# Chapter 4: Digital Transmission

## *Outline*

***4.1 DIGITAL-TO-DIGITAL CONVERSION***

***4.2 ANALOG-TO-DIGITAL CONVERSION***

# 4-1 DIGITAL-TO-DIGITAL CONVERSION

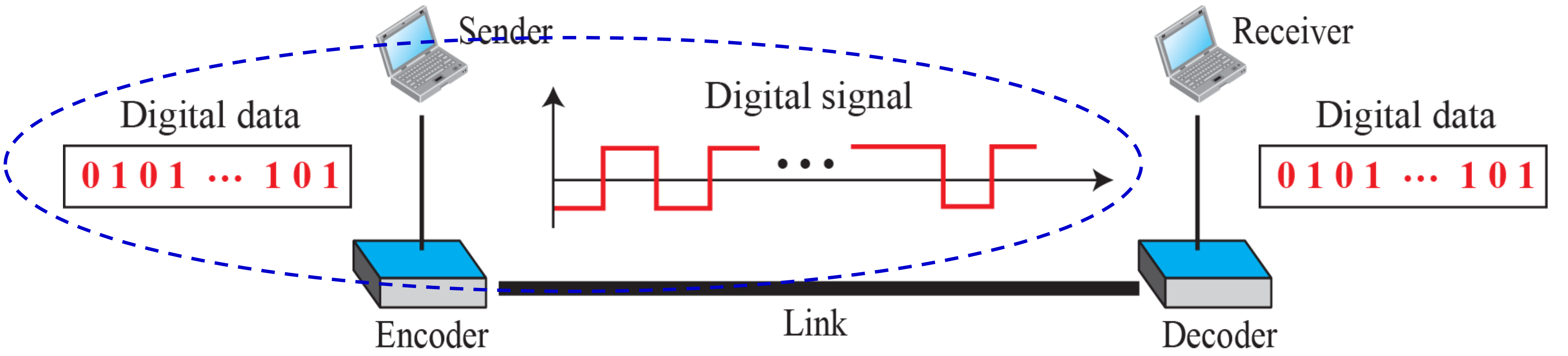
***In Chapter 3, we discussed data and signals. We said that data can be either digital or analog. We also said that signals that represent data can also be digital or analog.***

***In this section, we see how we can represent digital data using digital signals.***

# *4.1.1 Line Coding*

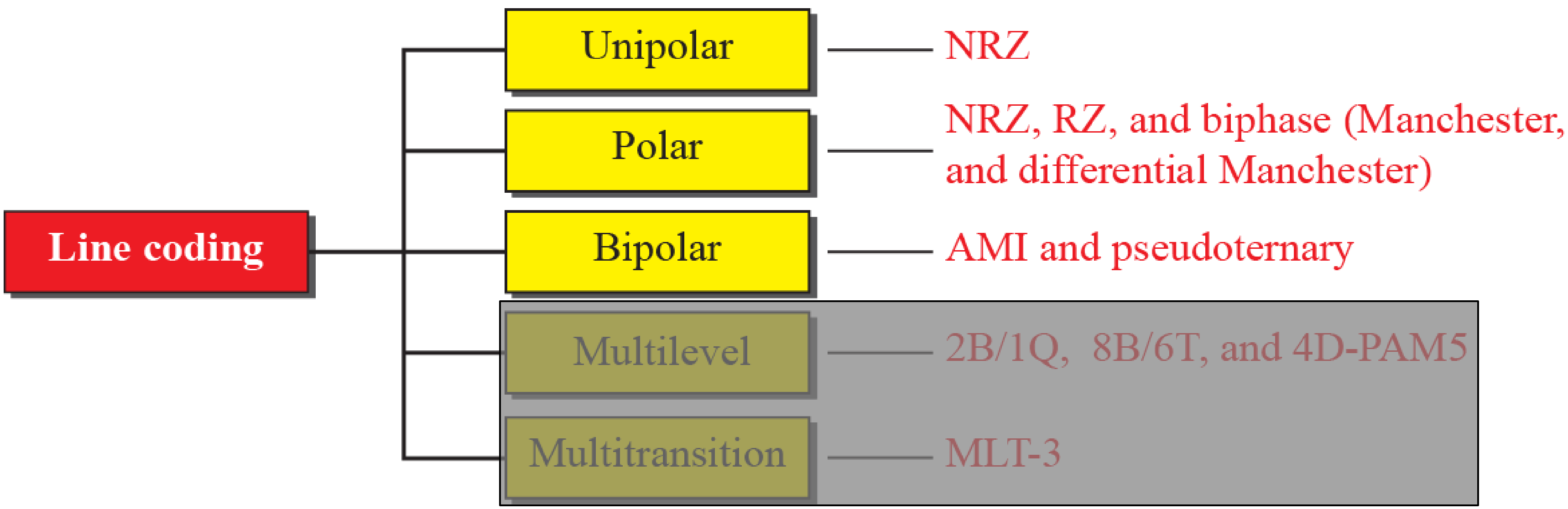
***<u>Line coding</u> is the process of <u>converting digital data to digital signals</u>. It converts a sequence of bits to a digital signal.***

