1985

جامعة محمد بوضياف - المسيلة
Université Mohamed Boudiaf - M'sila

Computer Networks

Lecture 2 : OSI Physical layer

By : Prof. Lamri SAYAD

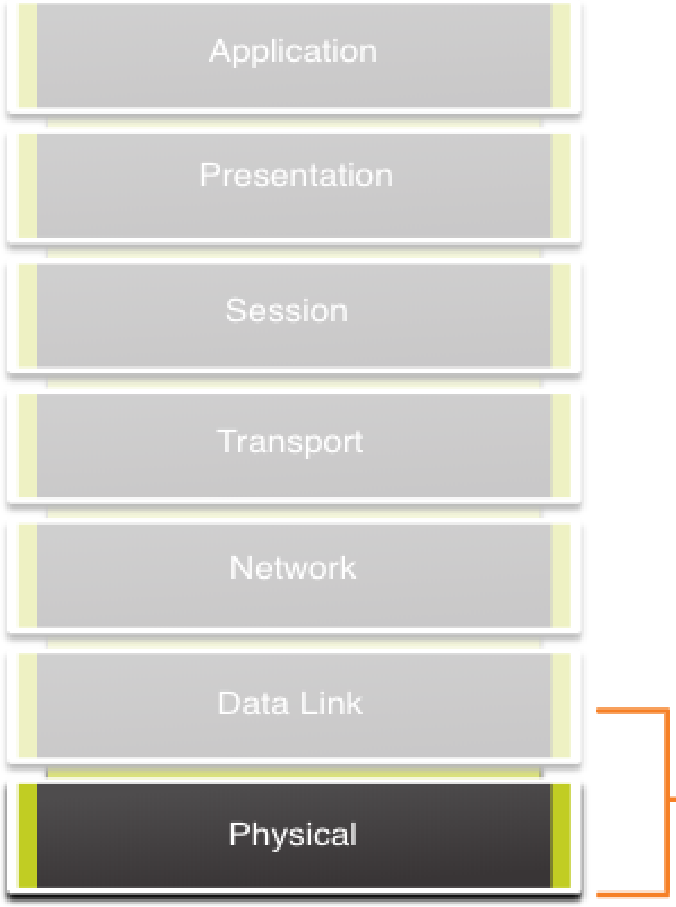
2025

Chapter Layout

- Introduction
- Signals
- Physical Layer duties
- Digital transmission
- Analog transmission
- Multiplexer and Hub

Introduction

Introduction



The physical layer standards are implemented in hardware and are governed by many organizations including:

- ISO
- EIA/TIA
- ITU-T
- ANSI
- IEEE

Introduction

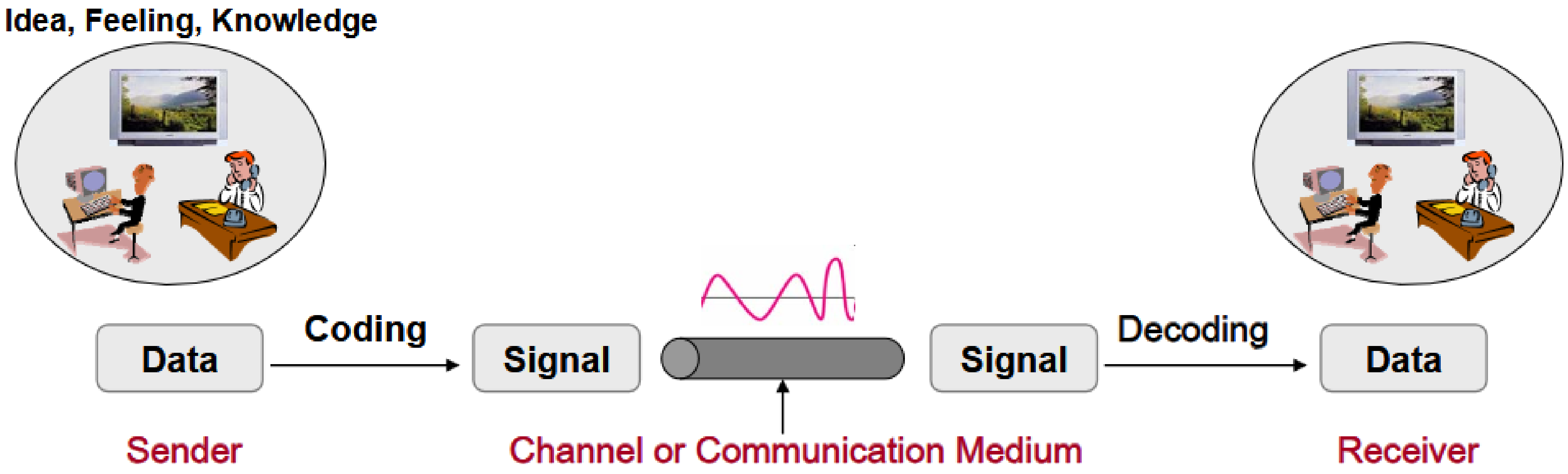
- Physical layer defines :
  - The hardware equipments
  - Cabling
  - Wiring
  - Frequencies
  - Pulses used to represent binary signals.

- PDU of a physical layer is the bit

Signals

Data vs Signal

- **Data** : information formatted in human/machine readable form
  - examples: voice, music, image, file
- **Signal** : electric or electromagnetic representation of data
  - transmission media work by conducting energy along a physical path; thus, to be transmitted, data must be turned into energy in the form of electro-magnetic signals
- **Transmission** : communication of data through propagation and processing of signals



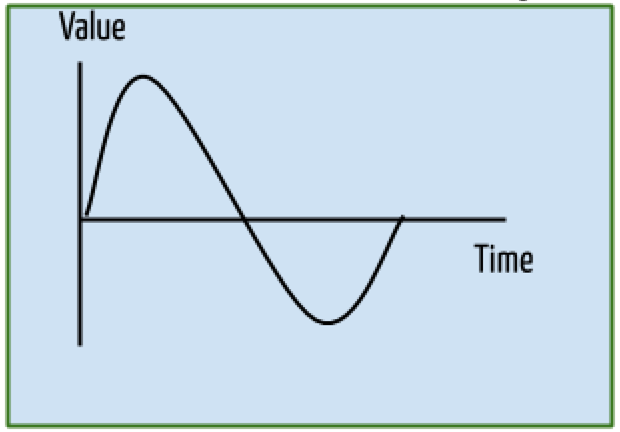
Analog Data vs Digital Data

- Both the data and the signal can be represented in form of analog and digital.
- **Data** can be
  - **Analog :** it is continuous data such as human voice,
  - **Digital :** it is represented in a binary format (0 and 1)such as files on the disk and softwares.

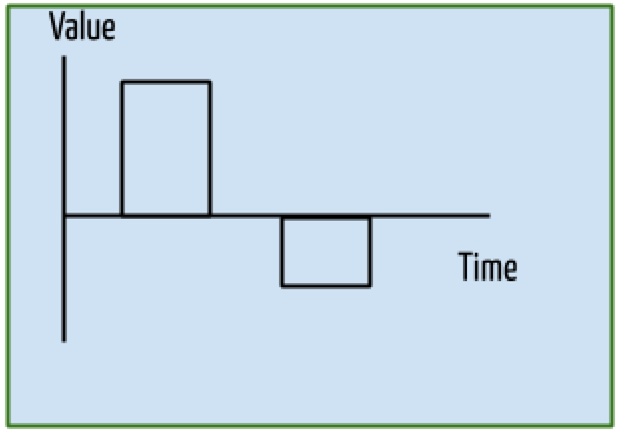
Analogue	Digital
Values change smoothly and gradually	Values are limited (1s and 0s)
Quality can be lost during storage/ transmission	Quality is maintained during storage/ transmission
Needs to be converted to digital to be used in digital devices	Needs to be converted to analogue to use in analogue devices
	Easily manipulated

Analog Signal vs Digital Signal

- When data is sent over physical medium, it needs to be first converted into electromagnetic signals.
- **Signals** can be :
  - **Digital :** are discrete (have limited number of values in a given range) in nature and represent sequence of voltage pulses.
  - **Analog :** are in continuous (can have infinite number of values in a given range) wave form in nature and represented by continuous electromagnetic waves.



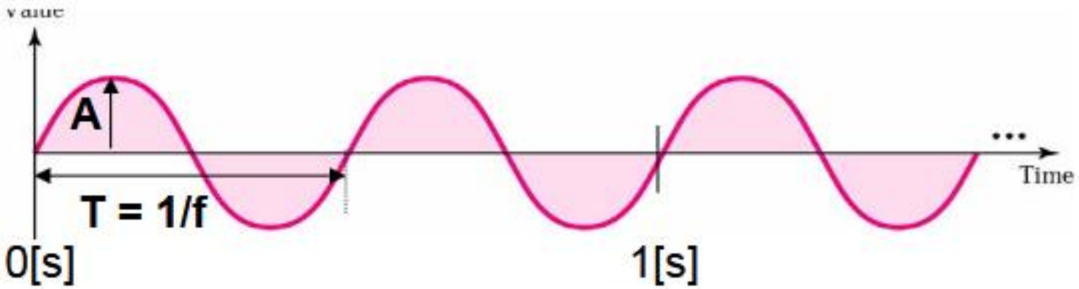
Analog Signal



Digital Signal

Analog signal – Sine wave

- sinewave is the most fundamental form of periodic analog signal
- Mathematically described with three parameters :
  - **Peak amplitude**: Absolute value of highest intensity of sine wave.
  - **Frequency and Period**: Period is the amount of time a signal takes to complete one cycle, it is denoted by T. Frequency refers to the number of cycles in 1 second, it is denoted by f. They are inversely proportional to each other which means f = 1/T.
  - **Phase**: Phase refers to the position of sine wave relative to the time 0.



Digital Signals

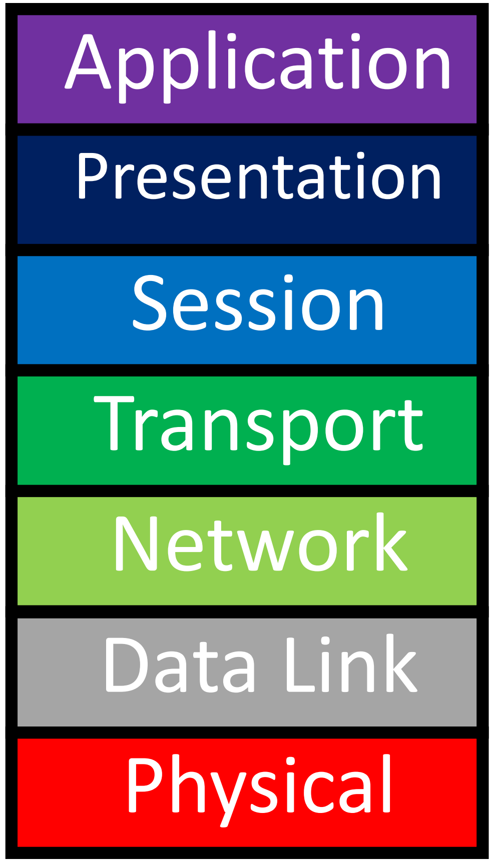
- Similar to analog signals, data can be transmitted in form of digital signals.
- For example data that is converted into a machine language (combination of 0s and 1s) such as 1001 can be represented in form digital signals. 1 represents high voltage and 0 represents low voltage.
- Some media that can transport digital signals directly:
  - Serial Communication Cable (ex : USB cable)
  - IR

Physical Layer duties

Physical Layer :
Key challenge

- Digital computers
  - 0s and 1s
- Analog world
  - Amplitudes and frequencies

Physical Layer :
Key challenge



- Function:
  - Get bits across a physical medium
- Key challenge:
  - How to represent bits in analog
  - Ideally, want high-bit rate
  - But, must avoid desynchronization

Physical Layer :
Purpose

- The **Physical Layer** (Layer 1) of the **OSI model** is the lowest layer.
- It is responsible for for transmitting **raw bits (0s and 1s)** over a physical medium.
- The physical layer does not understand frames, IP addresses, or MAC addresses — it only moves bits.
- It Accepts a complete frame from the Data Link Layer and encodes it as a series of signals that are transmitted to the local media

Key Duties (services) of the Physical layer

1. Bit transmission
2. Signal encoding and decoding
3. Physical medium specification
4. Data Rate
5. Bit synchronization
6. Physical topology
7. Transmission mode
8. Interface standard and media
9. Signal amplification and regeneration
10. Transmission techniques

Key Duties (services) of the Physical layer
1. Bit Transmission

- Sends signals over the medium (copper, fiber, wireless).
- Receives signals and converts them back into bits.
- It includes:
  - Sending bits
  - Receiving bits
  - Maintaining timing
  - Ensuring the bits arrive in order
- Example:
  - 10110010 → electrical pulses on a cable.

