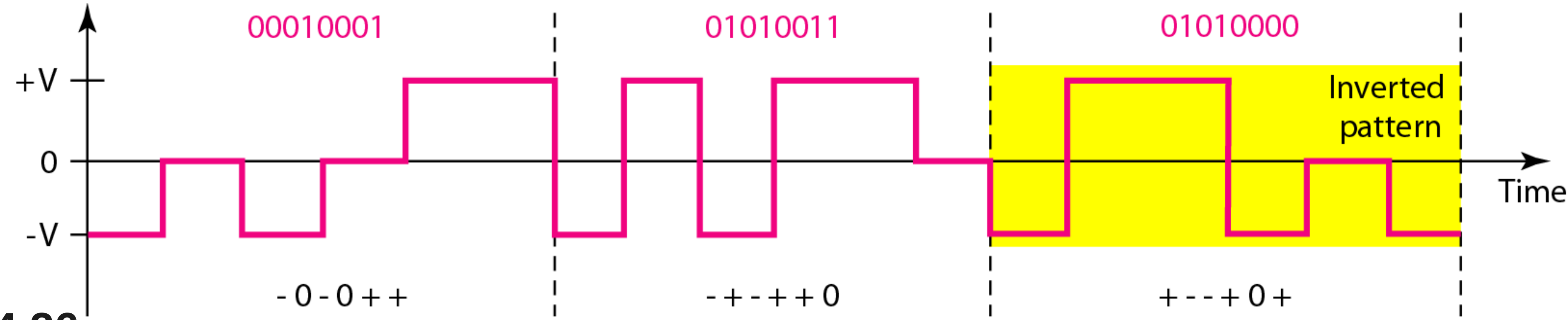
Multilevel: 2B1Q scheme

- The first *mBnL* scheme we discuss, **two binary, one quaternary (2B1Q),** uses data patterns of size 2 and encodes the 2-bit patterns as one signal element belonging to a four-level signal.
- In this type of encoding $m = 2$, $n = 1$, and $L = 4$ (quaternary).
- 2 times faster than by using NRZ-L
- There are no redundant signal patterns in this scheme because $2^2 = 4^1$.
- Used in DSL (Digital Subscriber Line) technology to provide a high-speed connection to the Internet by using subscriber telephone lines



Multilevel: 8B6T scheme

- The **eight binary, six ternary (8B6T)** is used with 100BASE-4T cable.
- Signal has three levels (ternary) $2^8 = 256$ different data patterns and $3^6 = 729$ different signal patterns.
- There are 729 - 256 = 473 redundant signal elements that provide synchronization, error detection and provide DC balance.
- The first 8-bit pattern 00010001 is encoded as the signal pattern - 0 - 0 + + with weight 0; the second 8-bit pattern 01010011 is encoded as - + - + + 0 with weight +1. The third 8-bit pattern 01010000 should be encoded as + - - + 0 + with weight +1.
- The receiver can easily recognize that this is an inverted pattern because the weight is -1.



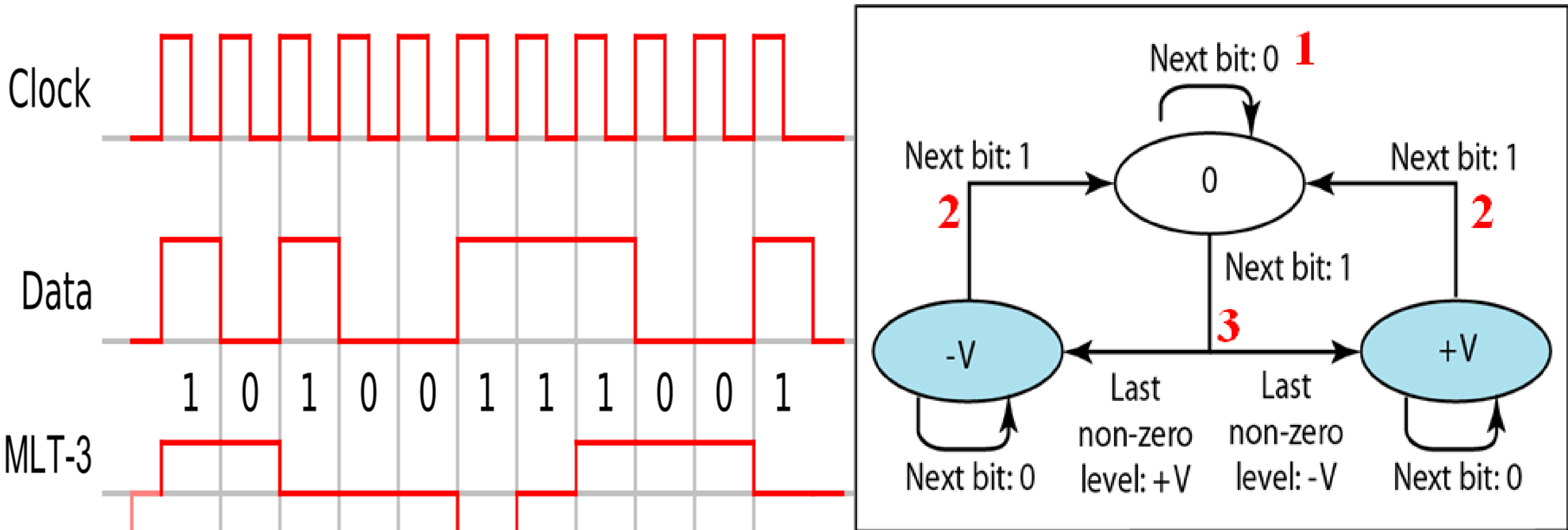
Multilevel: 8B6T codes

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

Multitransition: MLT-3

- The **multiline transmission, three-level (MLT-3) scheme uses** three levels (+*V*, 0, and -*V*) and three transition rules to move between the levels.
  1. If the next bit is 0, there is no transition.
  2. If the next bit is 1 and the current level is not 0, the next level is 0.
  3. If the next bit is 1 and the current level is 0, the next level is the opposite of the last nonzero level.
- The three voltage levels (-*V*, 0, and +*V*) are shown by three states (ovals).
- It turns out that the shape of the signal in this scheme helps to reduce the required bandwidth.
- MLT-3 a suitable choice when we need to send 100 Mbps on a copper wire that cannot support more than 32 MHz.
- 1 = level change. 0 = no change.

Multitransition: MLT-3



c. Transition states

Table 4.1 *Summary of line coding schemes*

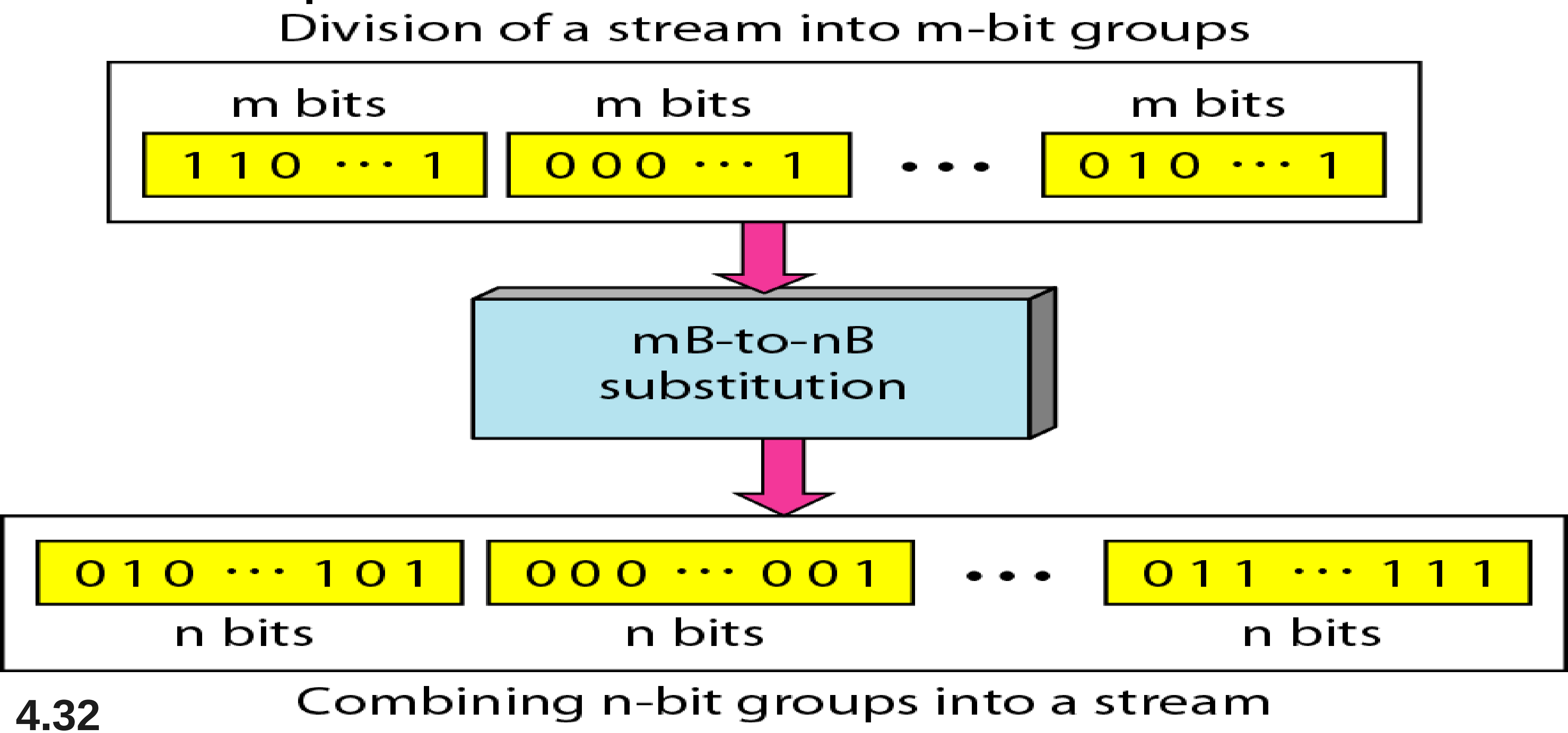
Category	*Scheme*	*Bandwidth (average)*	*Characteristics*
Unipolar	NRZ	$B = N/2$	Costly, no self-synchronization if long 0s or 1s, DC
Unipolar	NRZ-L	$B = N/2$	No self-synchronization if long 0s or 1s, DC
	NRZ-I	$B = N/2$	No self-synchronization for long 0s, DC
	Biphase	$B = N$	Self-synchronization, no 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
Multiline	MLT-3	$B = N/3$	No self-synchronization for long 0s

Note

Block coding is normally referred to as *m*B/*n*B coding;
it replaces each *m*-bit group with an *n*-bit group.