***At the sender, digital data are encoded into a digital signal; at the receiver, the digital data are recreated by decoding the digital signal.***



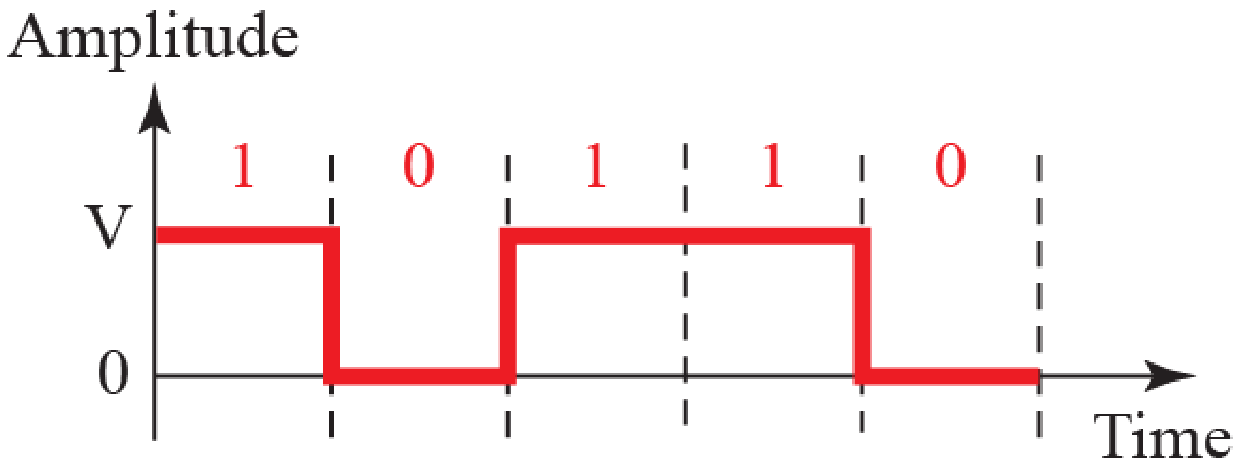
# *4.1.2 Line Coding Schemes*

***We can roughly divide <u>line coding schemes</u> into five broad categories. We will look at three categories: (A) <u>unipolar</u>, (B) <u>polar</u> and (C) <u>bipolar</u>. There are several schemes in each category.***



## *Figure 4.5: (A) Unipolar scheme (NRZ)*

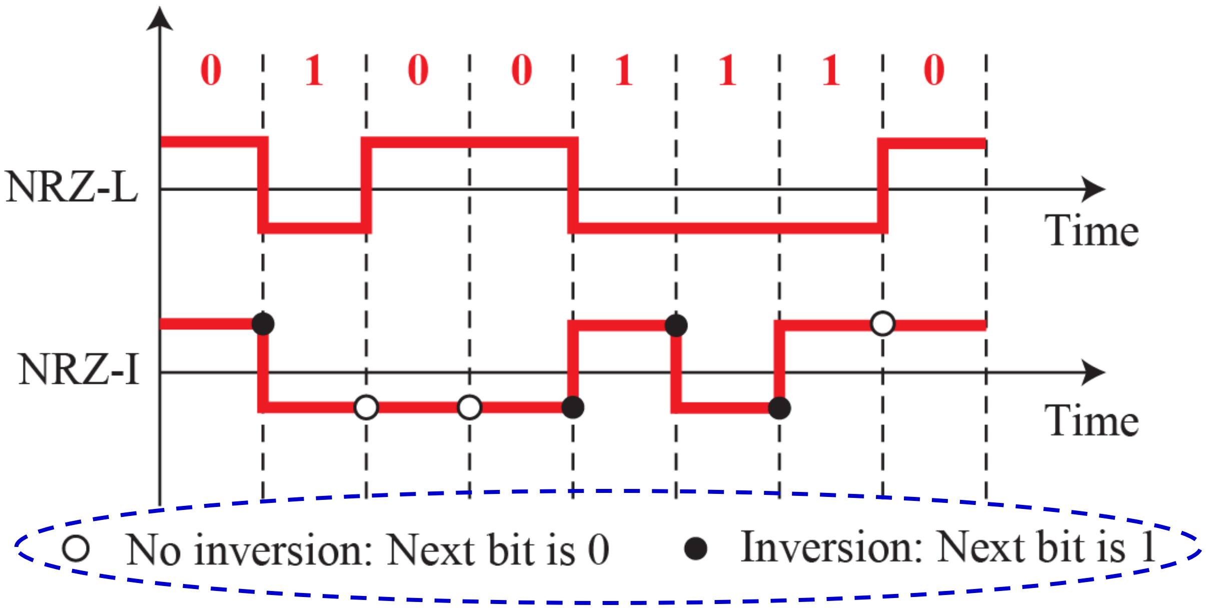
- **In a unipolar scheme, all voltage levels are on one side of the time axis (above or below).**
- **A positive voltage defines bit 1 and a zero voltage defines bit 0.**