Key Duties (services) of the Physical layer

2. Signal Encoding

- The physical layer defines **how bits are represented** as signals.
- Because we cannot send raw 0s and 1s — we must convert them into:
  - Voltage levels (electrical)
  - Light pulses (fiber)
  - Radio waves (wireless)
- Encoding answers:
  - “What physical signal corresponds to each bit?”
- Examples:
  - **NRZ (Non-Return to Zero)**
  - **Manchester encoding**
  - **Bipolar AMI**

Key Duties (services) of the Physical layer
3. Physical medium specification

- It defines:
  - Voltage levels (e.g., 0V = 0, 5V = 1)
  - Light pulses (fiber optics)
  - Radio frequencies (Wi-Fi)
  - Signal timing
- Example:
  - In Ethernet over copper, voltage differences represent bits.

Key Duties (services) of the Physical layer

4. Data rate (Bit rate)

- Defines how many bits are transmitted per second.
- Measured in **bps (bits per second)**.

$$\textbf{Data Rate} = \frac{\text{Number of Bits}}{\text{Time (seconds)}}$$

$$\textbf{Bit Duration} = \frac{1}{\text{Data Rate}}$$

- Example:
  - 100 Mbps
  - 1 Gbps

Key Duties (services) of the Physical layer
5. Bit synchronization

- The Problem :
  - The sender and receiver each have their own clocks. If their clocks:
    - Drift
    - Are slightly faster/slower
    - Are not aligned

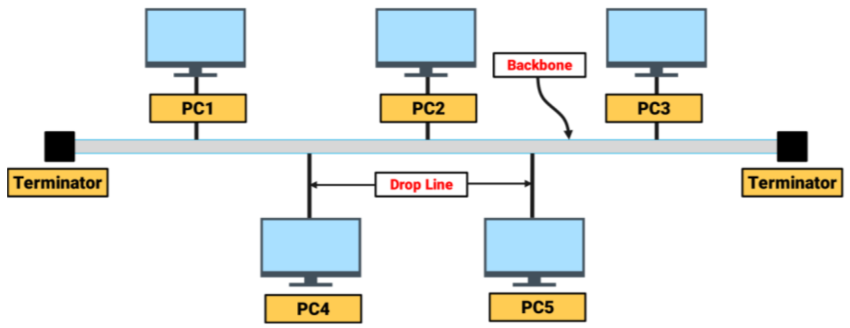
  → Bit errors occur.
- Solution : Bit Synchronization
- Bit synchronization is the mechanism that allows the receiver to know exactly when each bit starts and ends.
- Because in digital communication:
  - Bits are sent continuously
  - There are no spaces between bits
  - The receiver must know when to sample the signal
- If sender and receiver are not synchronized → bits are misinterpreted.

Key Duties (services) of the Physical layer

6. Physical topology

- Defines how devices are physically connected:
  - Bus
  - Star
  - Ring
  - Mesh

- Example

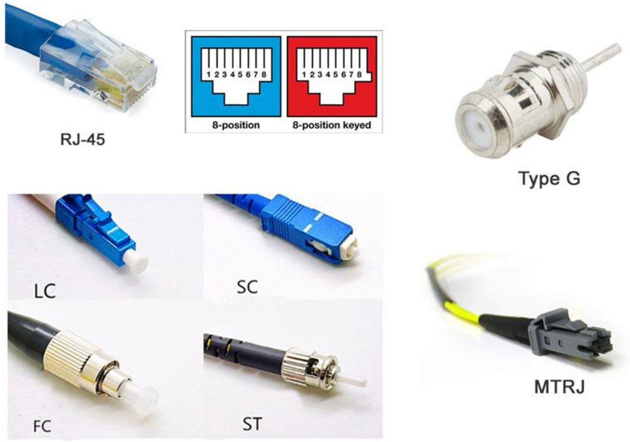


Key Duties (services) of the Physical layer
7. Transmission mode

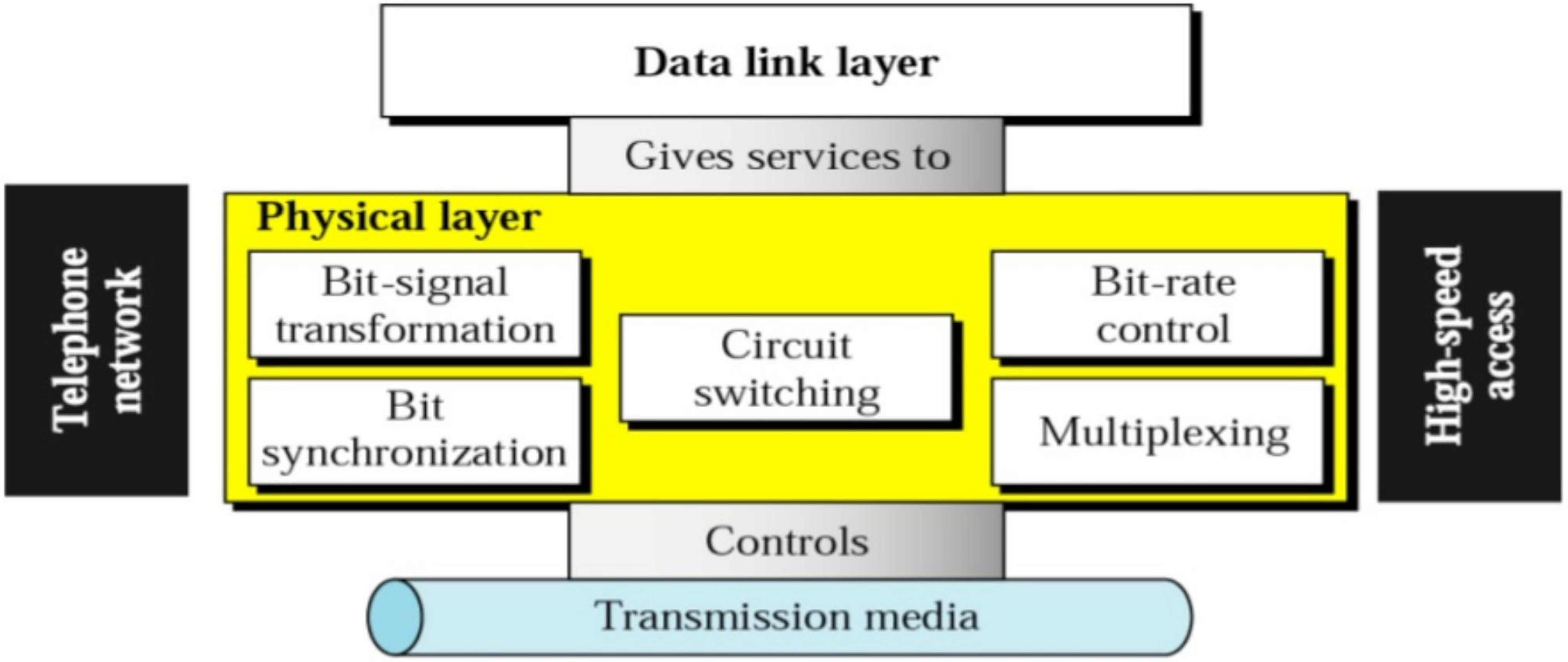
- Defines direction of data flow:
- **Simplex** → one direction
- **Half-duplex** → both directions, not simultaneously
- **Full-duplex** → both directions simultaneously

Key Duties (services) of the Physical layer
8. Interfaces and media

- It specifies:
  - Cable types (UTP, fiber, coaxial)
  - Connectors (RJ45)
  - Pin configurations



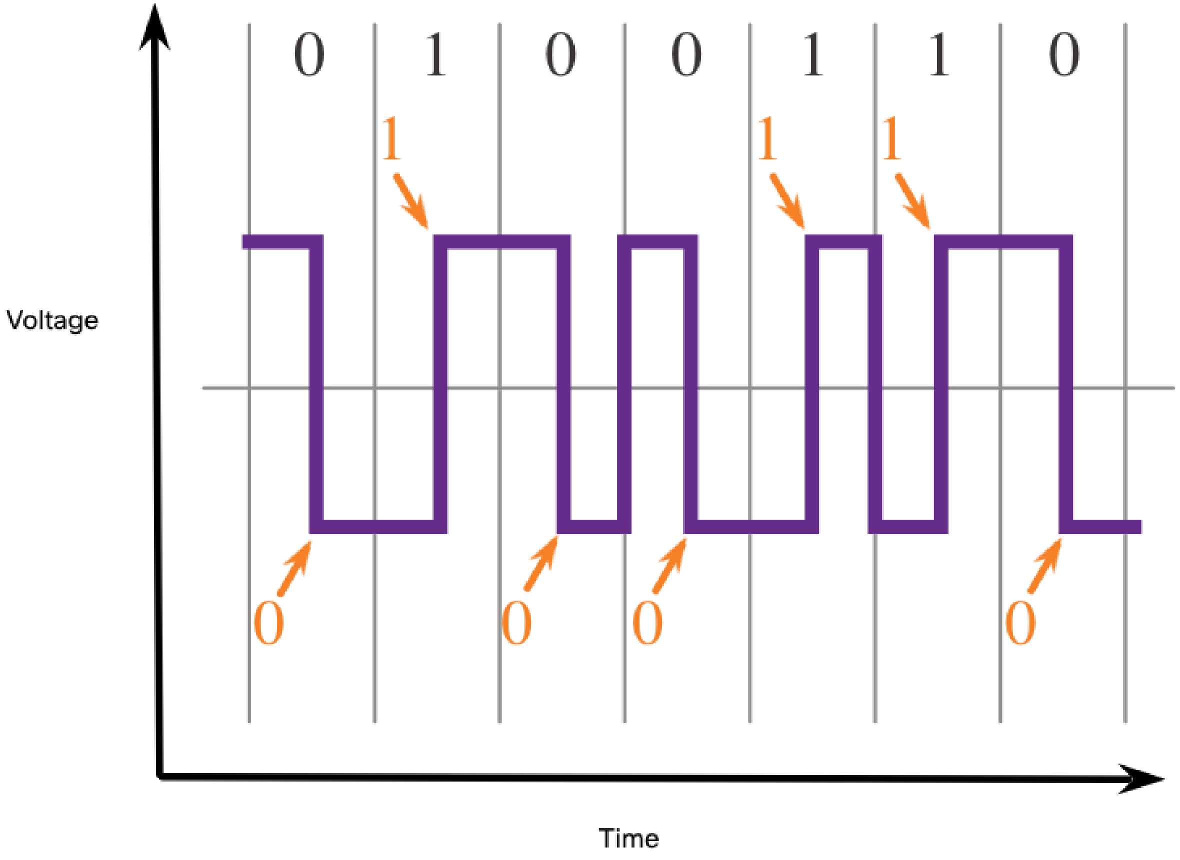
Physical layer position in OSI model



Signal Encoding

What is encoding ?