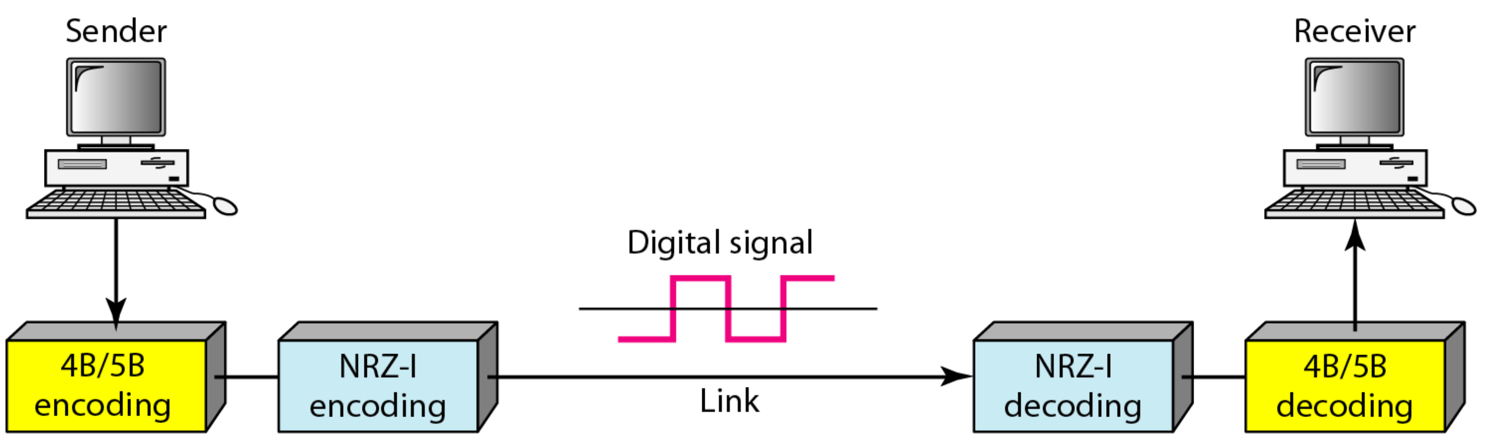
Block coding

- Block coding changes a block of m bits into a block of n bits, where n is larger than m.
- Block coding is referred to as an mB/nB encoding technique.



4B/5B Block coding with NRZ-I

- The four binary/five binary (4B/5B) coding scheme was designed to used in combination with NRZ-I.
- NRZ-I has a good signal rate but has synchronization problem(long 0’s stream).
- One solution is to change the bit stream, prior to encoding with NRZ-I so that it does not have a long stream of 0s.
- The 4B/5B scheme does not allowed more that three consecutive 0s.
- At the receiver, the NRZ-I encoded digital signal is first decoded into a stream of bits and then decoded to remove the redundancy.

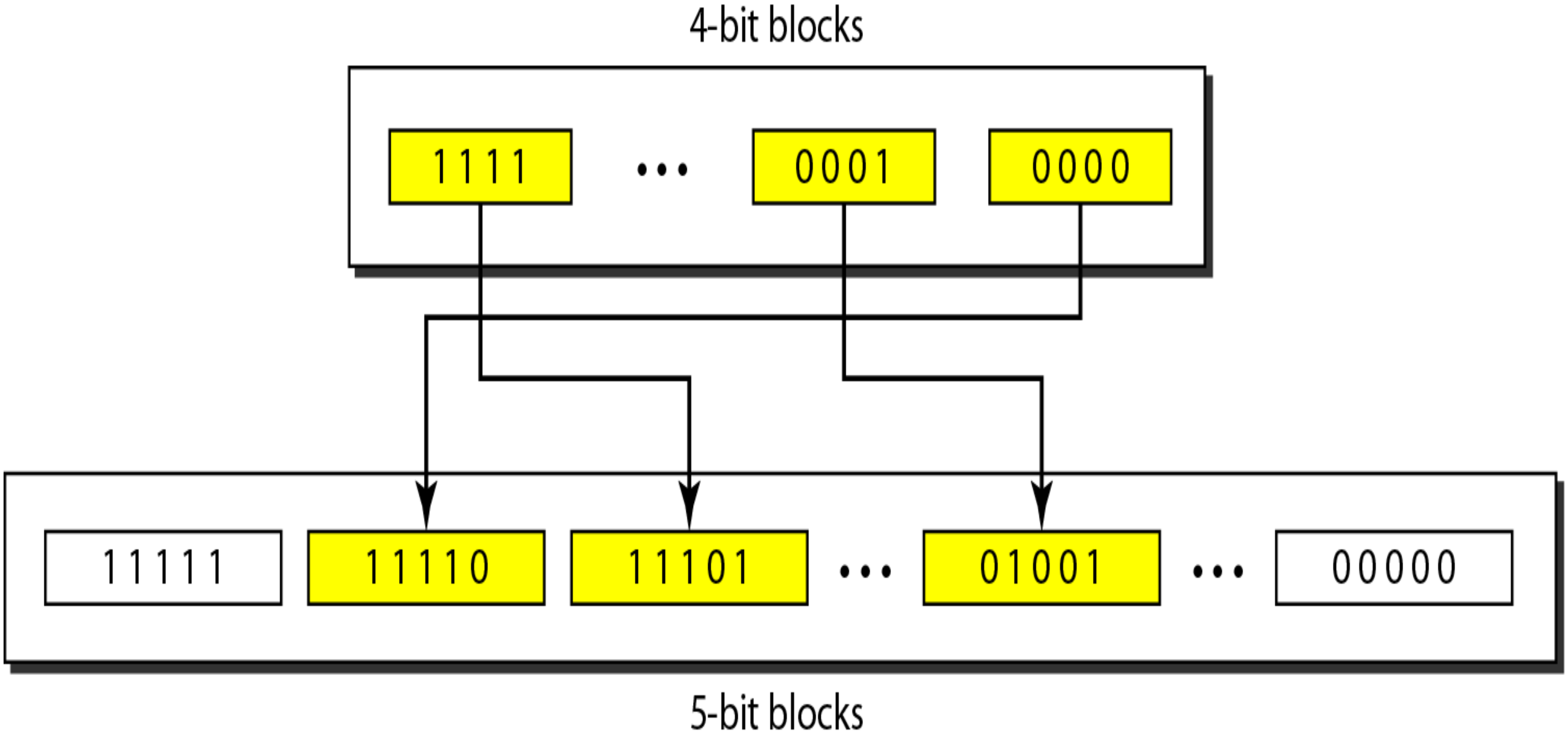


4B/5B mapping codes

- In 4B/5B, the 5-bit output that replaces with the 4-bit input has no more than one leading zero (left bit) and no more than two trailing zeros (right bits).
- So that, if we combine two encoded sequence that would not produce more that three consecutive 0s.
- Ex. 10100 01010

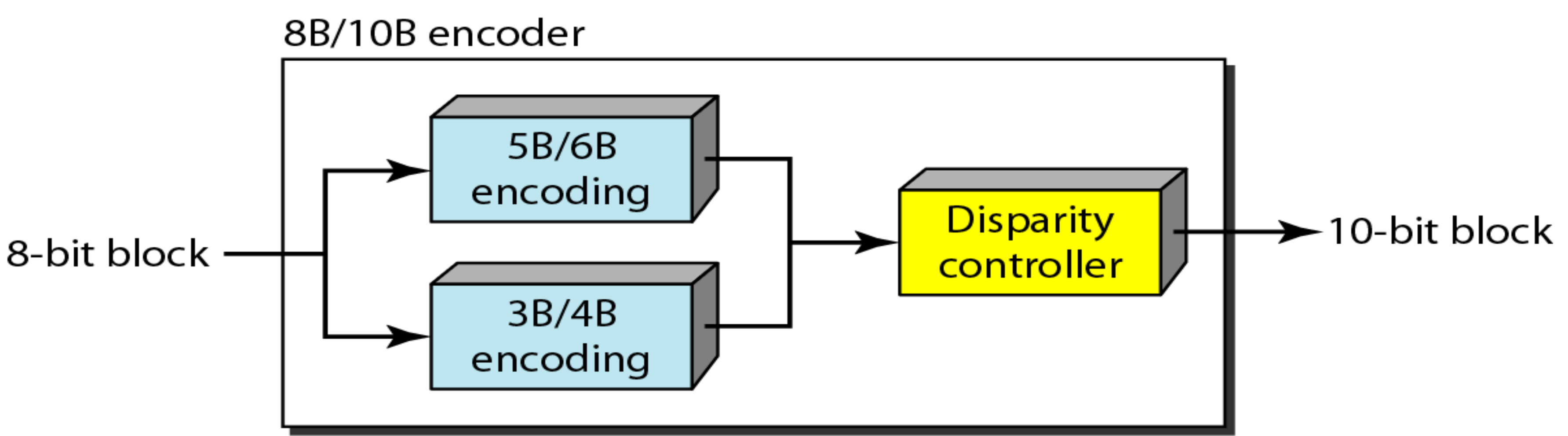
Data Sequence	*Encoded Sequence*	*Control Sequence*	*Encoded Sequence*
0000	11110	Q (Quiet)	00000
0001	01001	I (Idle)	11111
0010	10100	H (Halt)	00100
0011	10101	J (Start delimiter)	11000
0100	01010	K (Start delimiter)	10001
0101	01011	T (End delimiter)	01101
0110	01110	S (Set)	11001
0111	01111	R (Reset)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