**Note:**
**Non-Return-to-Zero (NRZ): signal does not return to zero in the middle of the bit.**

# *Figure 4.6:* *(B.1) Polar schemes (NRZ-L and NRZ-I)*

- **In polar schemes, the voltage levels are on both sides of the time axis.**
- **A positive voltage can define bit 0 and a negative voltage can define bit 1.**



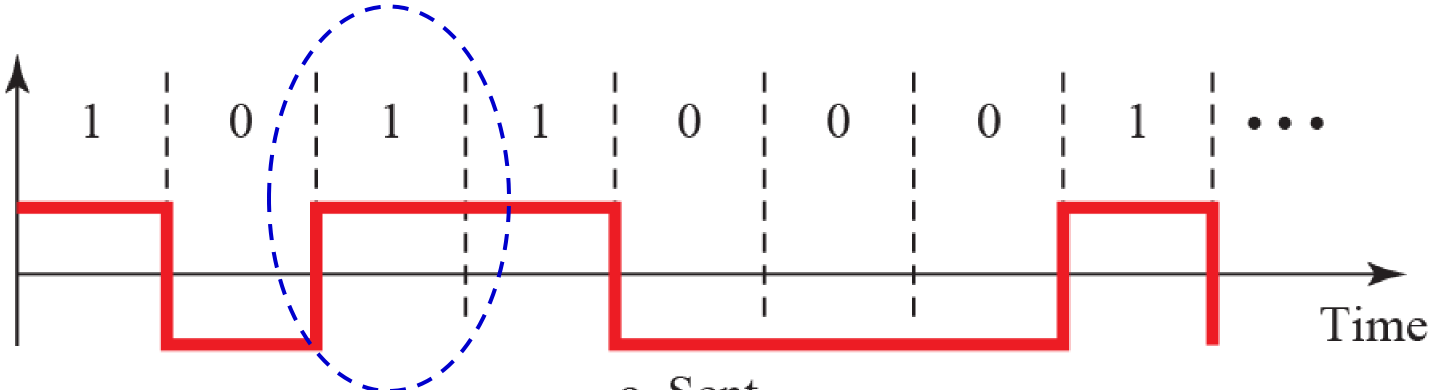

**NRZ-Level (NRZ-L): the level of voltage determines the value of the bit.**

**NRZ-Invert (NRZ-I): the change or lack of change in the voltage level determines the value of the bit.**

**Note: An issue with NRZ encoding schemes is that when the sender and receiver clocks are not synchronized, the receiver does not know when one bit has ended and when the next bit is starting.**

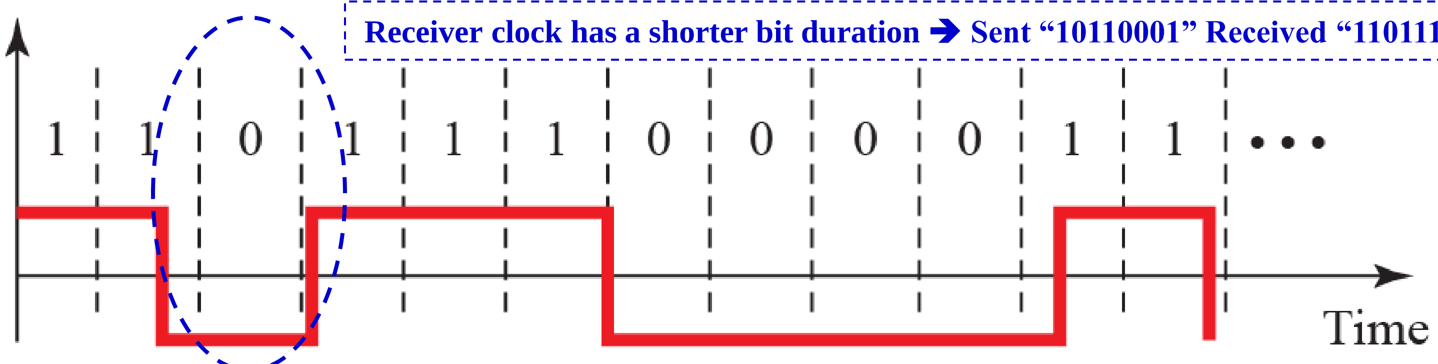
# Clock Synchronization

**To correctly interpret the signals received from the sender, the <u>receiver's bit period</u> needs to correspond to the <u>sender's bit period</u>.**




a. Sent

**Receiver clock has a shorter bit duration → Sent "10110001" Received "110111000011"**




b. Received

**Note: If the receiver clock is faster or slower, the bit intervals are not matched and the receiver might misinterpret the results.**