- Encoding refers to the process of **converting digital data (bits) into signals** that can be transmitted over a physical medium.
- Encoding ensures that the data can be accurately and efficiently transmitted, received, and interpreted by the sender and receiver.



Types of Encoding

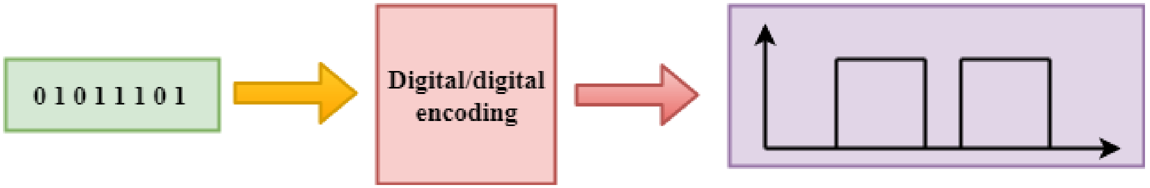
Depending on the type of signal used for transmission, encoding techniques can be broadly categorized into :

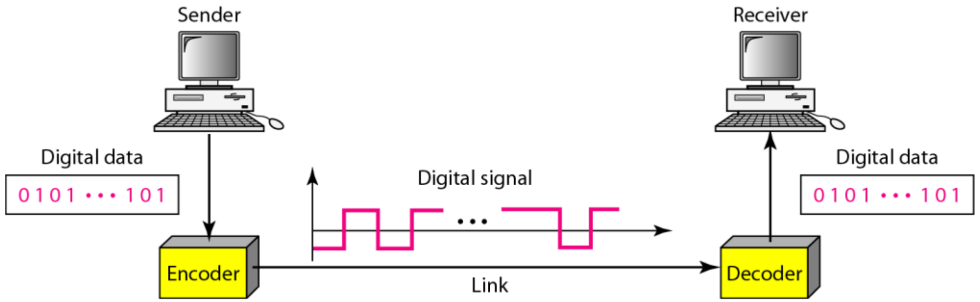
- **digital-to-digital encoding (Line coding)**
  - Converts digital data (bits) into digital signals for transmission over a digital medium (e.g., Ethernet cables, fiber optics).
- **digital-to-analog encoding (modulation)**
  - Converts digital data into analog signals for transmission over an analog medium (e.g., telephone lines, radio waves).

Line coding or
Digital transmission
(Digital data, Digital signal)

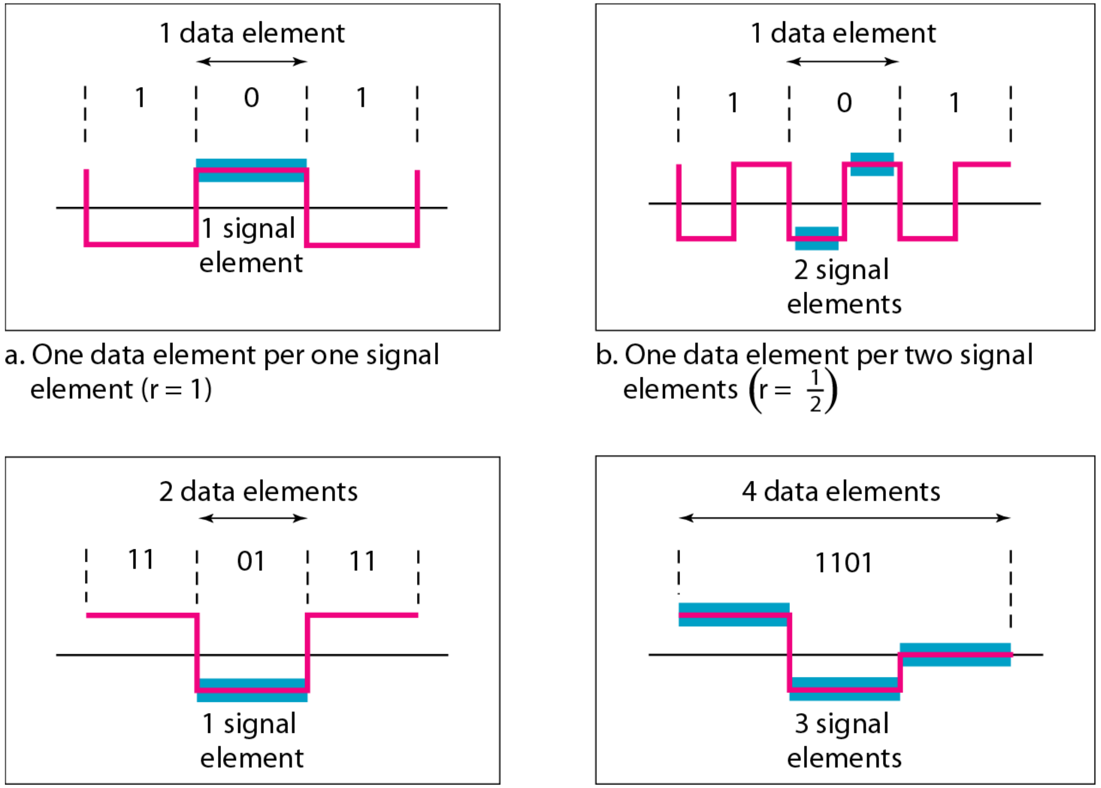
Digital transmission

- Physical layer needs to send these bits to the receiver side through **transmission media**.
- Digital-to-digital encoding is the representation of digital information by a digital signal.
- When binary 1s and 0s generated by the computer are translated into a sequence of voltage pulses that can be propagated over a wire, this process is known as digital-to-digital encoding.





Signal Element and Data Element Rate



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

- Signal rate S is the number of signal elements sent in 1s: baud.

  Signal rate = pulse rate, modulation rate, baud rate.
- Data rate N defines the number of data elements (bits) sent in 1s: bps.

  Data rate = bit rate.
- $S = c * N * 1/r$, where
  - S is the number of signal elements;
  - N is the data rate;
  - c is the case factor (worst, best, average);
  - r mapping ratio ‘r’ is the number of data elements ‘d’ carried by a signal element ‘s’ $r=d/s$.
- The actual bandwidth of a digital signal is infinite, but effective bandwidth is finite
- The bandwidth is proportional to the signal rate (baud rate)
- The minimum bandwidth: $B_{min} = c * N * 1/r$
- The maximum data rate: $N_{max} = 1/c * B * r$
- Goal is to increase the data rate N while reducing the baud rate S.

Example 1. Data Rate vs Baud Rate

- **Task**
- A signal is carrying data in which one data element is encoded as one signal element (r = d/s = 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 50 kbaud.

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

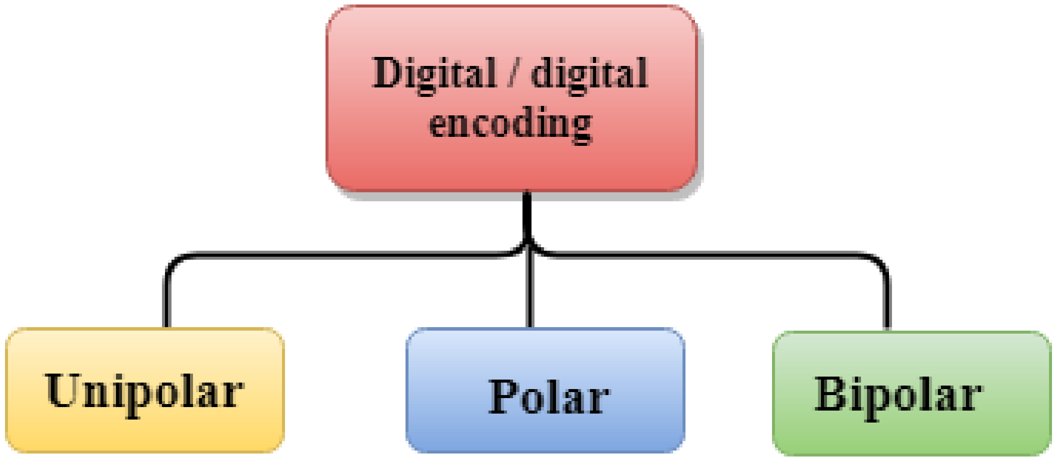
Digital To Digital Conversion

- For this conversion we will use three popular techniques:
    1. Line Coding (always needed)
    2. Block Coding
    3. Scrambling

- Line coding is always needed, block coding and scrambling may or may not be needed.

Line Coding

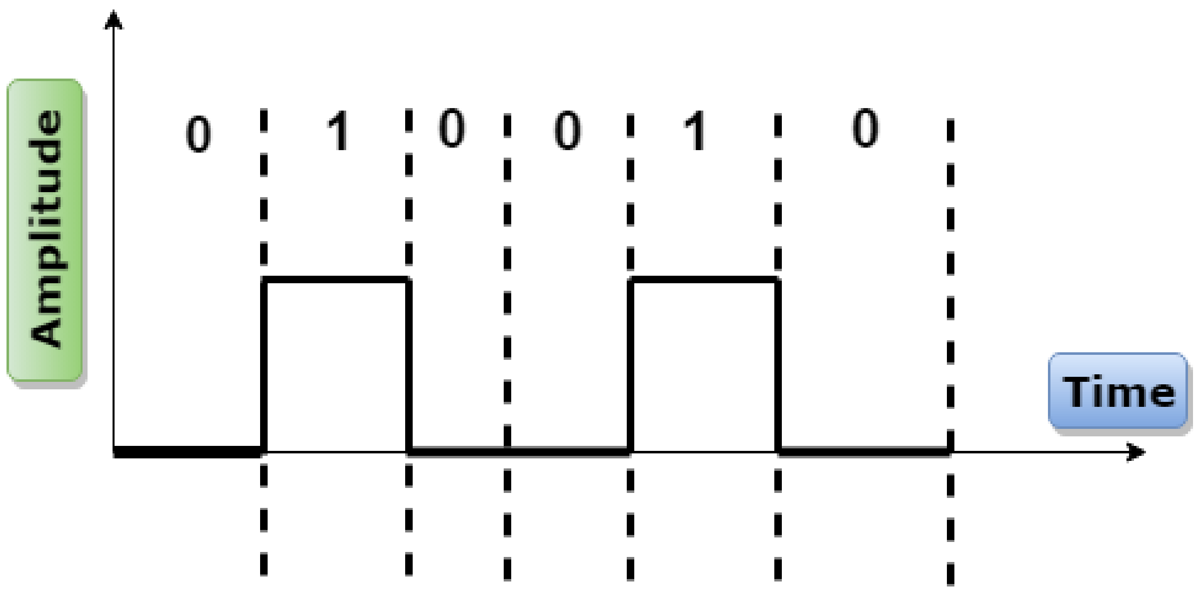
- The main purpose of this technique is to convert digital data to digital signal.
- Line Coding technique is further divided in the following categories:
  a) Unipolar
  b) Polar
  c) Bipolar