Substitution in 4B/5B block coding



8B/10B block encoding

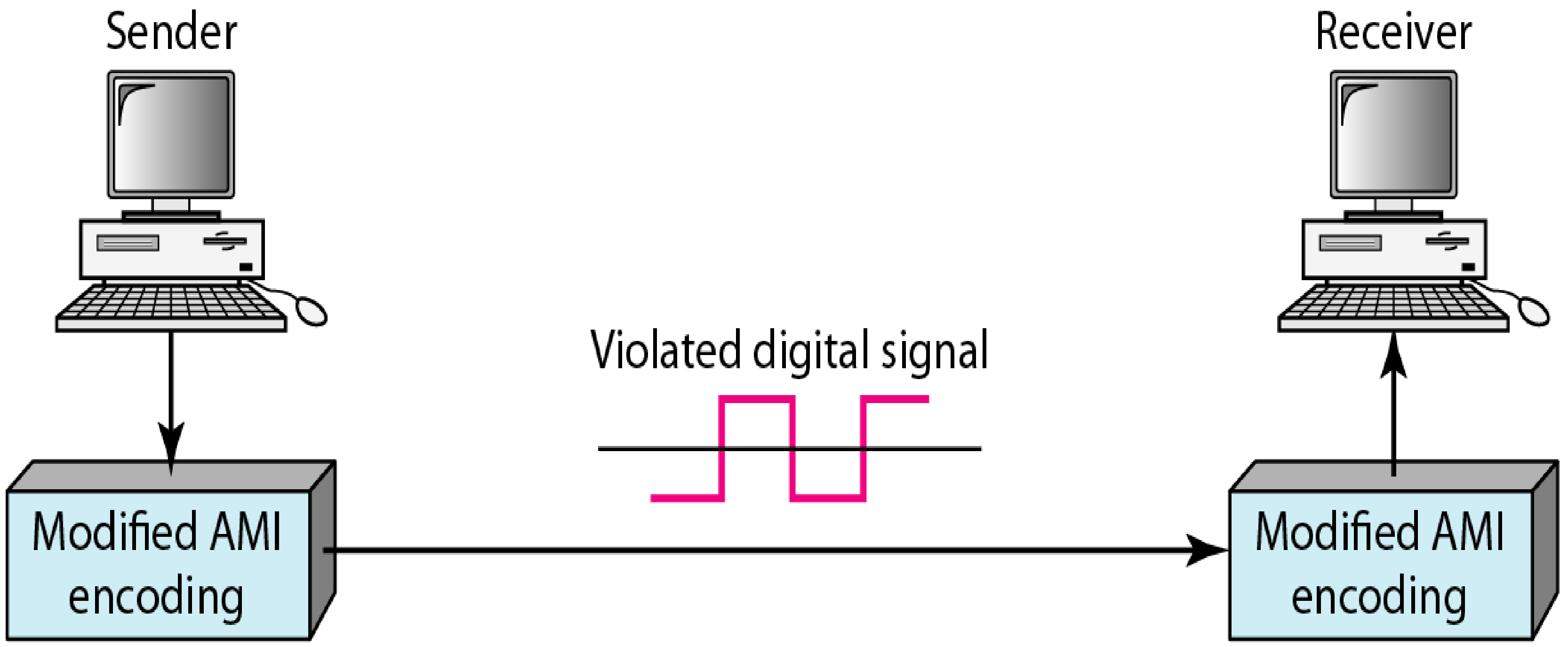
- The eight binary/ten binary (8B/10B) encoding is similar to 4B/5B encoding except that a group of 8 bits of data is now substituted by a 10-bit code.
- The 8B/10B block coding is actually a combination of 5B/6B and 3B/4B encoding.
- The five most significant bits of a 10-bit block are fed into the 5B/6B encoder and
- The three least significant bits are fed into a 3B/4B encoder.
- To prevent a long run of consecutive 0s or 1s, the code uses a disparity controller which keeps track of excess 0s over 1s (or 1s over 0s).
- The coding has $2^{10} - 2^{8}$= 768 redundant groups that can be used for disparity checking and error detection.
- The technique is superior to 4B/5B because of better built-in error-checking capability and better synchronization.



Scrambling

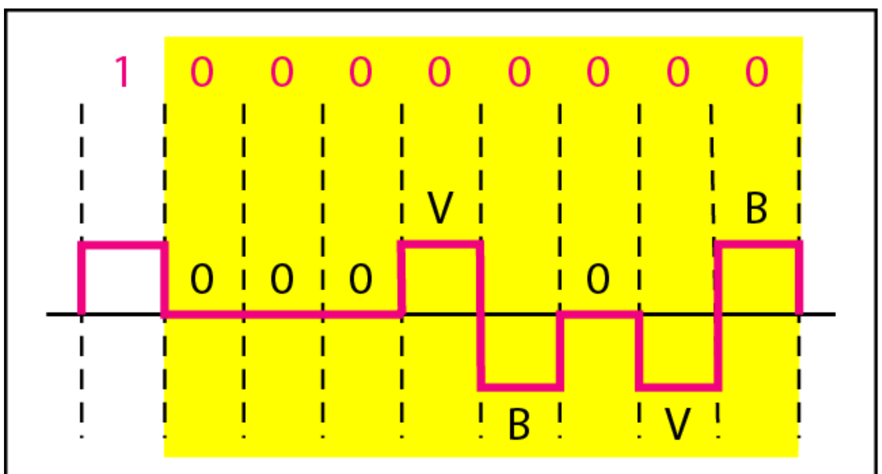
- We are looking for a technique that does not increase the number of bits and does provide synchronization.
- We are looking for a solution that substitutes long zero-level pulses with a combination of other levels to provide synchronization.
- One solution is called **scrambling.**
- It is done at the same time when encoding.
- Two common scrambling techniques are B8ZS and HDB3.
- **Bipolar with 8-zero substitution (B8ZS)**: In this technique, eight consecutive zero-level voltages are replaced by the sequence **000VB0VB.**

AMI used with scrambling

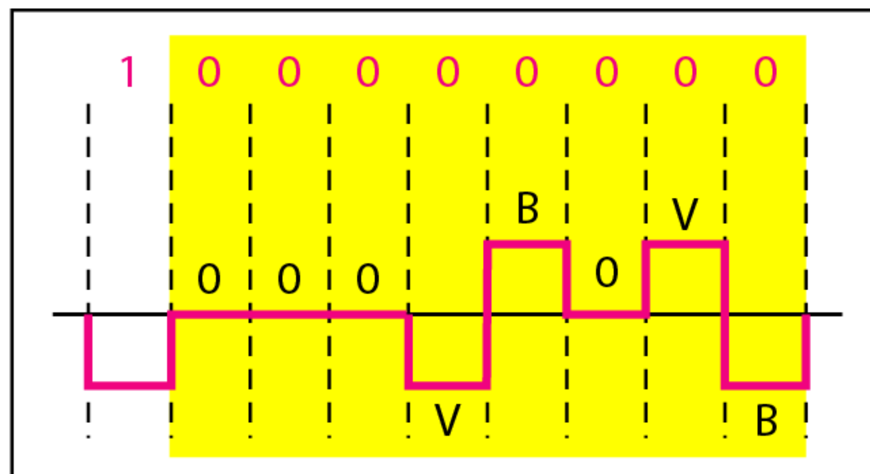


Bipolar with 8-zero substitution (B8ZS)

- Bipolar with 8-zero substitution (B8ZS) is commonly used in North America
- In this technique, 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 (opposite polarity from the previous).
- The **B** in the sequence denotes bipolar, which means a nonzero level voltage in accordance with the AMI rule.
- **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.**



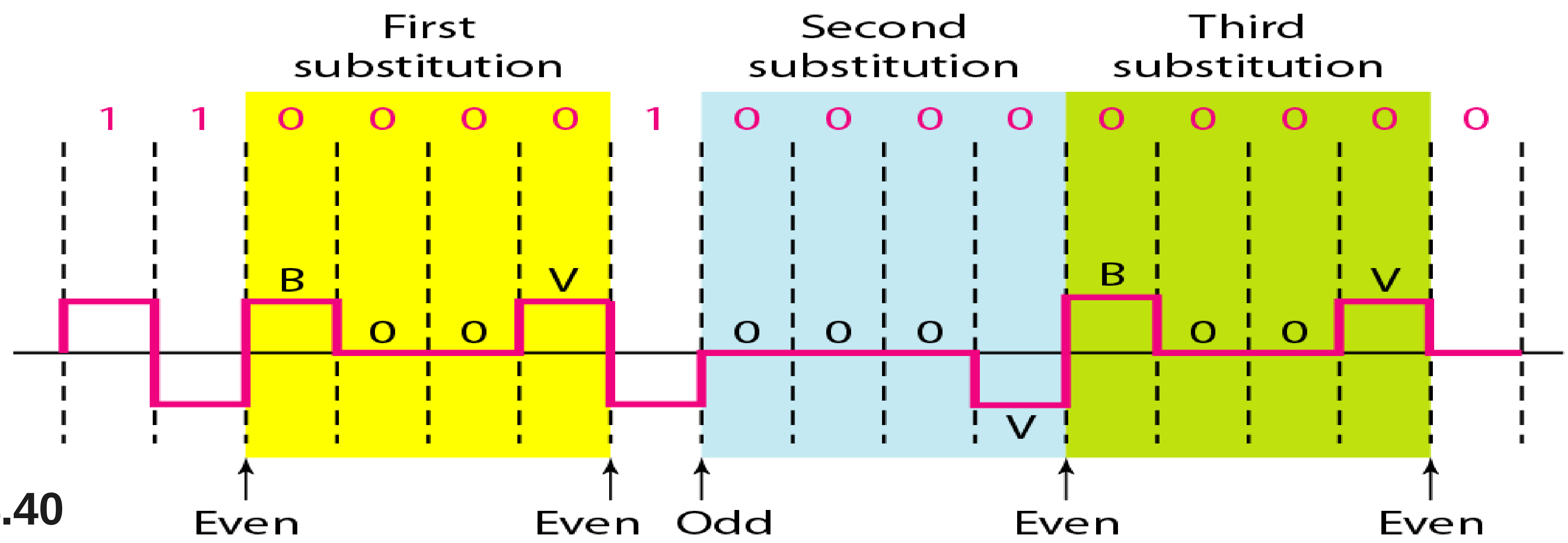
a. Previous level is positive.



b. Previous level is negative.

High-density bipolar 3-zero (HDB3)

- In HDB3, four consecutive zero-level voltages are replaced with a sequence of 000V or B00V depending on the number of nonzero pulses after the last substitution.
- The reason for two different substitutions is to maintain the even number of nonzero pulses after each substitution.
- It follows two rules
- 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.