# Problem

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

## Solution

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

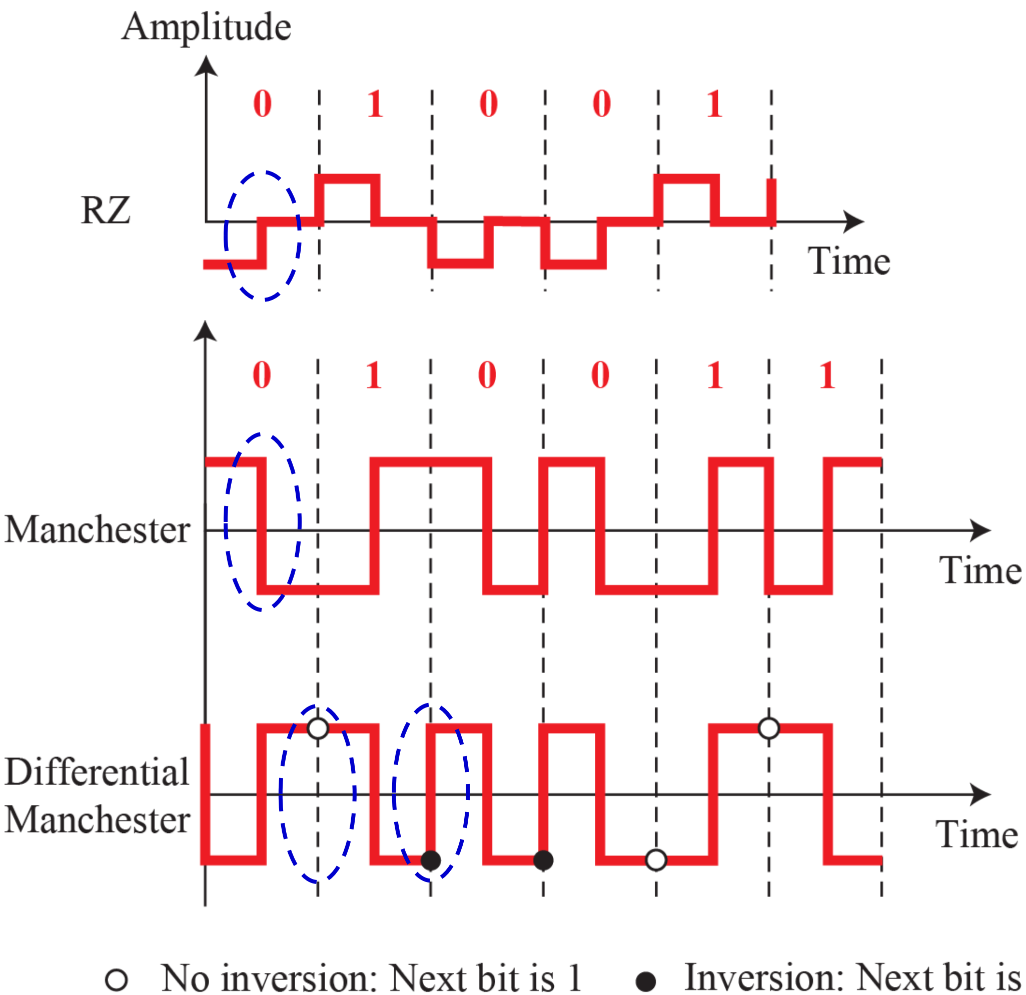
**1000 bits sent → 1001 bits received → 1 extra bps**

b) 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**

# *Figure 4.7:* *(B.2) Polar schemes (RZ, Manchester, Differential Manchester)*

**A solution to the clock synchronization issue in NRZ is return-to-zero (RZ) schemes. In RZ, the signal changes not between bits but during the bit, i.e., self-synchronizing signal that includes timing information in the data being transmitted.**




**RZ: the signal changes during the bit, i.e., signal goes to 0 in the middle of each bit.**

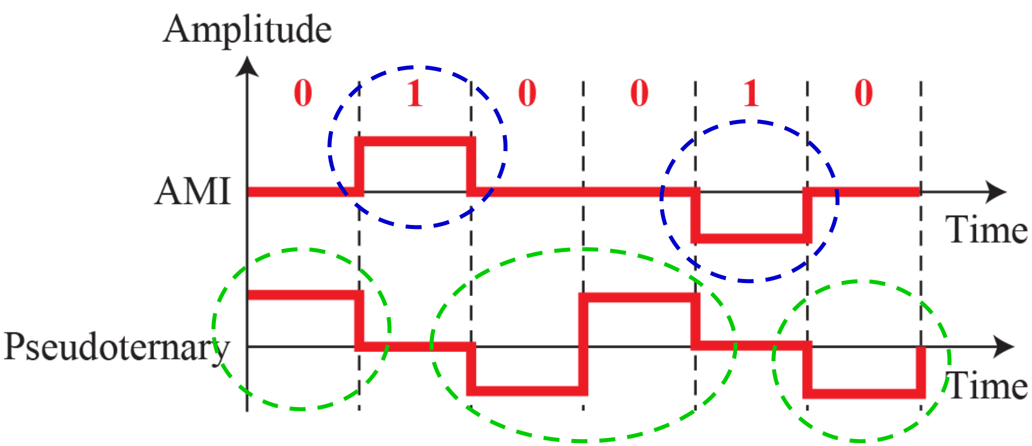
**Manchester: combines RZ and NRZ-L. Voltage remains at one level during the first half and moves to other level in the second half.**