Unipolar Encoding

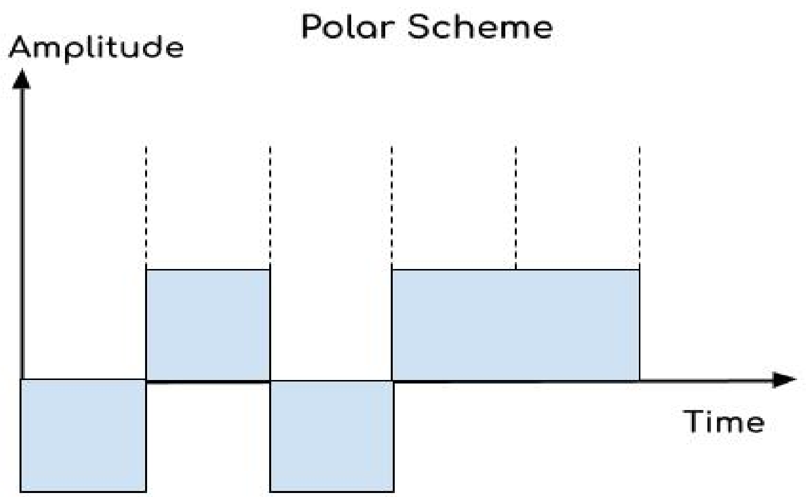
- In most types of encoding, one voltage level represents 0, and another voltage level represents 1.
- This type of encoding is known as Unipolar encoding as it uses **only one polarity**.
- In Unipolar encoding, **the polarity is assigned to the 1 binary state**.
- In this Scheme **“1” is considered as high** voltage while the **‘0” is considered low** voltage (zero).
- Due to its **heavy cost**, this scheme is not used in data communications nowadays.

Unipolar : example



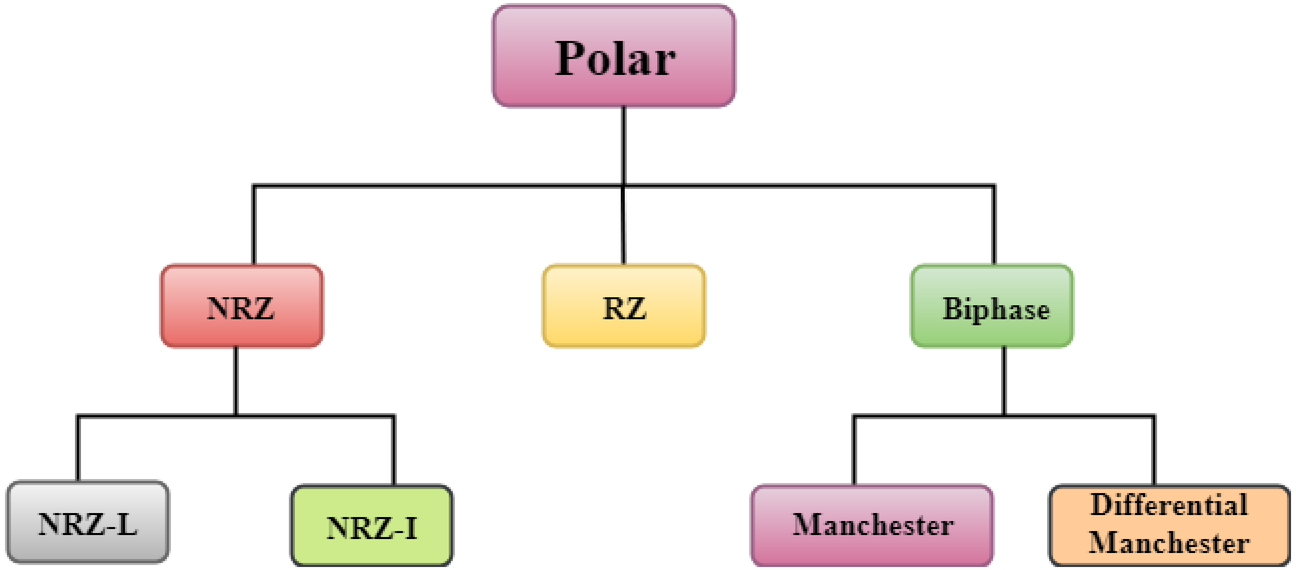
Polar Encoding

- Polar encoding is an encoding scheme that uses **two voltage levels**: one is **positive**, and another is **negative**.



Polar Encoding

- The polar Schemes are categorized as three types:
  - Non-Return-To-Zero (NRZ)
    - NRZ-L
    - NRZ-I
  - Return-To-Zero (RZ)
  - Biphase:
    - Manchester
    - Differential Manchester

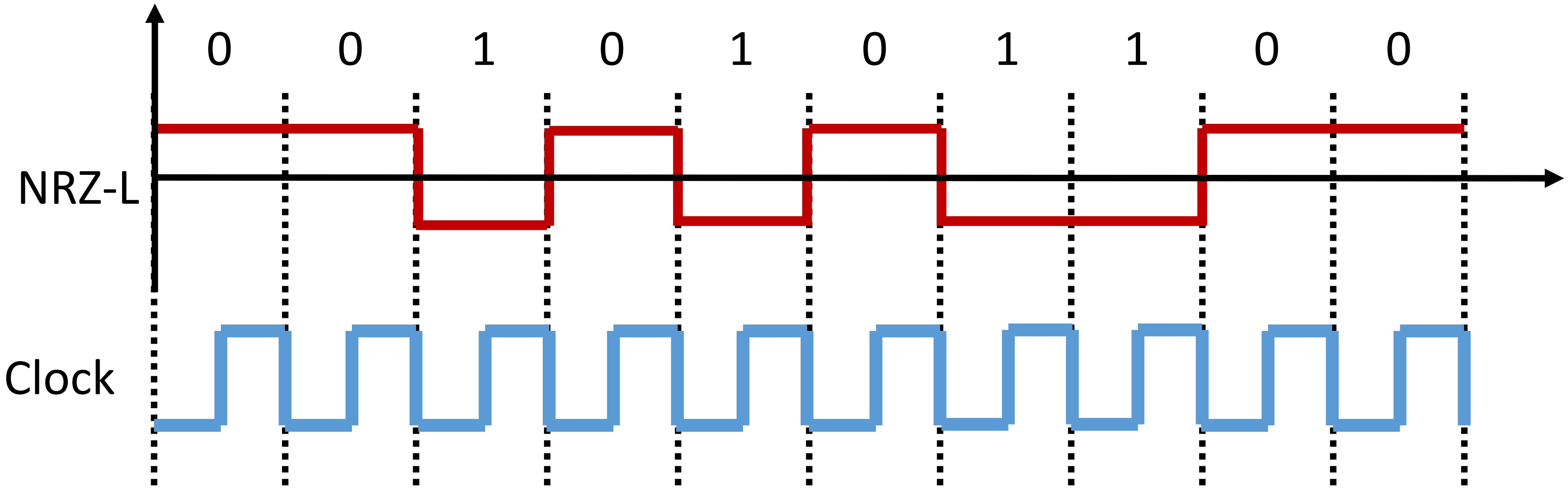


Non-Return to Zero (NRZ)

The two most common methods used in NRZ are:

- **NRZ-L (NRZ-Level):**
  - In NRZ-L encoding, the level of the signal depends on the type of the bit that it represents.
  - If a bit is 0 or 1, then their voltages will be positive and negative respectively (convention).
  - Therefore, we can say that the level of the signal is dependent on the state of the bit.
- **NRZ-I (NRZ-Invert):**
  - NRZ-I is an inversion of the voltage level that represents 1 bit (convention).
  - In the NRZ-I encoding scheme, a transition occurs between the positive and negative voltage that represents 1 bit.
  - In this scheme, 0 bit represents no change and 1 bit represents a change in voltage level.

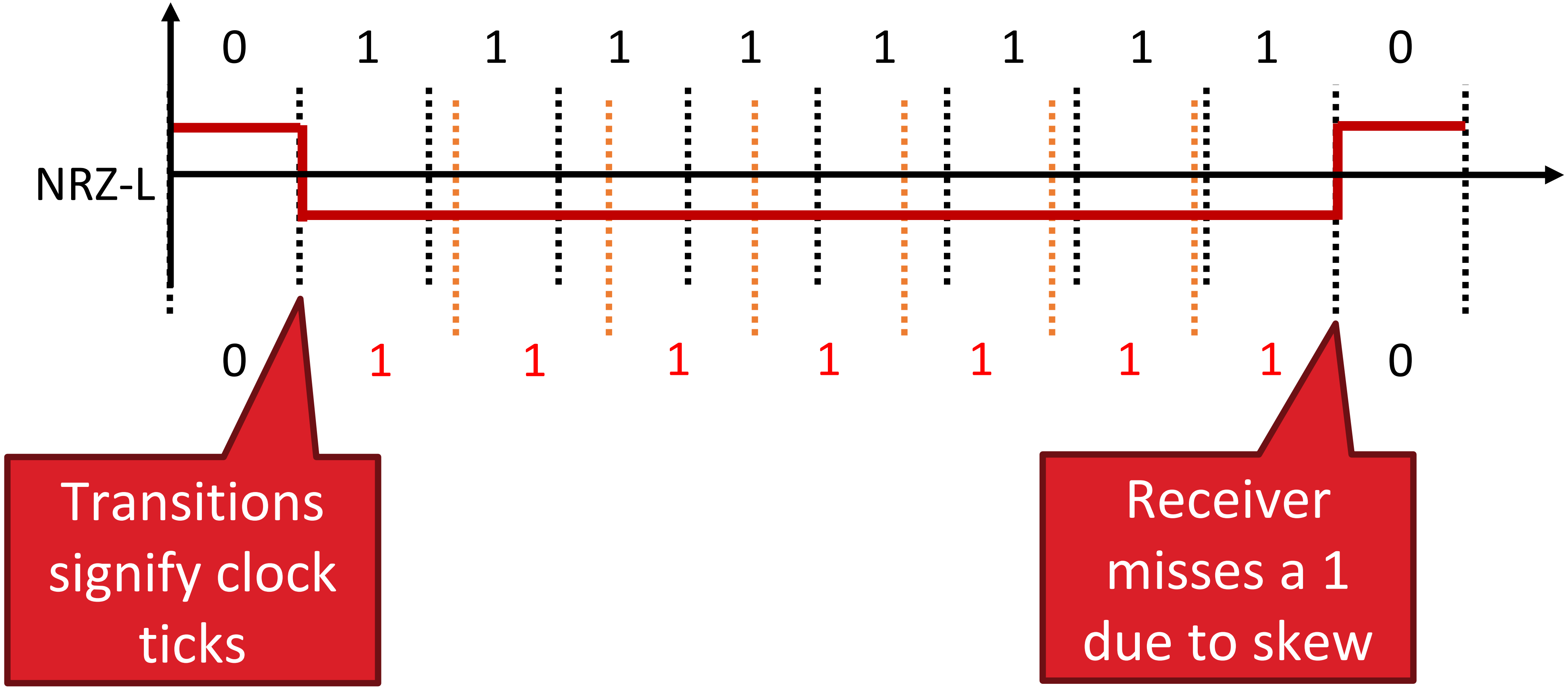
Non-Return to Zero (NRZ-L)



- Problem: long strings of 0 or 1 cause desynchronization
  - How to distinguish lots of 0s from no signal?
  - How to recover the clock during lots of 1s?