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)

Pulse Code Modulation (PCM)

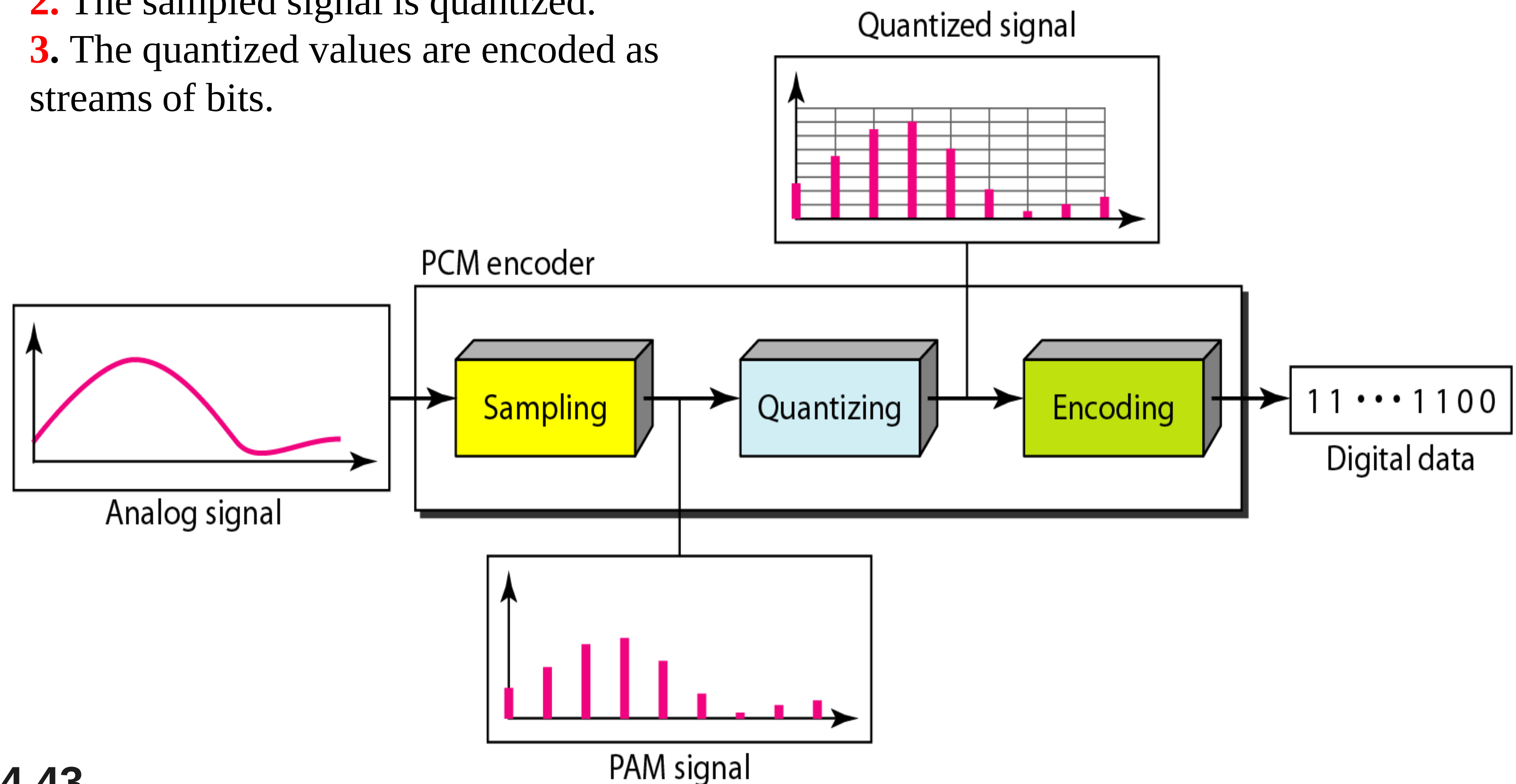
- The most common technique to change an analog signal to digital data (digitization) is called **pulse code modulation (PCM).**
- A PCM encoder has three processes
  - The analog signal is sampled.
  - The sampled signal is quantized.
  - The quantized values are encoded as streams of bits.

Components of PCM encoder

Pulse Code Modulation.

Analog to digital conversion. Digitization.

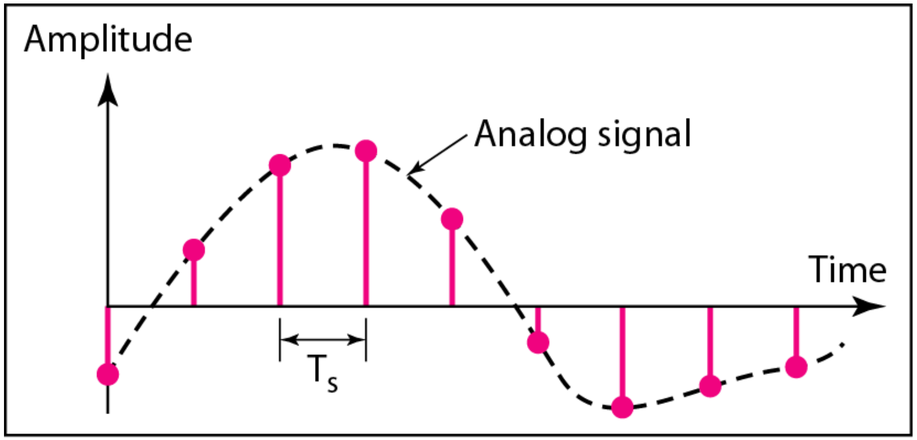
1. The analog signal is sampled.
2. The sampled signal is quantized.
3. The quantized values are encoded as streams of bits.



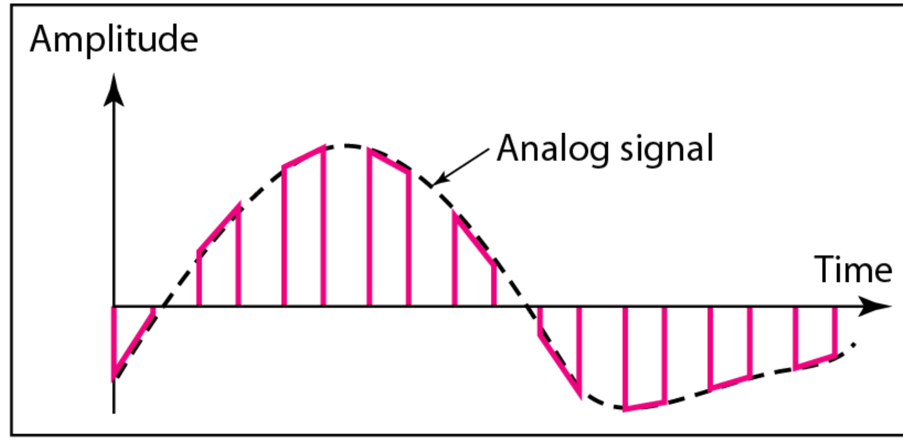
Sampling

- The first step in PCM is sampling.
- The analog signal is sampled every T_s sec, where T_s is the sample interval or period.
- The inverse of the sampling interval is called the sampling rate or sampling frequency and denoted by f_s, where $f_s = 1/T_s$
- There are three sampling methods—**ideal, natural, and flat-top**.
- In **ideal sampling**, pulses from the analog signal are sampled. This is an ideal sampling method and cannot be easily implemented.
- In **natural sampling**, a high-speed switch is turned on for only the small period of time when the sampling occurs.
- The result is a sequence of samples that retains the shape of the analog signal.
- The most common sampling method, called **sample** and **hold.**
- The **flat-top** samples created by using a circuit.
- The sampling process is sometimes referred to as **pulse amplitude modulation (PAM).**

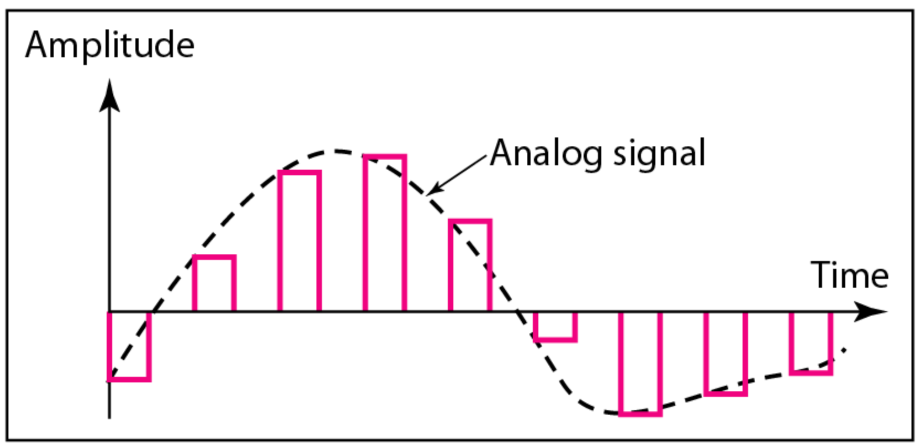
Three different sampling methods for PCM



a. Ideal sampling



b. Natural sampling



c. Flat-top sampling

Sampling Rate

- The important consideration is the sampling rate or frequency
- Defined by **Nyquist theorem.**
- According to the Nyquist theorem, to reproduce the original analog signal, the sampling rate be at least twice the highest frequency in the original signal.
- we can sample a signal only if the signal is band-limited.
- The sampling rate must be at least 2 times the highest frequency, not the bandwidth.
- If the analog signal is low-pass, the bandwidth and the highest frequency are the same value.
- If the analog signal is bandpass, the bandwidth value is lower than the value of the maximum frequency