**Differential Manchester: combines RZ and NRZ-I. If the next bit is a 0, there is a transition; if the next bit is 1, there is no transition.**

**The main disadvantage of RZ, Manchester and Differential Manchester schemes is that it requires signal changes to encode a bit and therefore occupies greater bandwidth.**

# *Figure 4.9:* *(C) Bipolar schemes: AMI and Pseudoternary*

- **In bipolar schemes, there are three voltage levels: positive, negative and zero.**
- **The voltage for one data element is at zero, while the voltage level for the other element alternates between positive and negative.**




**Alternate Mark Inversion (AMI): a zero voltage represents bit 0 and bit 1 is represented by alternating positive and negative voltages.**

**Pseudoternary: variation of AMI. Bit 1 is represented by a zero voltage and bit 0 is represented by alternating positive and negative voltages.**

**As with physical topologies, the choice of line coding schemes depends on the desired characteristics. There are many factors for consideration, including: complexity, signaling rate and bandwidth requirement, self-synchronization, etc., (additional details in Section 4.1.2).**

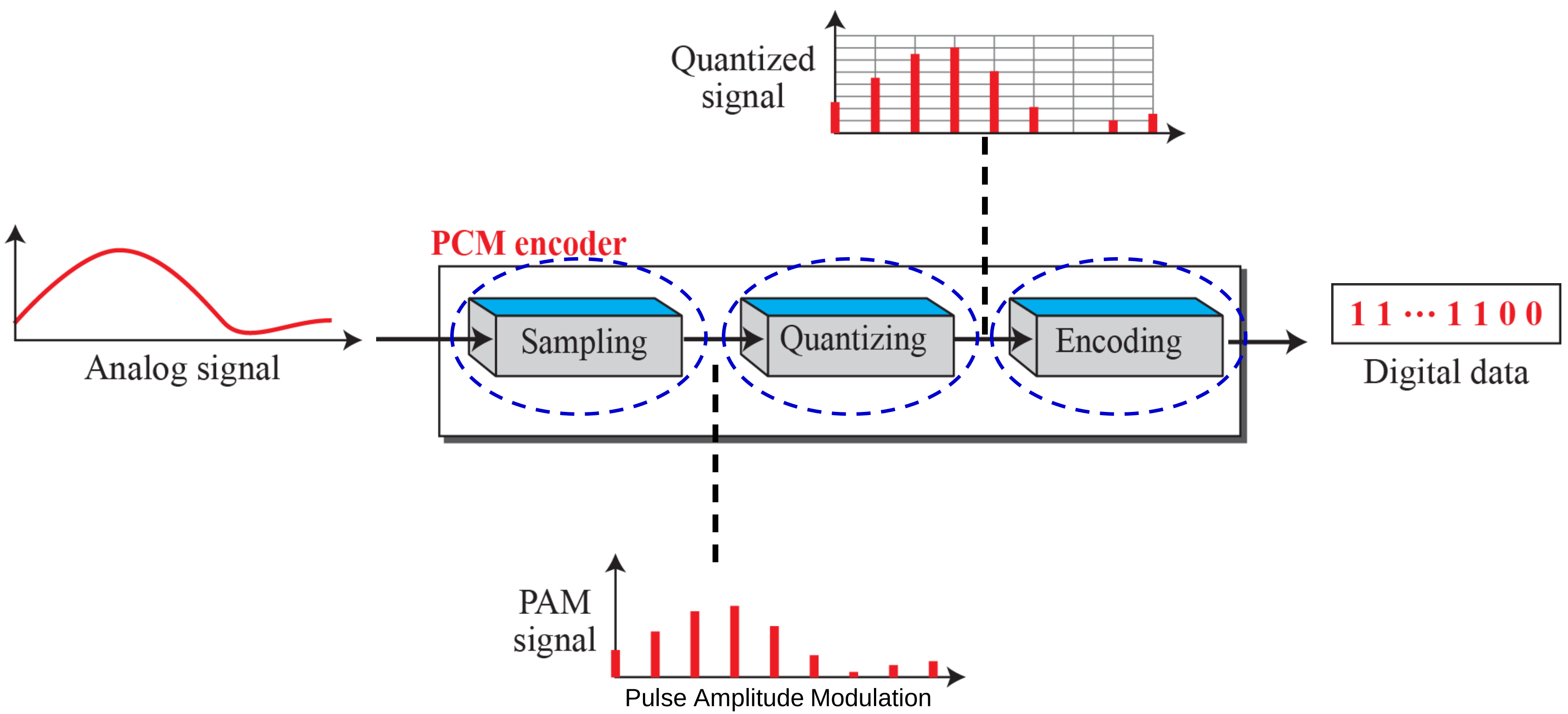
# 4-2 ANALOG-TO-DIGITAL CONVERSION

***The techniques described in Section 4.1 convert digital data to digital signals. Sometimes, however, we have an analog signal such as one created by a microphone or camera.***