Desynchronization

- Problem: how to recover the clock during sequences of 0's or 1's?

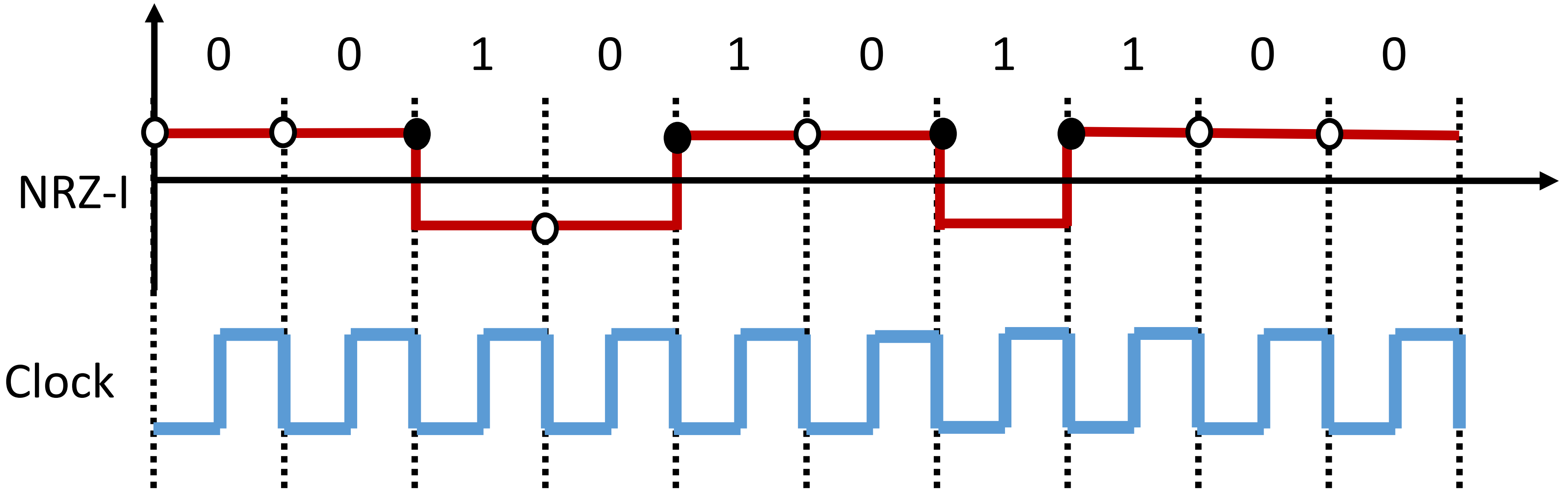


Clock Recovery

- When to sample voltage?
- Synchronized sender and receiver clocks
- Need easily detectible event at both ends
  - Signal transitions help resync sender and receiver
  - Need frequent transitions to prevent clock skew
  - SONET XOR's bit sequence to ensure frequent transitions

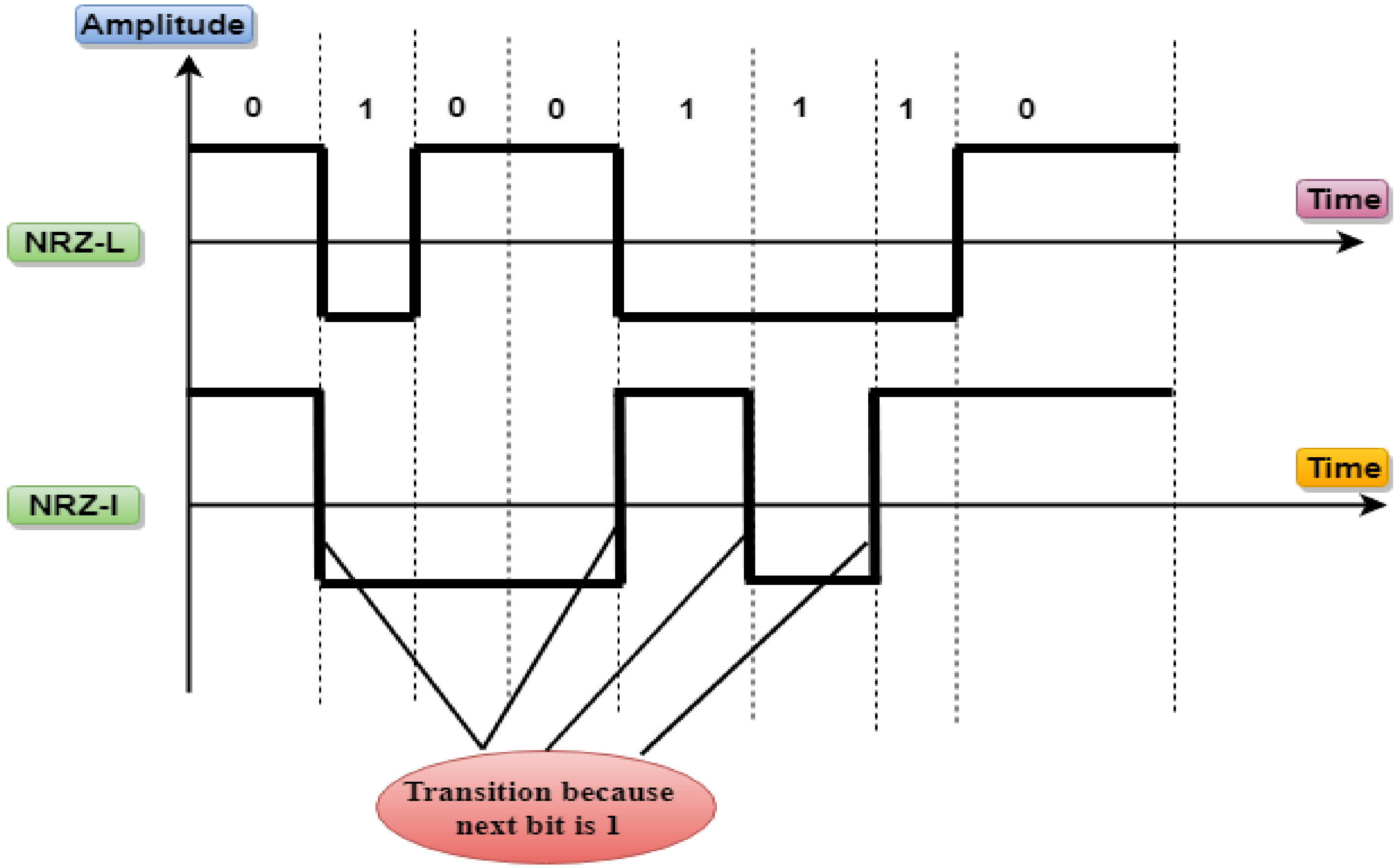
Non-Return to Zero Inverted (NRZ-I)

- 1 → make transition, 0 → remain the same (convention)



□ Solves the problem for sequences of 1s, but not 0s

NRZ-L vs NRZ-I


Amplitude
0
1
0
0
1
1
1
0
NRZ-L
Time
NRZ-I
Time
Transition because
next bit is 1

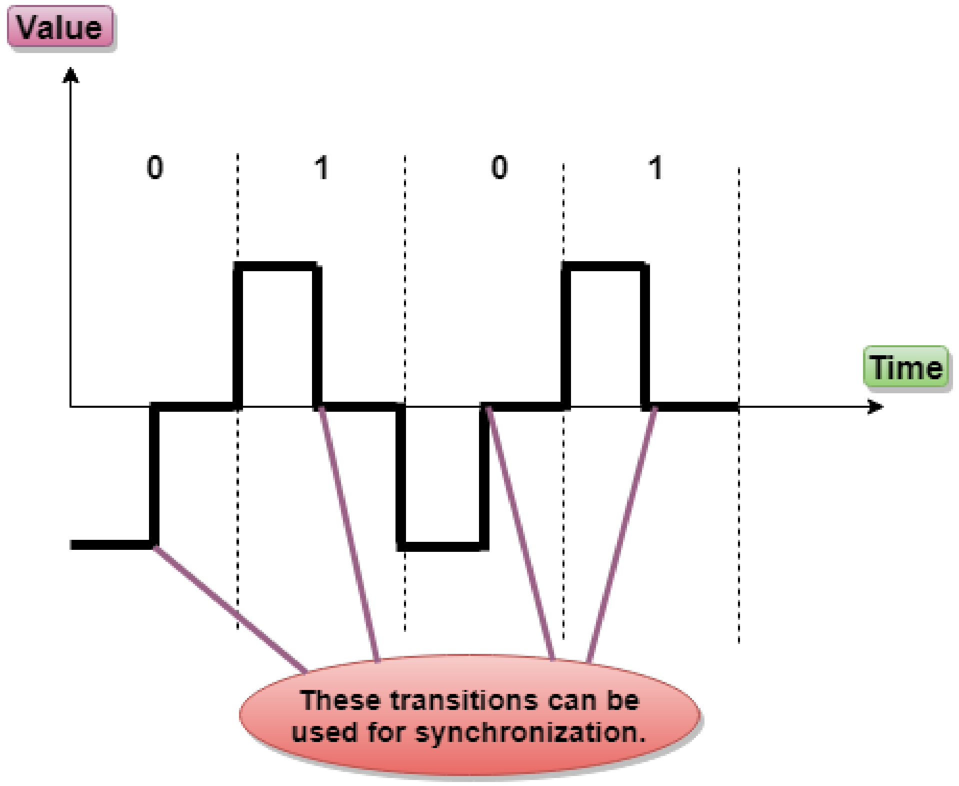
Return to Zero (RZ)

- There must be a signal change for each bit to achieve synchronization.
- The signal returns to zero voltage within each bit period
- Provides three values:
  - Positive voltage represents 1,
  - Negative voltage represents 0,
  - and zero voltage represents none.
- Halfway through each interval, the signal returns to zero.
- The bit 1 is represented by positive-to-zero pulse and 0 bit is represented by negative-to-zero pulse.
- Provides clear bit boundaries but uses more bandwidth compared to NRZ schemes.

Return to Zero (RZ)

Disadvantage of RZ:

- It performs two signal changes to encode one bit that acquires more bandwidth.



Biphase coding

- Biphase is a line encoding scheme in which signal changes at the middle of the bit interval but does not return to zero.
- That is why it is called **bi-phase** (two signal phases per bit).
- There is **always a transition at the center of every bit** → This provides built-in synchronization.
- Biphase encoding is implemented in two different ways:
  - **Manchester**
  - **Differential Manchester**

Manchester

- Combines RZ and transition signaling.

- In the middle of each bit duration, there's a transition (either high-to-low or low-to-high).