Sampling Rate

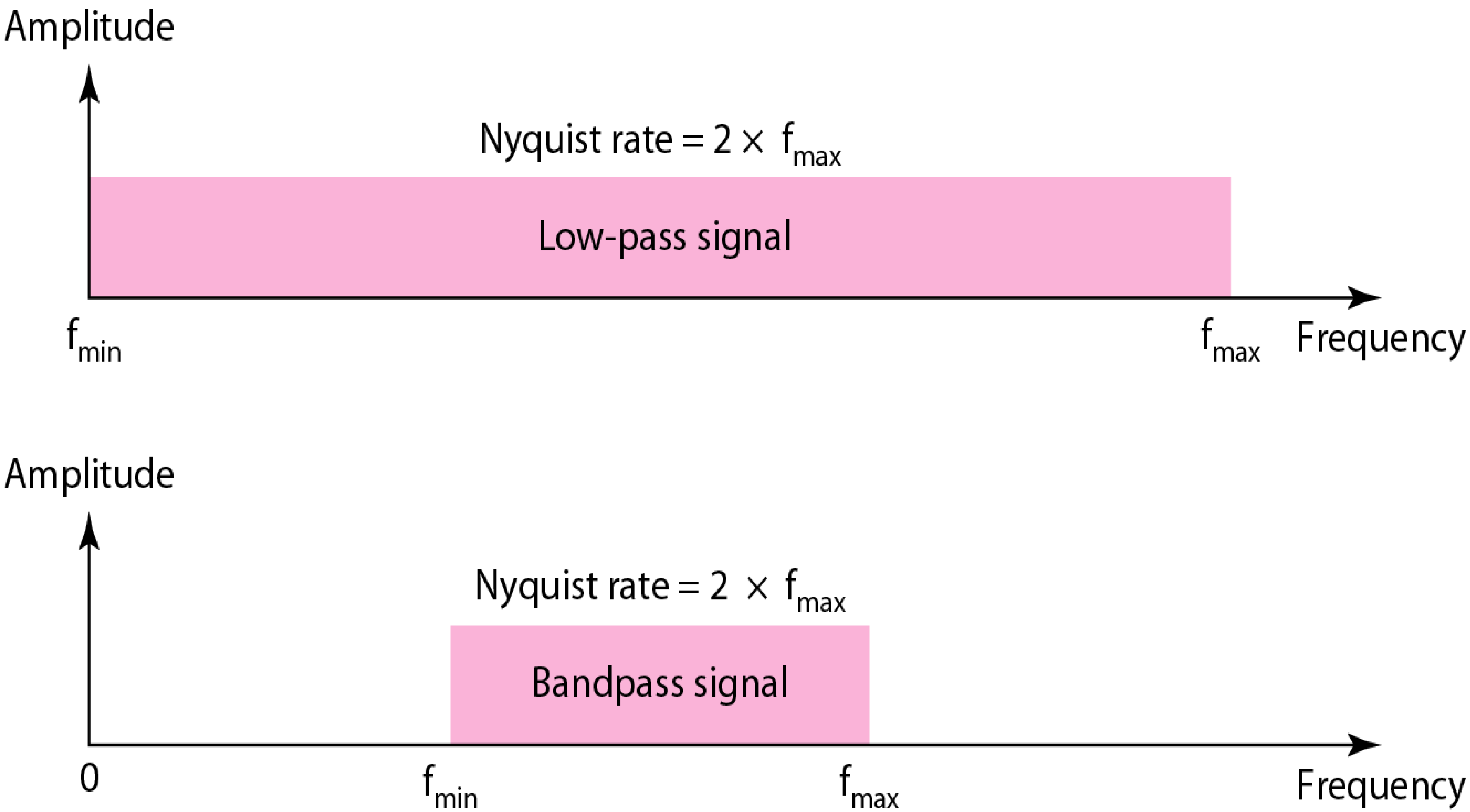


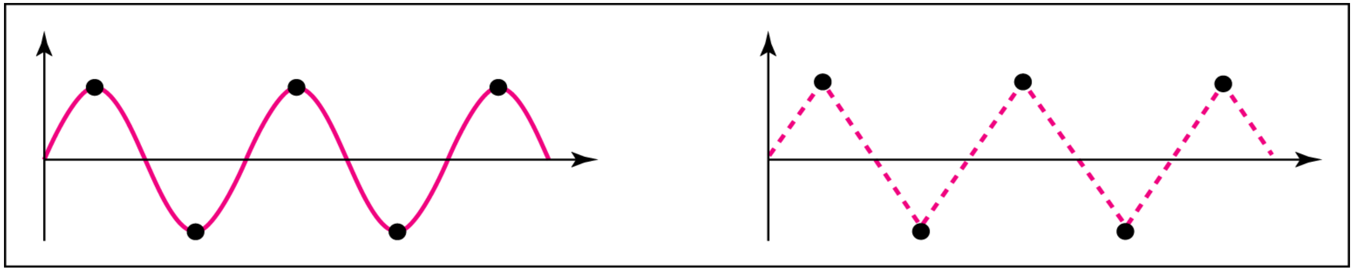
Figure 4.23 ***Nyquist sampling rate for low-pass and bandpass signals***

Example 4.6

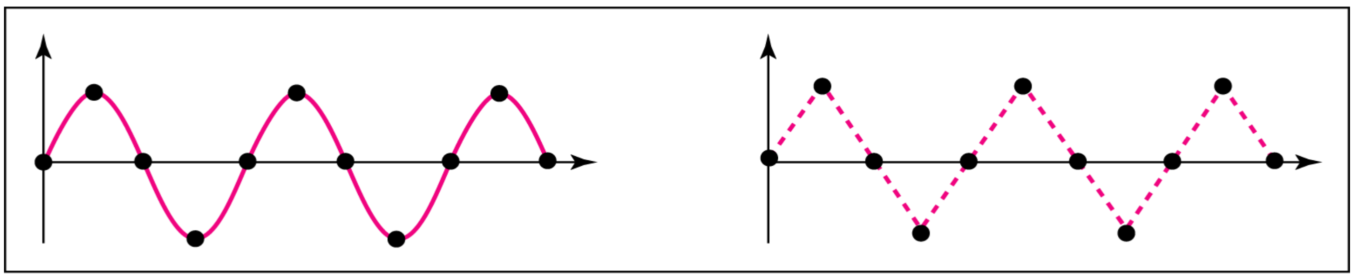
For an intuitive example of the Nyquist theorem, let us sample a simple sine wave at three sampling rates: $f_s = 4f$ (2 times the Nyquist rate), $f_s = 2f$ (Nyquist rate), and $f_s = f$ (one-half the Nyquist rate). Figure 4.24 shows the sampling and the subsequent recovery of the signal.

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.

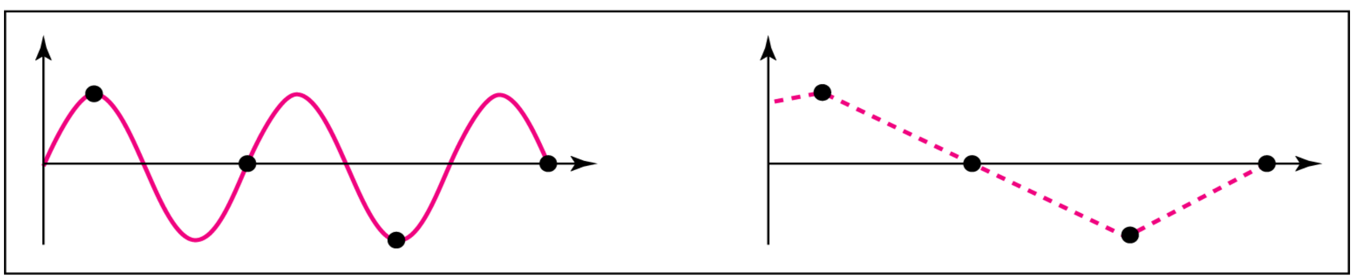
Figure 4.24 *Recovery of a sampled sine wave for different sampling rates*



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



b. Oversampling: $f_s = 4f$



c. Undersampling: $f_s = f$

Example 4.10

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.

Quantization

- The result of sampling is a series of pulses with amplitude values between the maximum and minimum amplitudes of the signal.
- The set of amplitudes can be infinite with nonintegral values between the two limits, which cannot be used for encoding.
- So, we need Quantization. The steps for Quantization are as follows:

1. We assume that the original analog signal has instantaneous amplitudes between *V*min and *V*max.

2. We divide the range into *L* zones, each of height Δ (delta).

$$\Delta = \frac{\text{Vmax} - \text{Vmin}}{L}$$

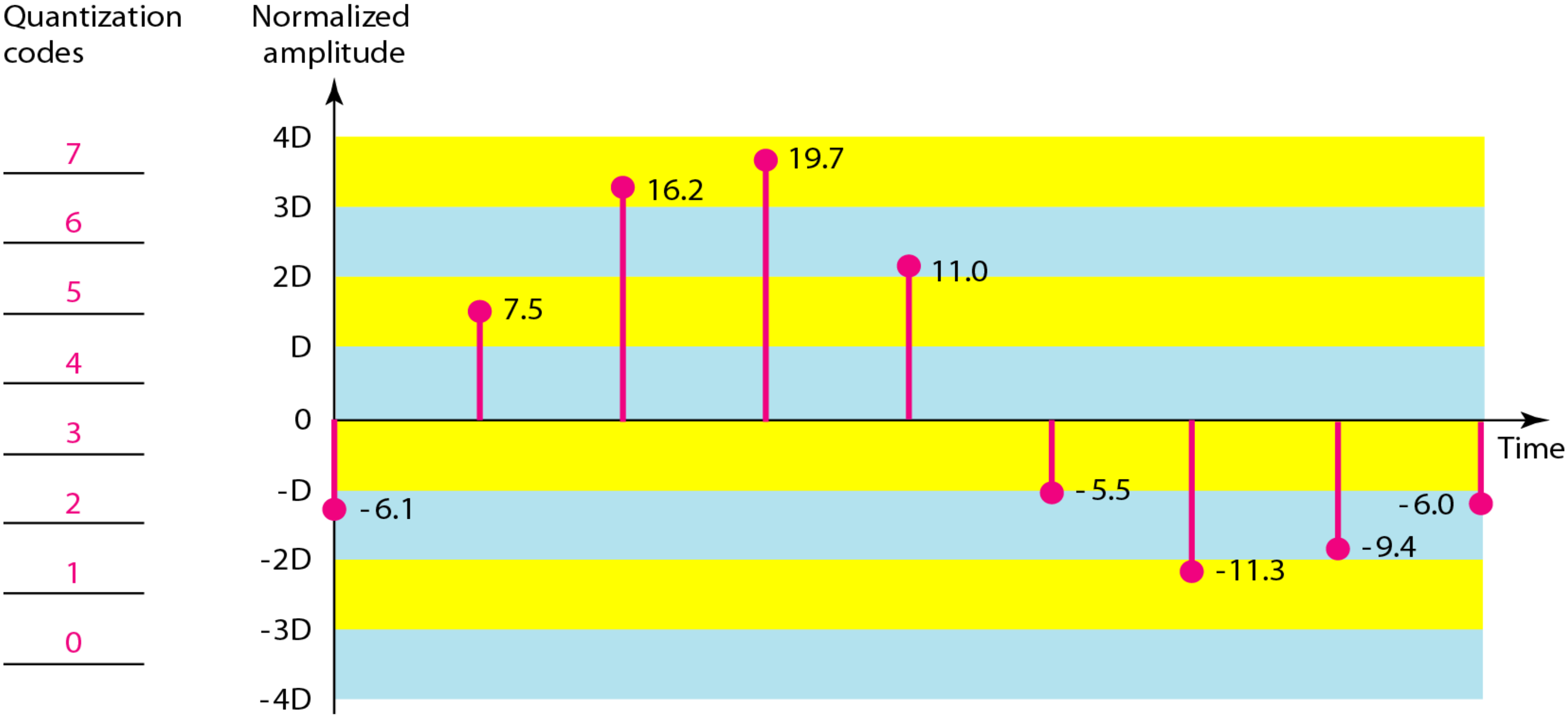
3. We assign quantized values of 0 to *L* - 1 to the midpoint of each zone.