***In this section, we see how we can <u>represent analog data</u> using <u>digital signals</u>.***

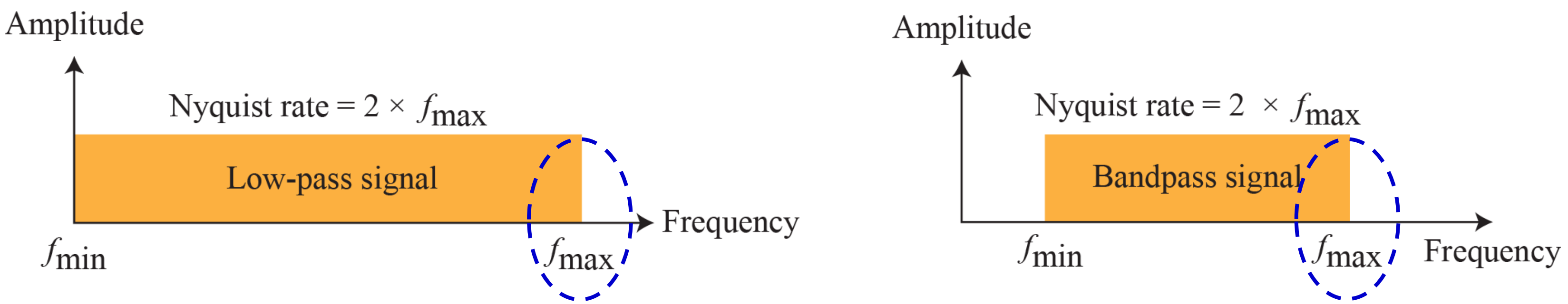
# 4.2.1 Pulse Code Modulation (PCM)

***The most common technique to change an analog signal to digital data (digitization) is called <u>pulse code modulation</u> (PCM). A PCM encoder has three processes (<u>sampling</u>, <u>quantizing</u> and <u>encoding</u>), as shown below:***



# Step 1: Sampling

**Based on the Nyquist theorem, to reproduce the original analog signal, the <u>sampling rate</u> must be <u>at least 2 times</u> the <u>highest frequency</u>, $f_{max}$, contained in the signal.**

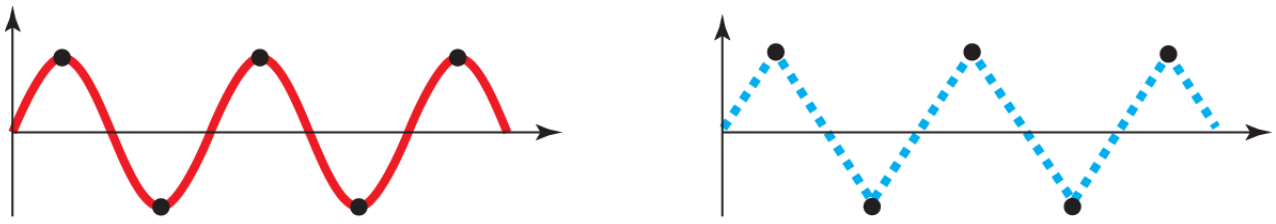



**Nyquist rate, $f_s = 2 \times f_{max}$**

**$(f_{max} \neq B$ *for bandpass signal*$)$**

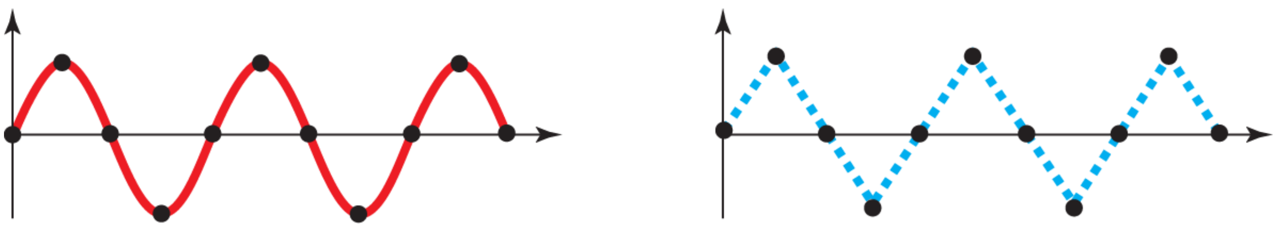
# *Recovery of a sine wave with different sampling rates*

***For an example of the Nyquist theorem, let us sample a simple sine wave with frequency, f, at three sampling rates: (a) Nyquist rate, (b) 2 times the Nyquist rate and (c) ½ the Nyquist rate.***