- Rules (one common convention):
  - 0 → High to Low transition (in middle)
  - 1 → Low to High transition (in middle)

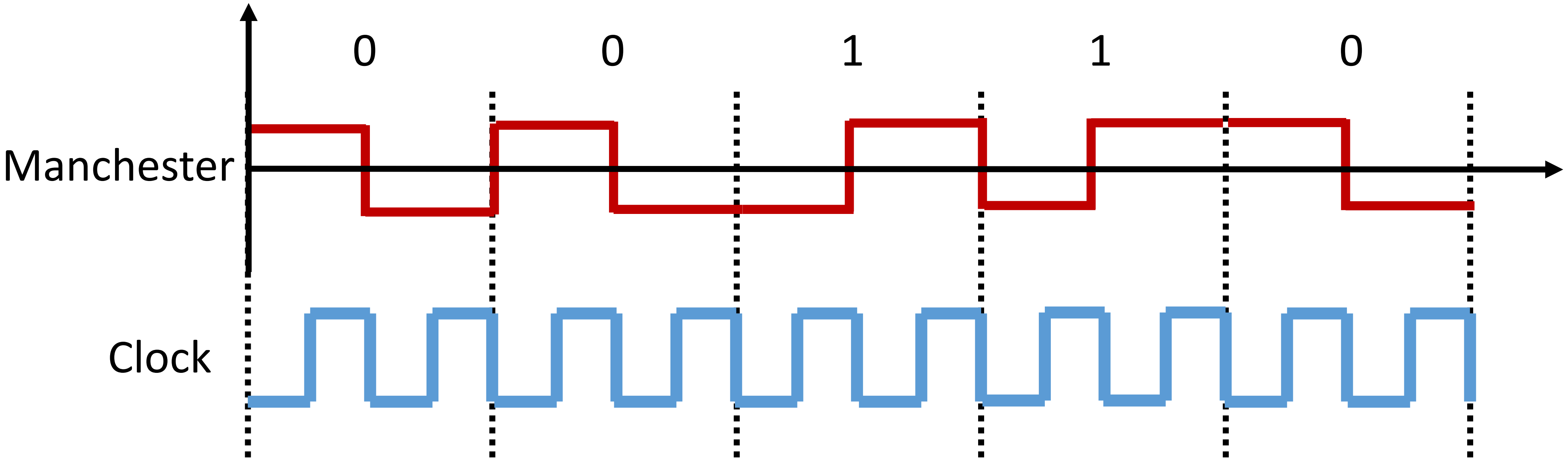
- Advantage:
  - Self-synchronizing

Why?

- Disadvantage:
  - Requires double bandwidth compared to NRZ
  - **More transitions → more frequency → more bandwidth**

Manchester

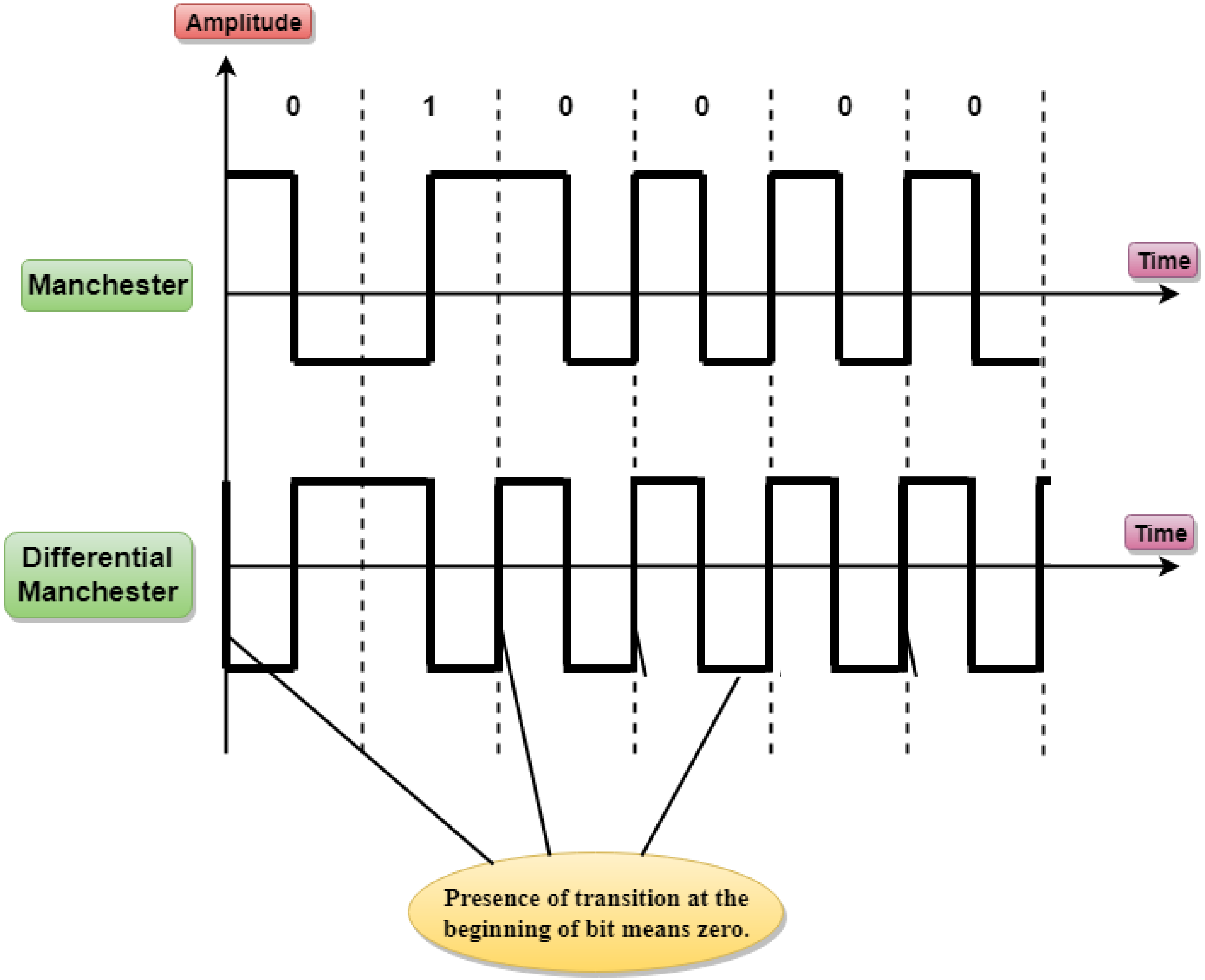
- 0 → high-to-low, 1 → low-to-high (IEEE convention used in Ethernet)



- Used by Ethernet
- Good: Solves clock skew (every bit is a transition)
- Bad: Halves throughput (two clock cycles per bit)

Differential Manchester

- Rules:
  - There is always a transition in the middle (clocking)
  - Data is determined by presence/absence of transition at the beginning of the bit interval
- Convention (IEEE convention) :
  - Bit 0 → A transition at the beginning of the bit period
  - Bit 1 → No transition at the beginning of the bit period.
- In other words:
  - Two signal changes represent 0
  - One signal change represent 1.


Amplitude
0
1
0
0
0
0
Manchester
Time
Differential Manchester
Time
Presence of transition at the beginning of bit means zero.

Bipolar coding

- In the Bipolar coding scheme, we have three voltage levels:
  - zero,
  - positive,
  - and negative.
- Here either 0 or 1 is represented by neutral voltage on time axis and the other binary value is represented by negative and positive voltages on time axis.

Block coding

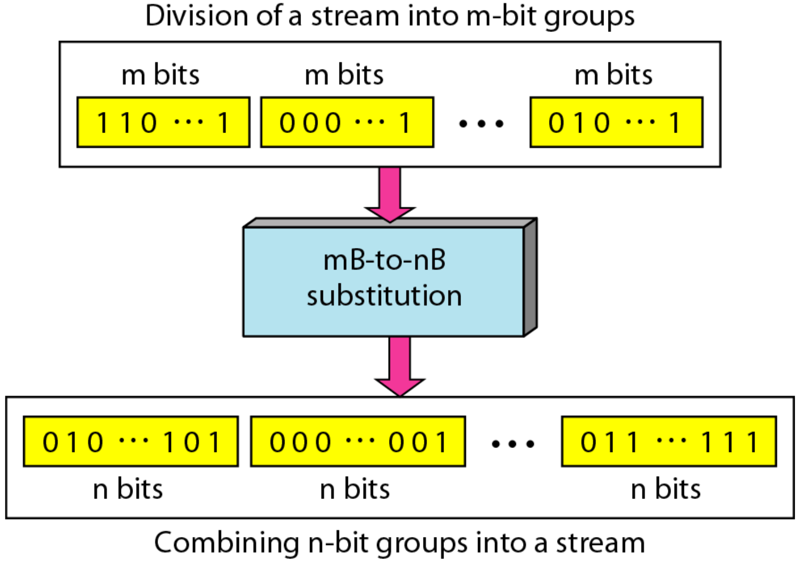
- Some line encoding schemes have synchronization issues (Ex: NRZ).
- → Solution : Block coding ensures enough transitions before applying Line encoding.

  Data → Block Coding → Line Encoding → Transmission

- Block coding is a technique that converts a group of data bits (m bits) into a larger group of bits (n bits), where n > m. → referred to as mB/nB coding
- It adds **redundancy** before line encoding which helps in synchronization at receiver's and sender's end and also providing some kind of error detecting capability.
- It normally involves three steps: division, substitution, and combination.

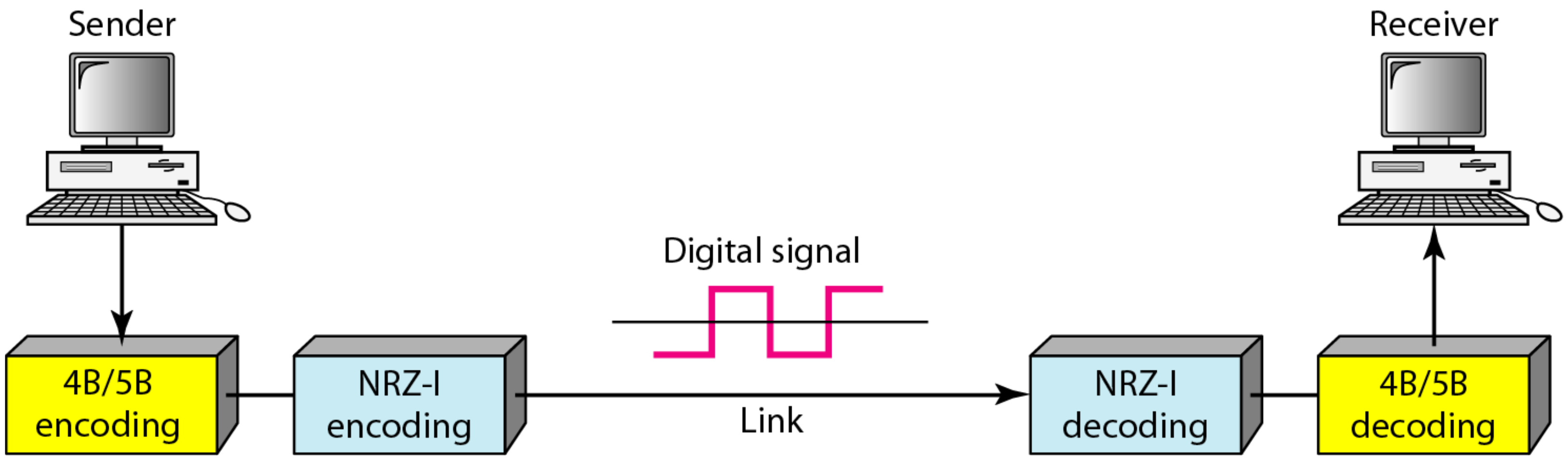
Block coding

- In the division step, a sequence of bits is divided into groups of m-bits.
- In the substitution step, we substitute an m-bit group for an n-bit group.
- Finally, the n-bit groups are combined together to form a stream which has more bits than the original bits.
- Examples of mB/nB coding: **4B/5B (four binary/five binary)**