4. We approximate the value of the sample amplitude to the quantized values.

Quantization

- Consider, sampled signal and the sample amplitudes are between -20 and +20 V.
- We decide to have eight levels (L = 8). This means that Δ = 5 V.
- We have shown only nine samples using ideal sampling.
- Actual amplitude is shown in the graph.
- Normalized value for each sample is calculated for actual amplitude/Δ.
- The quantization process selects the quantization value from the middle of each zone. This means that the normalized quantized values (second row).
- The difference is called the *normalized error* (third row).
- The fourth row is the quantization code for each sample based on the quantization levels at the left of the graph.
- The encoded words (fifth row) are the final products of the conversion to binary.

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	-0.28	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

- In audio digitizing, *L* is normally chosen to be 256; in video it is normally thousands.
- Choosing lower values of *L* increases the quantization error if there is a lot of fluctuation in the signal.
- Quantization is an approximation process.
- Input is real value and output is approximation.
- Error occurs only when the input value is not the middle of the level.
- The quantization error changes the signal-to-noise ratio of the signal, which in turn **reduces the upper limit capacity according to Shannon**.
- **Quantization error** to the SNRdB of the signal depends on the number of quantization levels *L*, or the bits per sample *nb*, with formula.

$$\mathrm{SNR}_{\mathrm{dB}} = 6.02 n_b + 1.76 \quad \mathrm{dB}$$

Example 4.12

What is the SNR_{dB} in the example of Figure 4.26? Means, if we have eight levels and 3 bits per sample what will be the SNR_{dB} ?

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

$$\boxed{\mathrm{SNR_{dB}} = 6.02n_b + 1.76 = 40 \quad \Rightarrow \quad n = 6.35}$$

Telephone companies usually assign 7 or 8 bits per sample.

Encoding

- The last step in PCM is encoding. After each sample is quantized and the number of bits per sample is decided, each sample can be changed to an *nb*-bit code word.
- Last row in the figure of quantization.
- A quantization code of 2 is encoded as 010; 5 is encoded as 101; and so on.
- If the number of quantization levels is *L*, the number of bits is ***nb* = log2 *L*.**
- The bit rate can be found from the formula:

$$\text{Bit rate} = \text{sampling rate} \times \text{number of bits per sample} = f_s \times n_b$$

Example 4.14

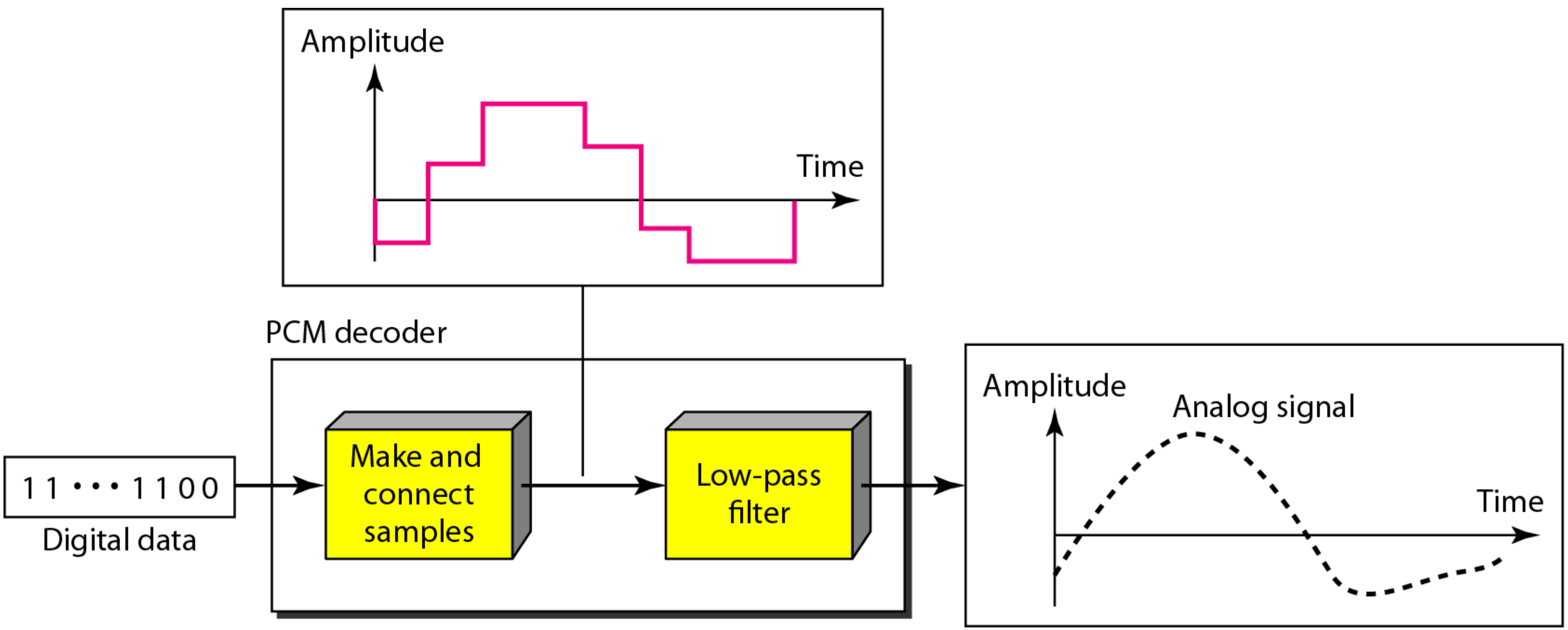
We want to digitize the human voice. What is the bit rate, assuming 8 bits per sample?

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*



PCM Bandwidth the minimum Bandwidth

The minimum bandwidth can be

$$B_{min}=C \times N \times (1/r) \text{ and bit rate, } N=f_s \times n_b$$

$$\text{so, } B_{min}=C \times f_s \times n_b \times (1/r)$$

$$B_{min}=C \times \boxed{2 \times B_{analog}} \times n_b \times (1/r)$$

According to Nyquist theorem

For NRZ or bipolar signal 1/r=1 and c=1/2 for average situation, the minimum bandwidth can be

$$B_{min}=n_b \times B_{analog}$$

This means the minimum bandwidth of the digital signal is *nb* times greater than the bandwidth of the analog signal.

Maximum Data Rate of a Channel & Minimum Required Bandwidth

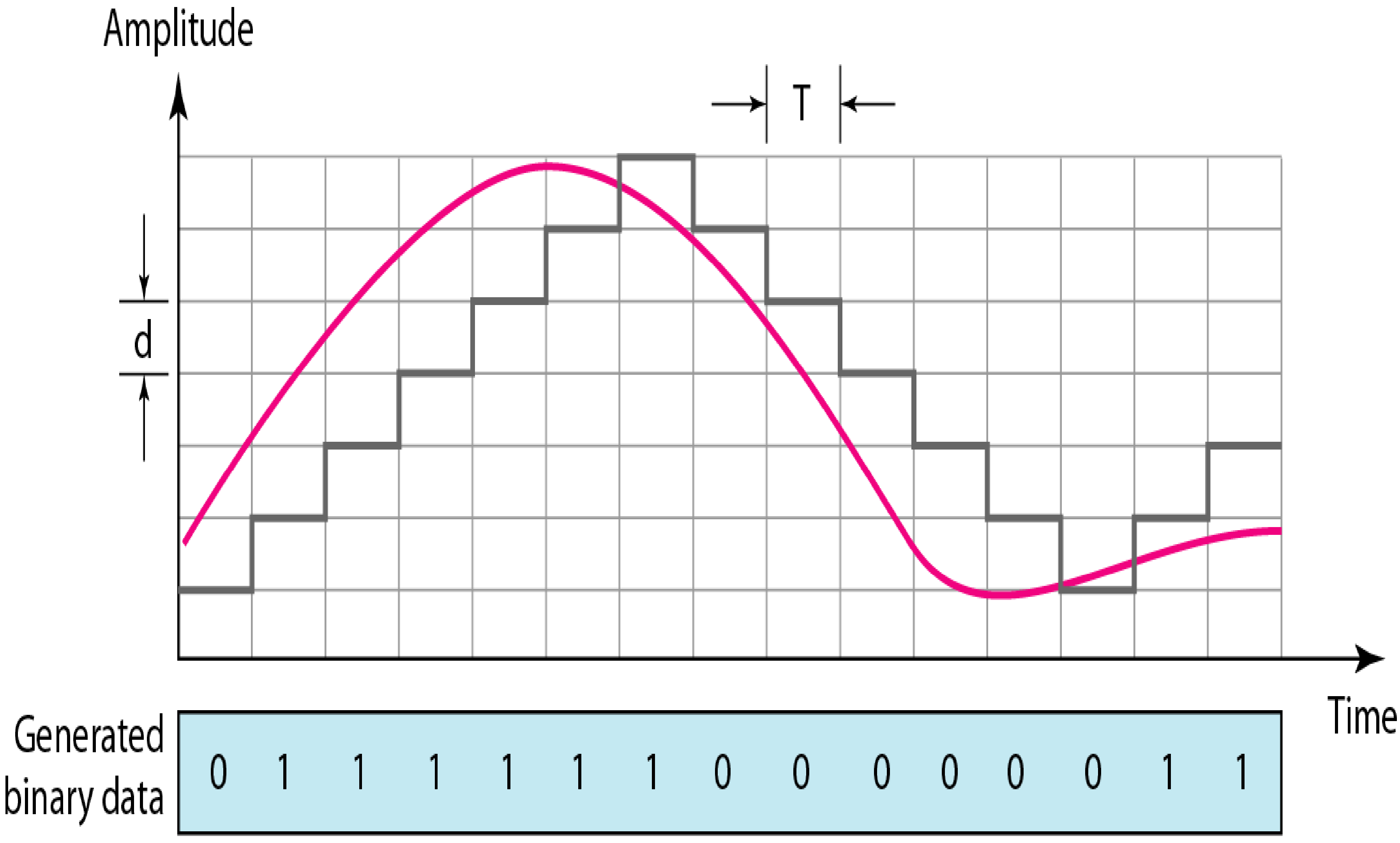
$$N_{max} = 2 \times B \times \log_2 L \quad \text{bps}$$

$$B_{min} = \frac{N}{(2 \times \log_2) L} \quad \text{Hz}$$

Delta Modulator

- PCM is a very complex technique. Other techniques have been developed to reduce the complexity of PCM.
- The simplest is **delta modulation.**
- PCM finds the value of the signal amplitude for each sample; DM finds the change from the previous sample.
- Note that there are no code words here; bits are sent one after another.