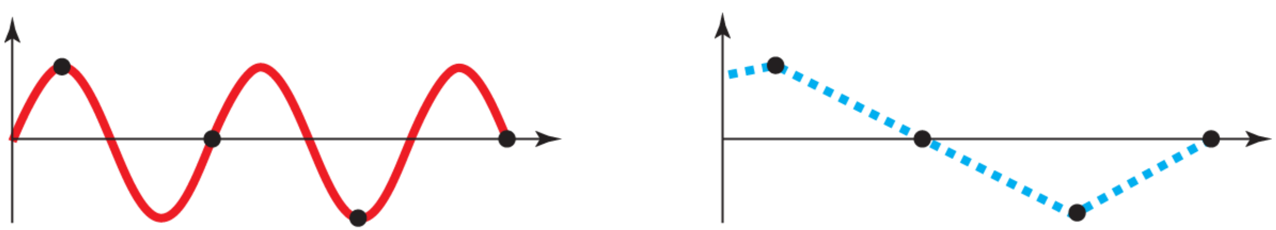
a. Nyquist rate sampling: $f_s = 2\ f$

**$f_s = 2f$ (e.g., Nyquist):**
Can create a good approximation of the original sine wave.



b. Oversampling: $f_s = 4\ f$

**$f_s > 2f$ (e.g., 2 x Nyquist):**
Oversampling can also create the same approximation but it is redundant and unnecessary.

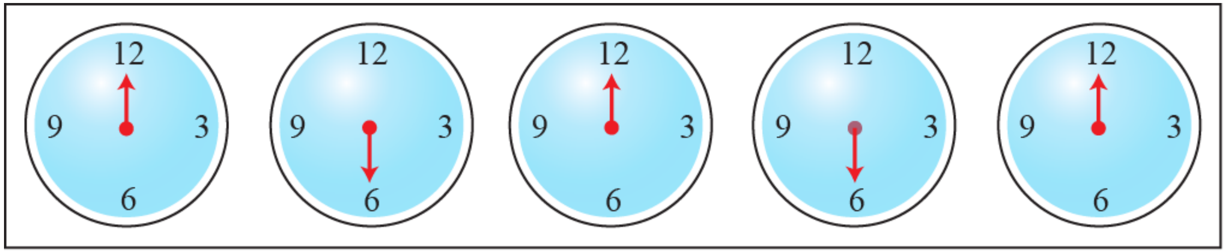


c. Undersampling: $f_s = \ f$

**$f_s < 2f$ (e.g., ½ Nyquist):**
Undersampling does not produce a signal that looks like the original sine wave 🡐 **<u>aliasing</u>** (note that we are not only losing information, we are getting the wrong information about the signal).

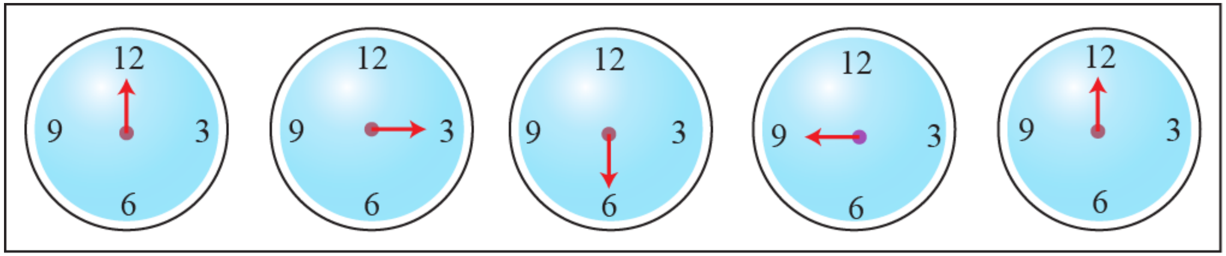
# Example

An interesting example: let's sample a periodic event such as the revolution of a clock with only a second hand (i.e., no hour hand) and has a period, $T$, of 60 sec.



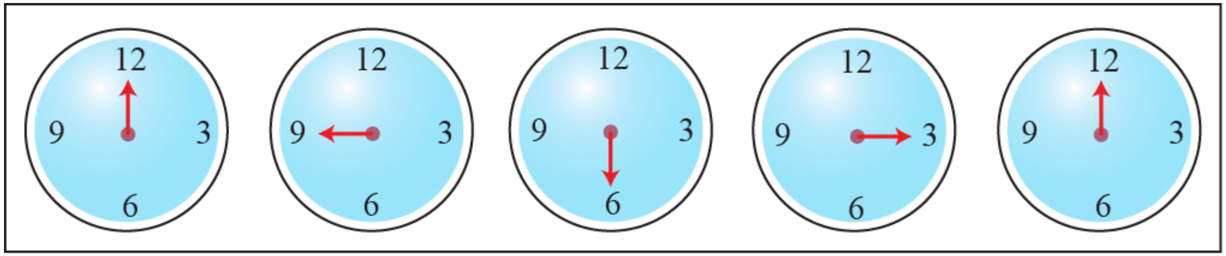
Samples can mean that the clock is moving either forward or backward. (12-6-12-6-12)

a. Sampling at Nyquist rate: $T_s = T\frac{1}{2}$ = 30 sec



Samples show clock is moving forward. (12-3-6-9-12)

b. Oversampling (above Nyquist rate): $T_s = T\frac{1}{4}$ = 15 sec



Samples show clock is moving backward. (12-9-6-3-12)

c. Undersampling (below Nyquist rate): $T_s = T\frac{3}{4}$ = 45 sec

# Problem

**A <u>low-pass</u> 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 $f_{max}$, where $f_{max}$ is the maximum frequency in the signal. Therefore, we can sample this signal at 2 times of $f_{max}$ (200 kHz). The sampling rate is therefore 2 x 200 k = 400,000 samples per second.