4-bit/5-bit

- Used by FDDI
- Uses 5bits to encode every 4bits
- Encoding ensures no more than 3 consecutive 0's
- Uses NRZI to encode resulting sequence
- 16 data values, 3 "special" illegal values, 6 "extra" values, 7 illegal values



4-bit/5-bit (100 Mbps Ethernet)

8-bit / 10-bit used in Gigabit Ethernet

- Observat
- Idea: enc no more than one leading 0 and two trailing 0

4-bit	5-bit
0000	11110
0001	01001
0010	10100
0011	10101
0100	01010
0101	01011
0110	01110
0111	01111

4-bit	5-bit
1000	10010
1001	10011
1010	10110
1011	10111
1100	11010
1101	11011
1110	11100
1111	11101

- Tradeoff: efficiency drops to 80%

Modulation
Analog transmission
(Digital Data, Analog Signals)

Modulation

- Modulation is a signal encoding technique
- Used in Analog transmission
  - → Converting digital data into analog signal.
  - Requires the use of a modem.
- How ?
  - By changing a property of a high-frequency carrier wave according to the digital data being transmitted.
  - Steps
    1. Generate a high-frequency carrier wave.
    2. Modify it according to the data.
- **What Is a Carrier Wave?**
  - A carrier is a continuous sinusoidal signal: $c(t) = A * \sin(2*\pi*f*t)$

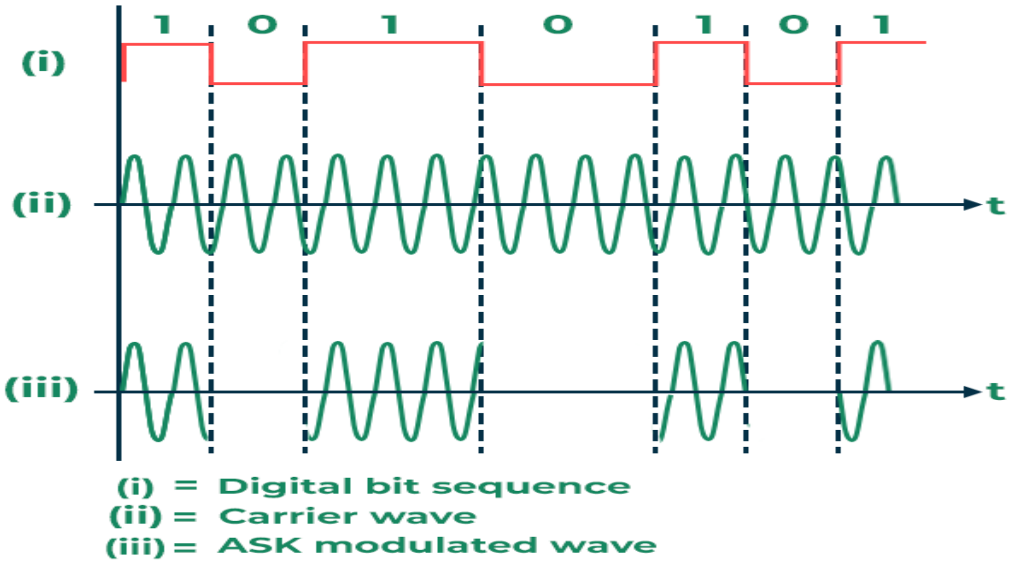
A : amplitude
f = frequency
t = time

Modulation

- **What Can We Modify?**
- There are three main properties:
  - Amplitude → Amplitude-shift keying (ASK)
  - Frequency → Frequency-shift keying (FSK)
  - Phase → Phase-shift keying (PSK)
  - Combination

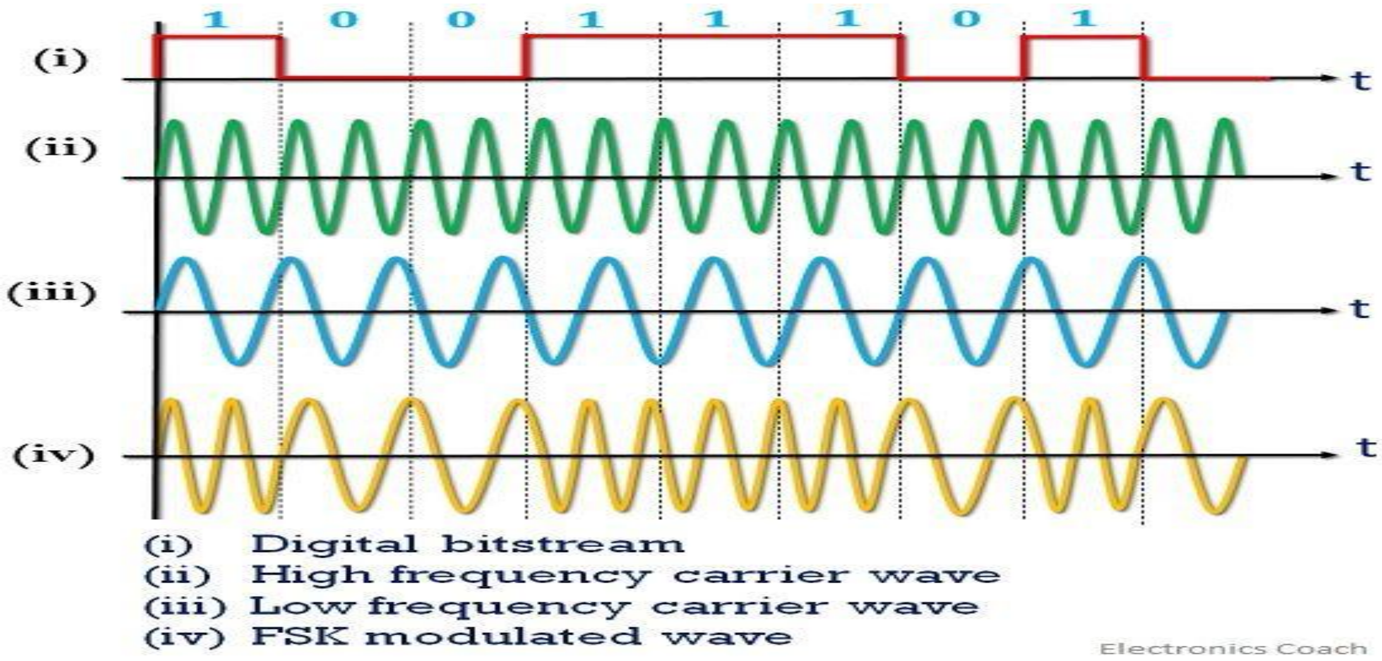
Amplitude-Shift Keying

- vary amplitude of the signal (called carrier)
- commonly, one of the amplitudes is zero
  - (ex : Bit 1 → High amplitude   Bit 0 → Low amplitude)



Frequency-Shift Keying

- vary frequency of the signal (called carrier)
- used in high-frequency radio transmission, among others

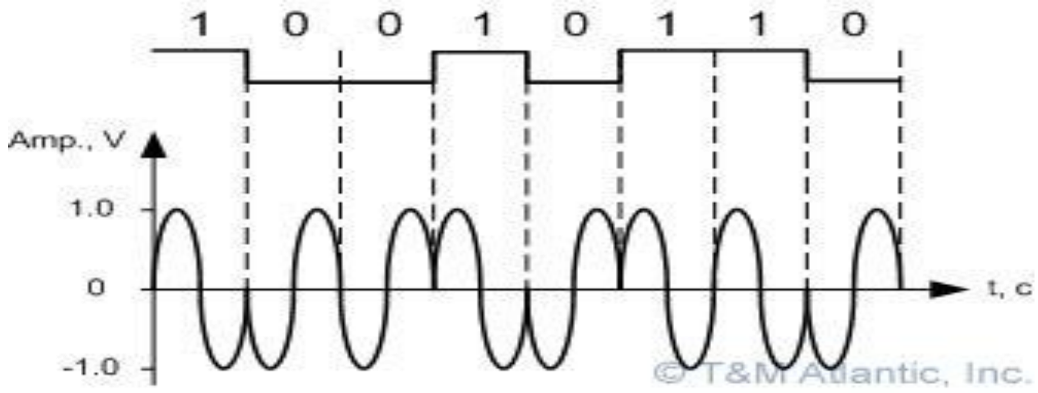


Phase-Shift Keying

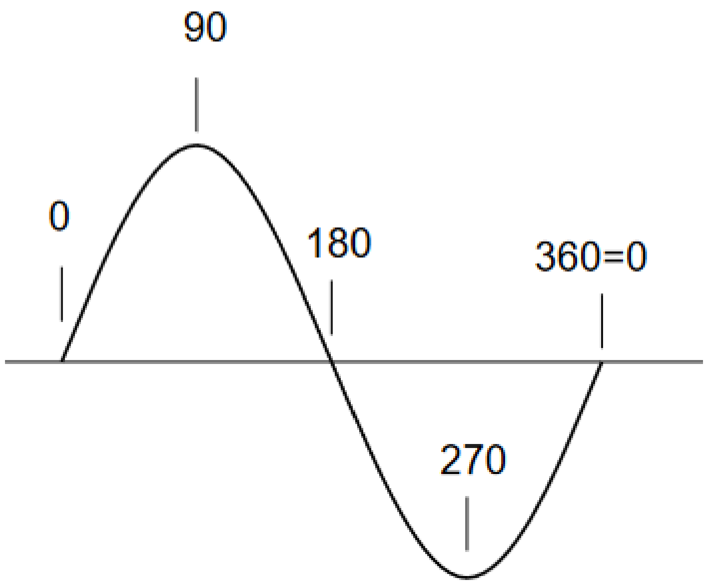
- vary phase of the signal
- a simple PSK scheme works like this:

The bit 0 has the phase 0 degree;