Process of delta modulation

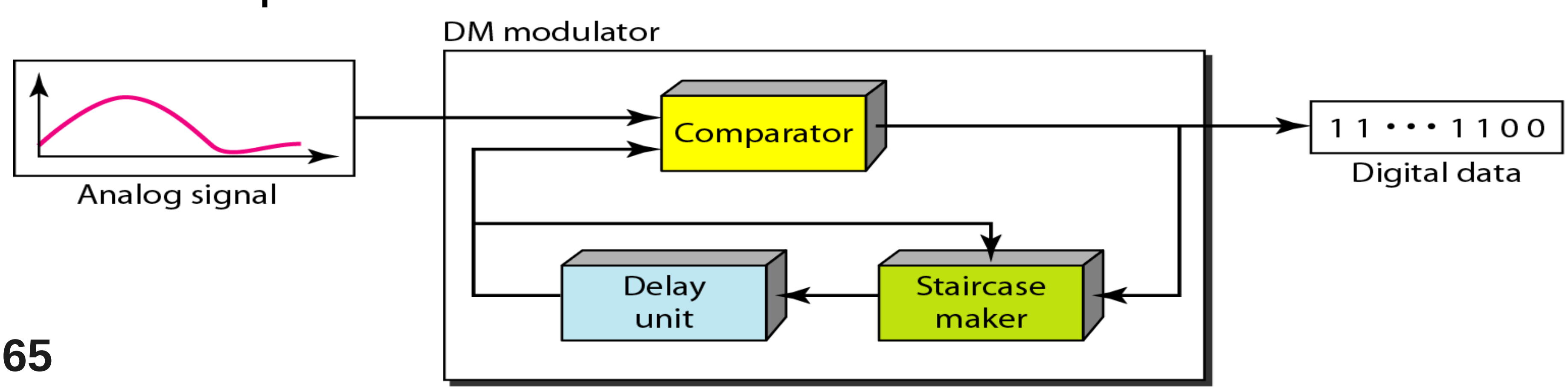


Modulator Operation

- The modulator is used at the sender site to create a stream of bits from an analog signal.
- The process records the small positive or negative changes, called delta δ.
- If the delta is positive, the process records a 1; if it is negative, the process records a 0.
- Base of comparison is required Which is done by Staircase Maker.
- The modulator, at each sampling interval, compares the value of the analog signal with the last value of the staircase signal.
- The modulator builds a second signal that resembles a staircase comparing the change of the analog signal.
- Note that we need a delay unit to hold the staircase function for a period between two comparisons.

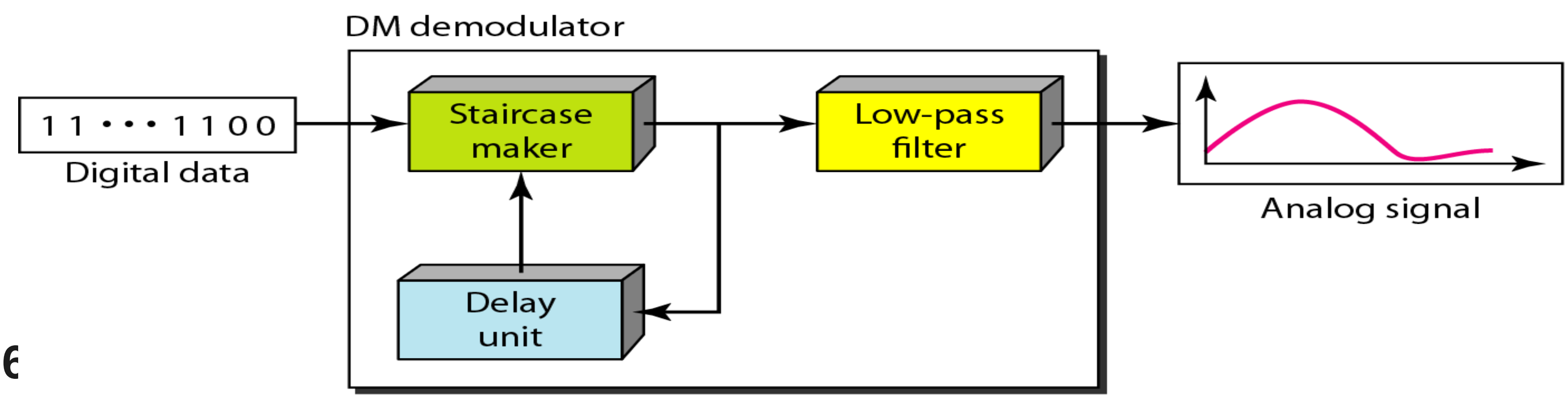
Delta modulation

- The modulator, at each sampling interval, compares the value of the analog signal with the last value of the staircase signal.
- If the amplitude of the analog signal is larger, the next bit in the digital data is 1; otherwise, it is 0.
- The output of the comparator makes the staircase itself. If the next bit is 1, the staircase maker moves the last point of the staircase signal δ up and
- If the next bit is 0, it moves it δ down.
- Delay unit to hold the staircase function for a period between two comparisons.



Delta demodulation

- Demodulator: The demodulator takes the digital data and, using the staircase maker and the delay unit, creates the analog signal.
- Low-pass filter is used for smoothing.
- Adaptive DM: A better performance can be achieved if the value of δ is not fixed. In adaptive delta modulation, 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. The quantization error of DM, however, is much less than that for PCM.



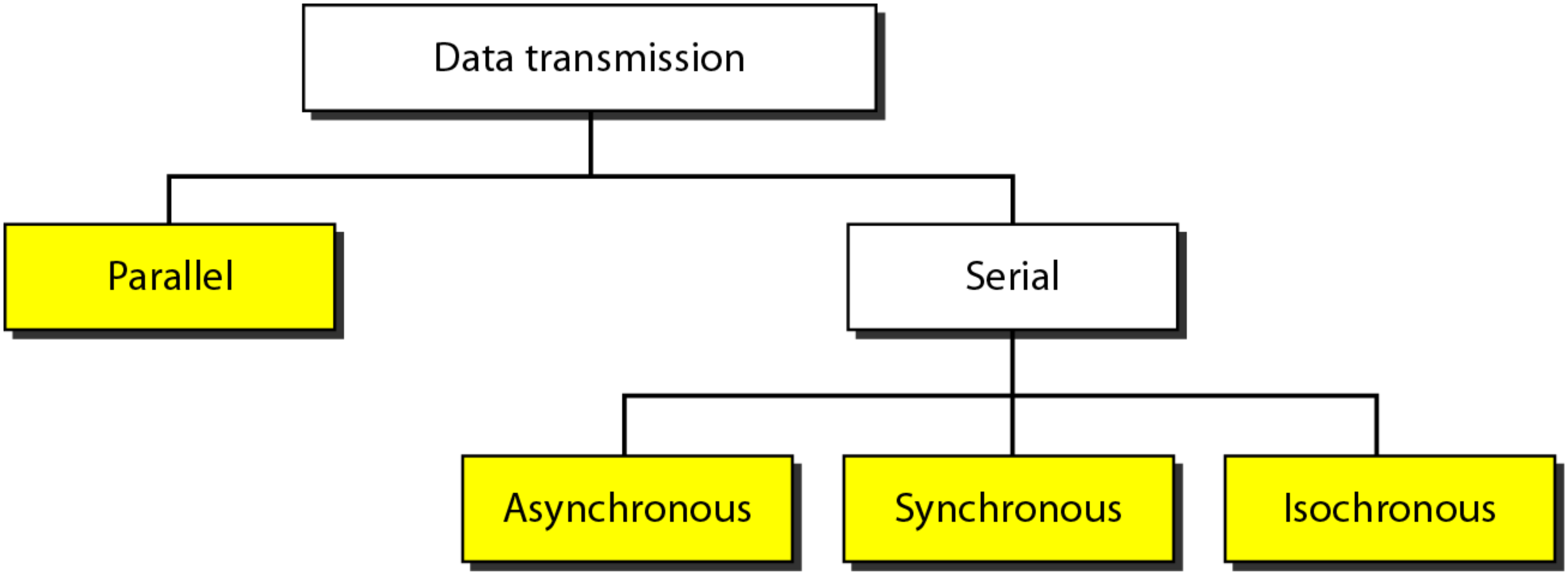
4-3 TRANSMISSION MODES

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:

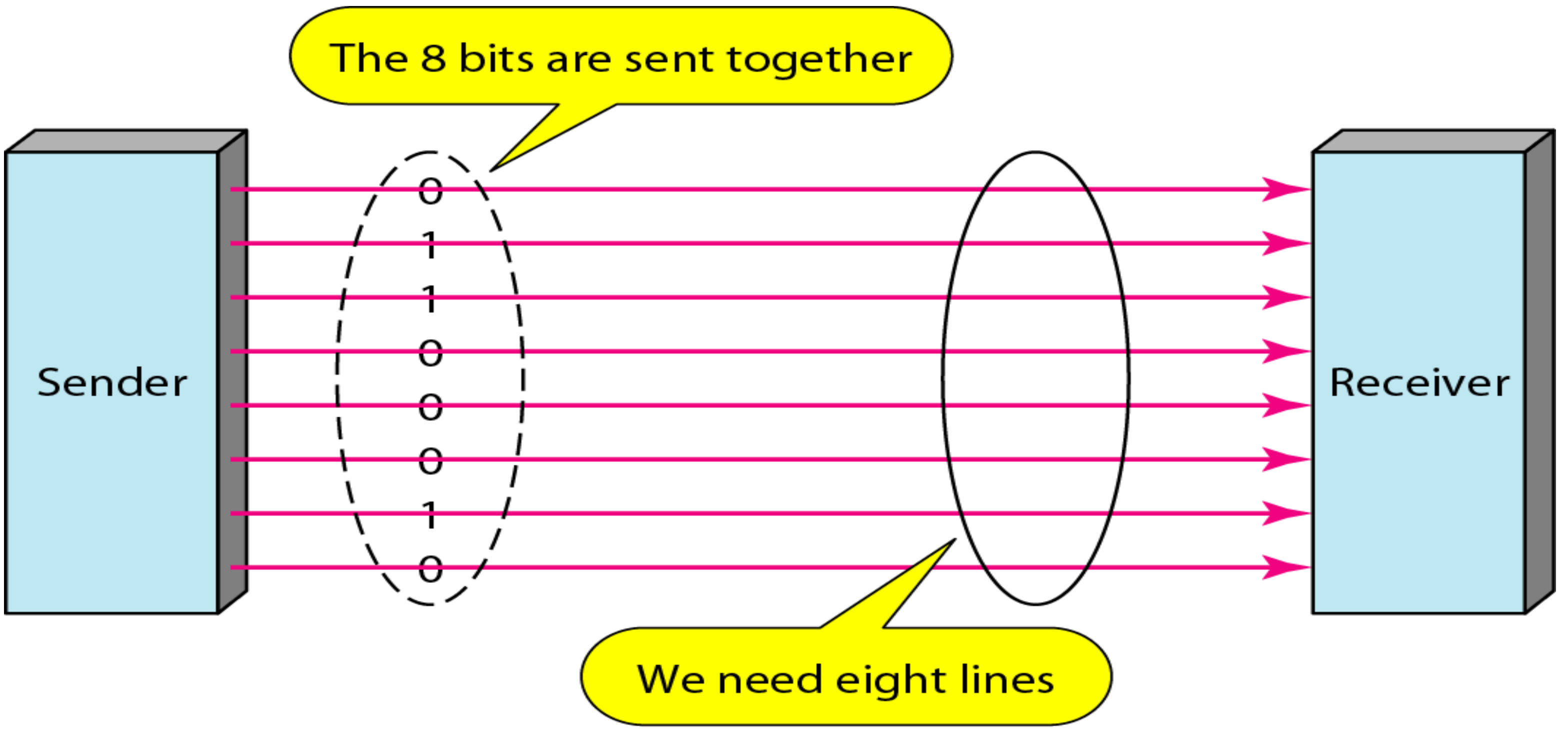
Parallel Transmission
Serial Transmission