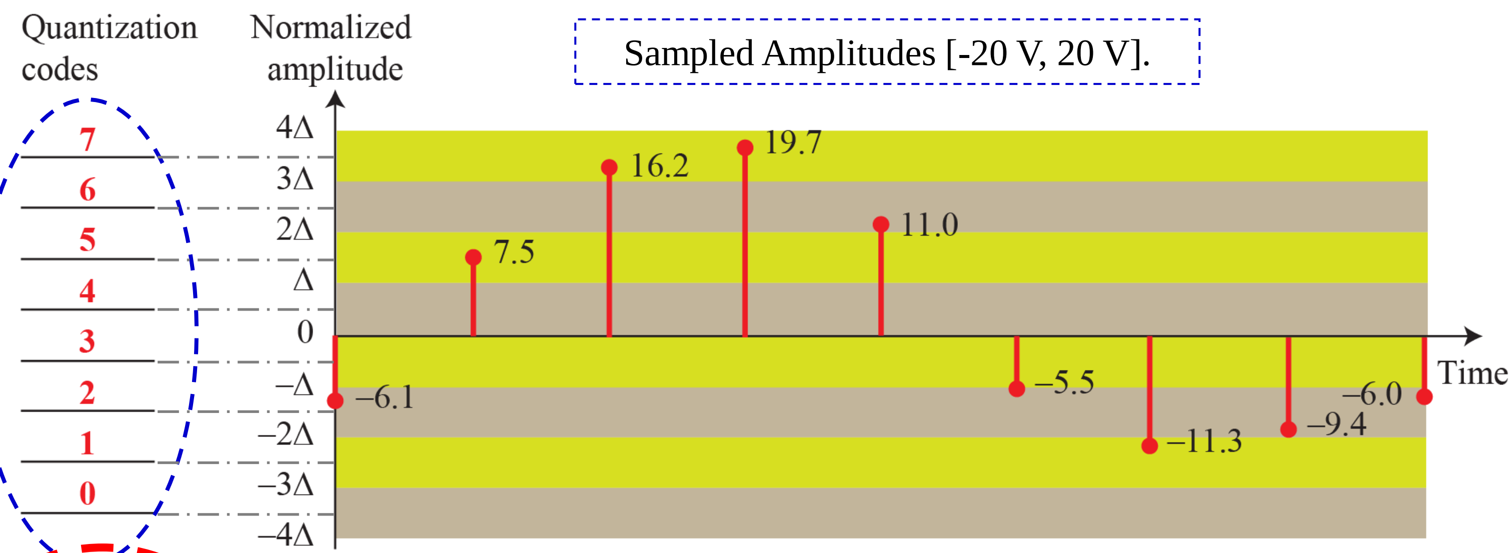
**A <u>band-pass</u> signal has a bandwidth of 200 kHz. What is the minimum sampling rate for this signal?**

## Solution

We cannot find the minimum sampling rate in this case because we do not know where the bandwidth starts or ends, i.e., we do not know the maximum frequency, $f_{max}$, in the signal.

# Step 2 & 3: Quantizing and Encoding

**Step 2: Quantization**




| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 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 Error**

**Step 3: Encoding**

# Quantization Error

**Quantization is an approximation process and an important issue is the error created. The quantization error changes the signal-to-noise ratio of the signal, which reduces the limit of the Shannon capacity.**

**The signal strength in relation to the quantization error, $SNR_Q$, in dB, is estimated by**

$$\boxed{SNR_Q = 6.02 n_b + 1.76}$$

**where $n_b = \log_2 L$, is the number of bits per sample.**

# Problem

A telephone subscriber line must have a quantizing signal-to-noise ratio of no less than 40 dB. What is the minimum number of bits per sample?

**Solution**

We can calculate the number of bits as

$$SNR_Q = 6.02n_b + 1.76 \geq 40 \text{ dB}$$
$$n_b \geq 6.352$$
$$n_b \geq 7 \text{ bits per sample}$$

➔ Telephone companies usually assign 7 or 8 bits per sample.

# Step 3: Encoding and Bit rate

The last step in PCM is encoding. After each sample is quantized and the number of bits per sample, $n_b$, is determined, each sample is represented by an $n_b$-bit code word. (Recall if the number of quantization levels is $L$, then the number of bits per sample is $n_b = \log_2 L$.)

The bit rate can be found as

$$\boxed{\text{Bit rate} = f_s \times n_b}$$

where $f_s$ is the sampling rate and $n_b$ is the number of bits per sample.

# Problem

We want to digitize the human voice. The human voice normally contains frequencies from 0 to 4000 Hz. What is the bit rate, assuming 8 bits per sample?

## Solution

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

$$\text{Sampling rate} = 4000 \times 2 = 8000 \text{ samples/s}$$

$$\text{Bit rate} = 8000 \times 8 = 64{,}000 \text{ bps} = 64 \text{ kbps}$$