Data transmission and modes



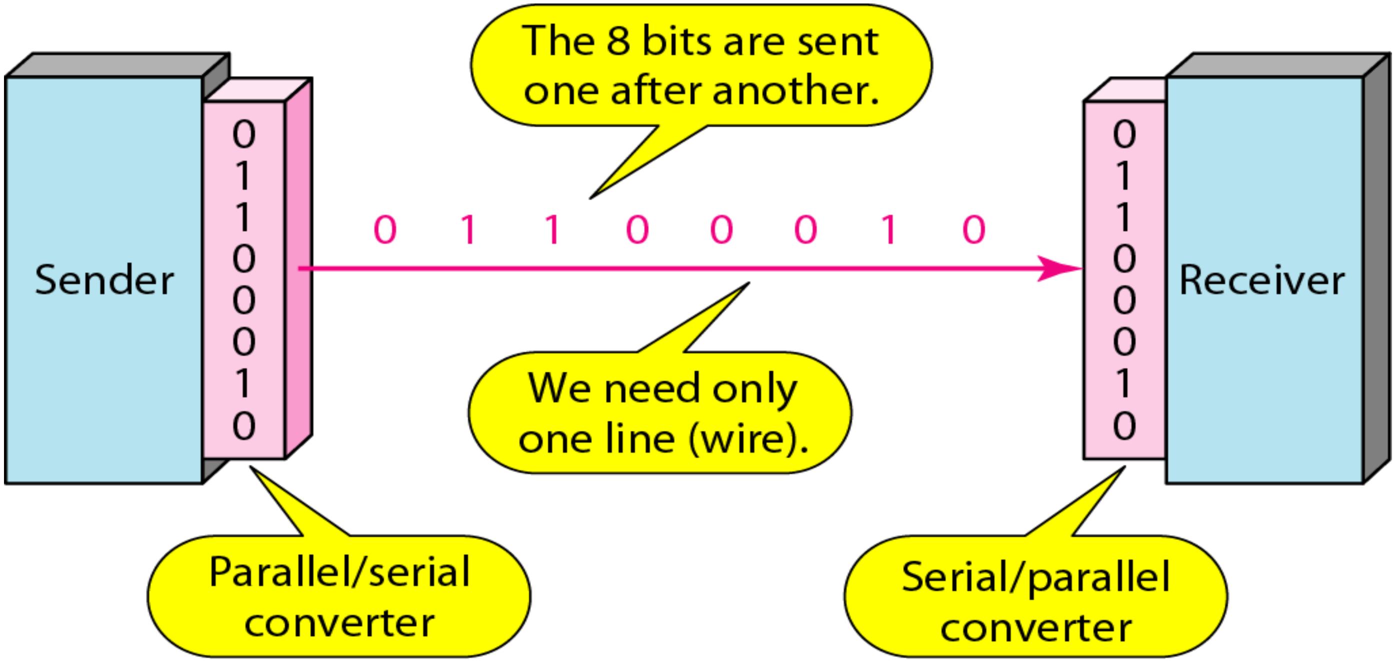
Parallel transmission

- Binary data, consisting of 1s and 0s, may be organized into groups of n bits each.
- By grouping, we can send data n bits at a time instead of 1.
- In parallel transmission Use n wires to send n bits at one time.
- Figure shows how parallel transmission works for n = 8



Serial transmission

- In serial transmission one bit follows another.
- Only one communication channel rather than n to transmit data between two communicating devices.
- The advantage of serial over parallel transmission is that with only one communication channel.
- Serial transmission reduces the cost of transmission over parallel by roughly a factor of n.



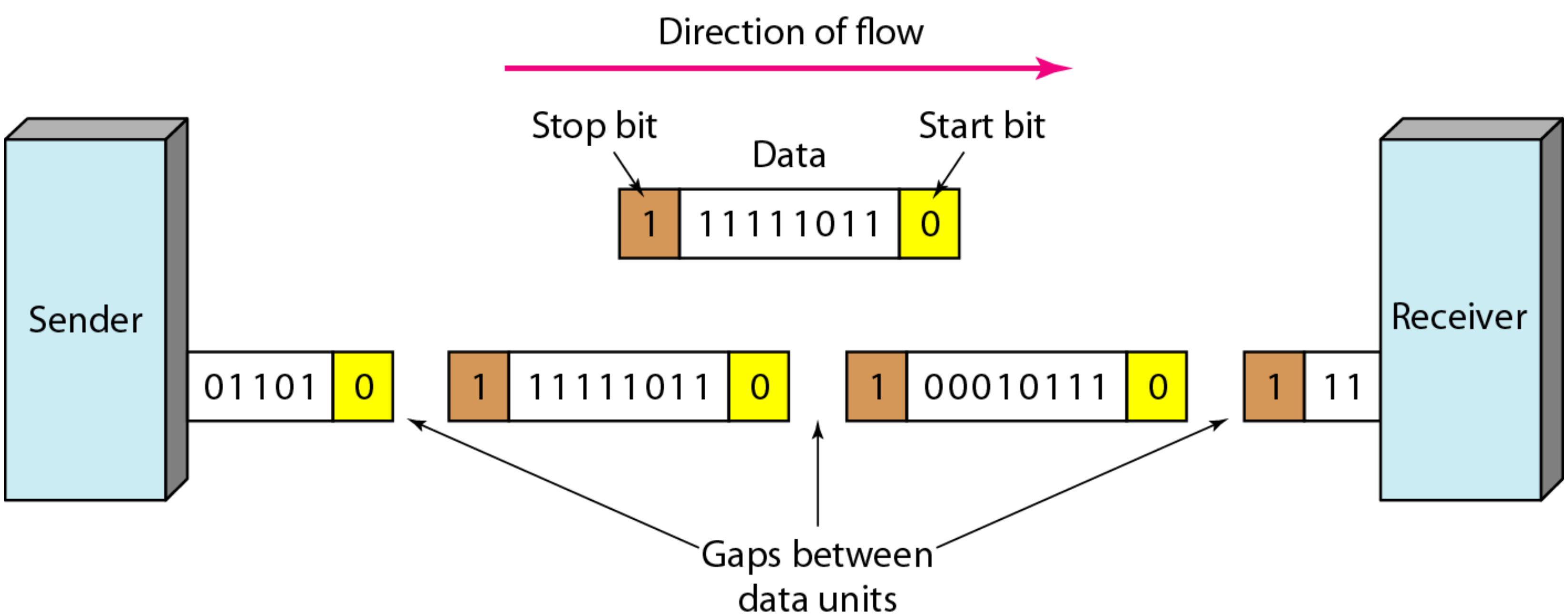
Note

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.

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

Asynchronous transmission

- 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.
- Each byte is increased in size to at least 10 bits, of which 8 bits is information and 2 bits or more are signals to the receiver.
- The start and stop bits and the gap alert the receiver to the beginning and end of each byte and allow it to synchronize with the data stream.
- This mechanism is called **asynchronous** because, at the byte level, the sender and receiver do not have to be synchronized.

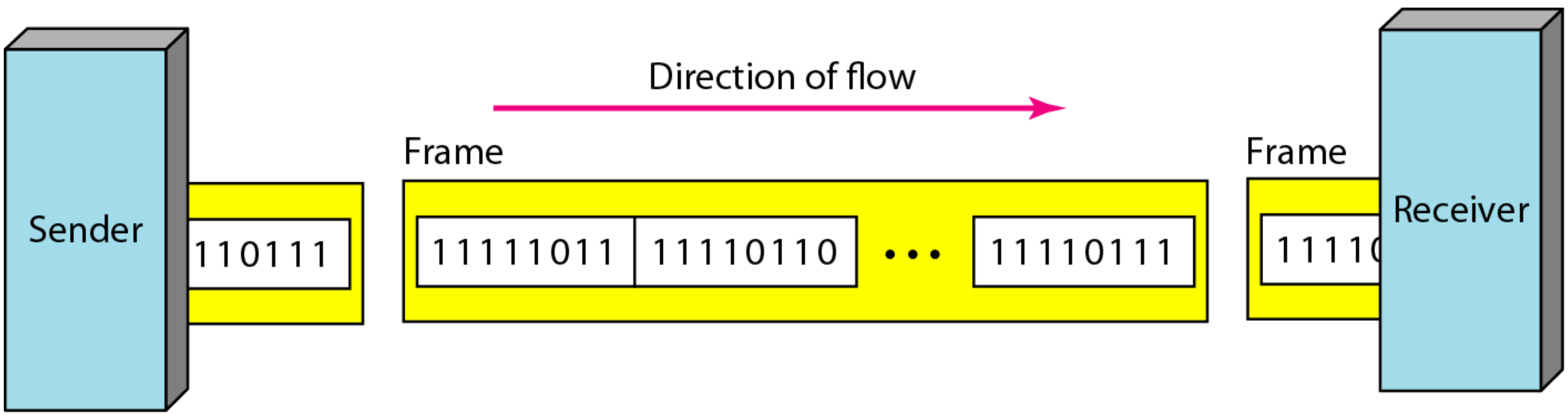


Note

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.

synchronous transmission

- In synchronous transmission, the bit stream is combined into longer "frames," which may contain multiple bytes.
- Each byte is introduced onto the transmission link without a gap between it and the next one.
- Data are transmitted as an unbroken string of 1s and 0s, and the receiver separates that string into the bytes for decoding purposes.



Thank You