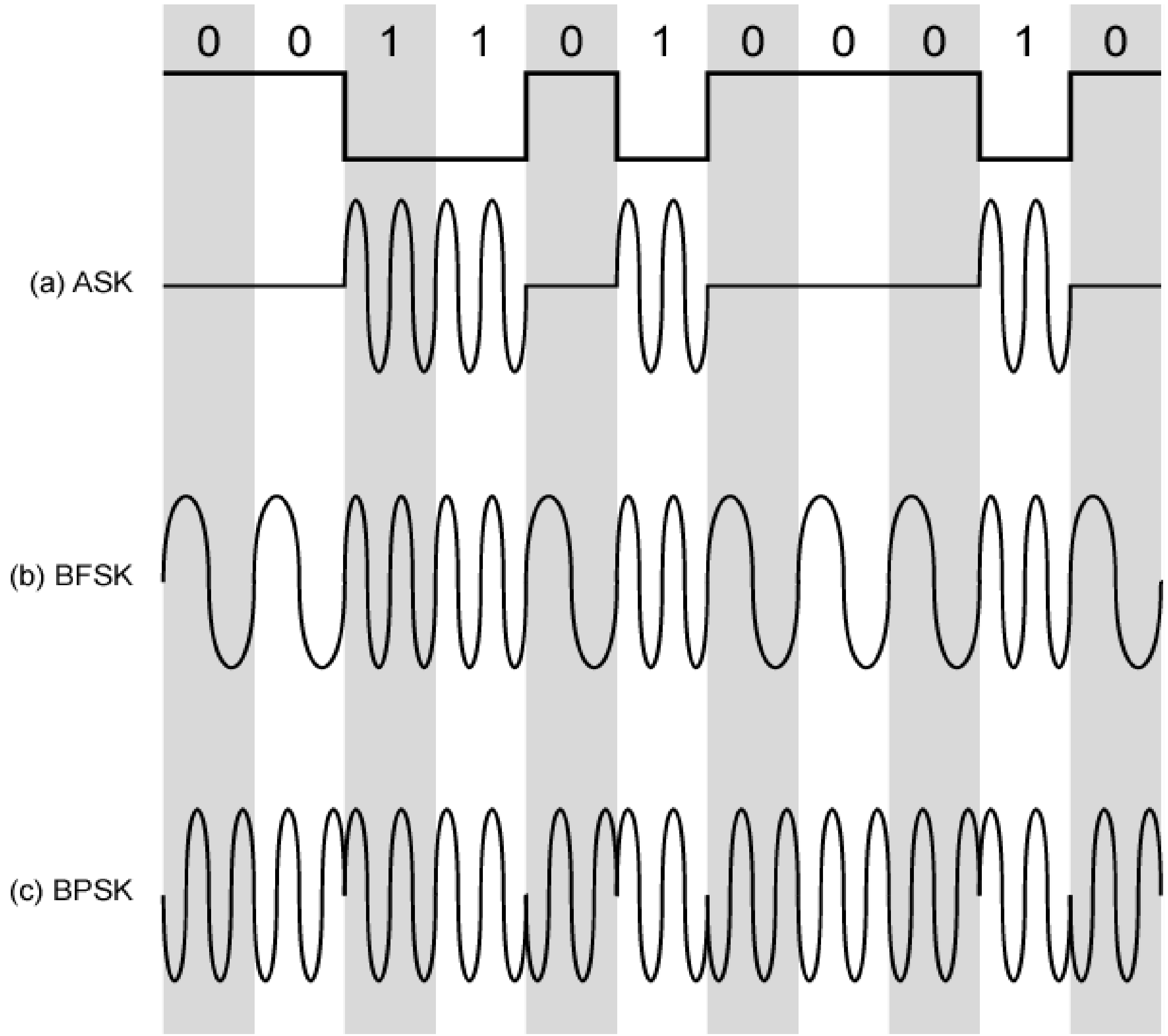
The bit 1 has the phase 180 degrees



Phase shift keying (PSK)



Modulation Techniques



Quadrature Amplitude Modulation (QAM)

- QAM (a combination of ASK and PSK) is used in ADSL and cable modems.

Example:

QAM-16 = QPSK and QASK

Idea - Increase the number of bits transmitted by increasing the number of levels used per symbol.

Example:

$R_{QAM\text{-}64} = 6\ R_{ASK}$

Quadrature Amplitude Modulation

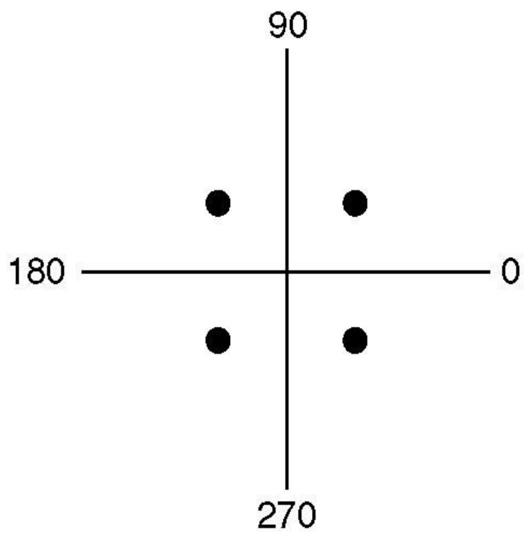
- A method of combining of amplitude-shift and phase-shift keying
- Example: to encode 3 bits per signaling element

Bit Combination	Phase Shift	Amplitude
000	0	low
001	0	high
010	90	low
011	90	high
100	180	low
101	180	high
110	270	low
111	270	high

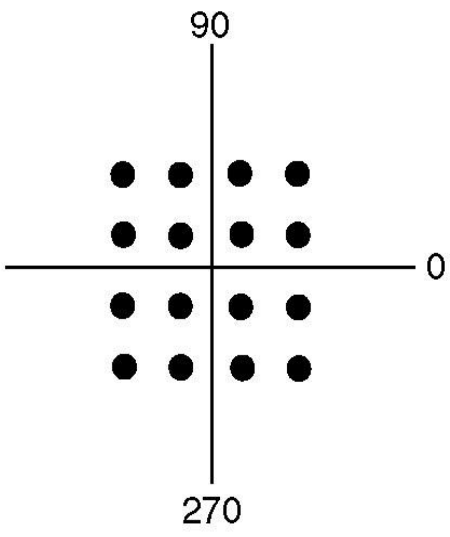
Modems

- All advanced modems use a ***combination of modulation techniques*** to transmit multiple bits per baud.
- Multiple amplitude and multiple phase shifts are combined to transmit several bits per symbol.
- **QPSK (Quadrature Phase Shift Keying)** uses four phase shifts per symbol.
- **Modems** actually use **Quadrature Amplitude Modulation (QAM).**
- These concepts are depicted using constellation points where a point determines a specific amplitude and phase.

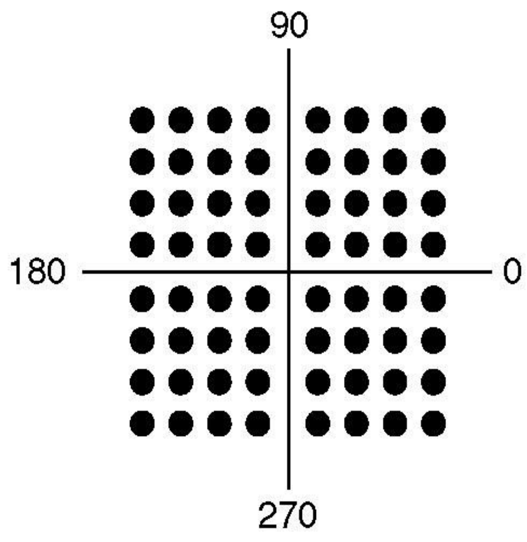
Constellation Diagrams



(a) QPSK.



(b) QAM-16.



(c) QAM-64.

$V = 64$
$v = \log_2 V = 6$

Data rate

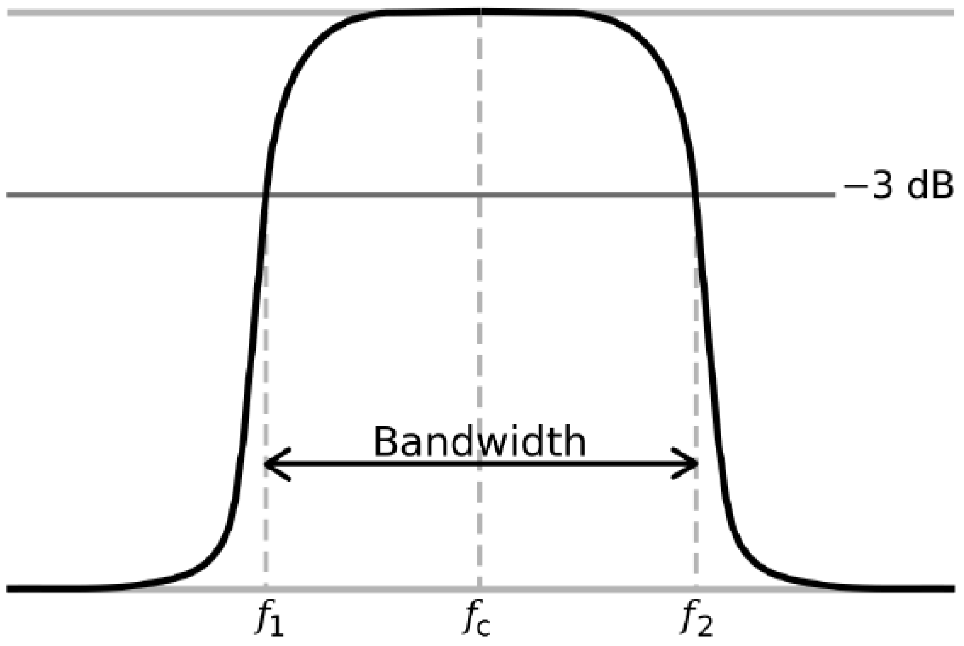
Some definitions

- The time required to transmit a character depends on both the **encoding method** and the **signaling speed** (i.e., the **modulation rate** - the number of times/sec the signal changes its voltage).
- **Hertz (Hz)** : is the number of periods (oscillations) per second
- **Baud (D)** – is the number of signal changes per second.
- **Data rate or channel capacity (C)** – is the no of bits per second.
- **Bandwidth (B)** - It defines the band of frequencies where signals are received correctly. The transmitted signal is constrained by the transmitter and the nature of the transmission medium in cycles/sec (hertz).

$$B = F_{max} - F_{min}$$

Bandwidth

- Every analog ***signal*** has a given ***bandwidth***



Bandwidth (in bps)

- Bandwidth is the capacity at which a medium can carry data.
- Digital bandwidth measures the amount of data that can flow from one place to another in a given amount of time; how many bits can be transmitted in a second.

Unit of Bandwidth	Abbreviation	Equivalence
Bits per second	bps	1 bps = fundamental unit of bandwidth
Kilobits per second	Kbps	1 Kbps = 1,000 bps = 10^3 bps
Megabits per second	Mbps	1 Mbps = 1,000,000 bps = 10^6 bps
Gigabits per second	Gbps	1 Gbps – 1,000,000,000 bps = 10^9 bps
Terabits per second	Tbps	1 Tbps = 1,000,000,000,000 bps = 10^{12} bps

Bandwidth Terminology

Latency

- Amount of time, including delays, for data to travel from one given point to another

Throughput

- The measure of the transfer of bits across the media over a given period of time

Goodput

- The measure of usable data transferred over a given period of time
- Goodput = Throughput - traffic overhead

Data Rate Limits

- How fast we can send data, in bits per second, over a channel.

- Two theoretical formulas were developed to calculate the data rate:
  - Nyquist for a noiseless channel
  - Shannon for a noisy channel.

Nyquist Theorem (assuming Noiseless channel)

- If an arbitrary signal is run through a low-pass filter of bandwidth **B**, the filtered signal can be *completely* reconstructed by making **2B** samples/sec.

- This implies for a signal of **L** discrete levels,

  Max. bit rate : C = 2 B log(L) (ignoring noise)

$$\text{BitRate} = 2 \times \text{bandwidth} \times \log_2 L$$

Examples

- Example 1.

  Consider a noiseless channel with a bandwidth of 3000 Hz transmitting a signal with two signal levels. What is the maximum bit rate?

$$\text{BitRate} = 2 \times 3000 \times \log_2 2 = 6000 \text{ bps}$$

- Example 2. {noiseless capacity}

  D = 2400 baud {note D = 2B}

  V = each pulse encodes 16 levels

  C = $2B \log_2 (V) = D \times \log_2 (V)$

  = 2400×4 = 9600 bps.

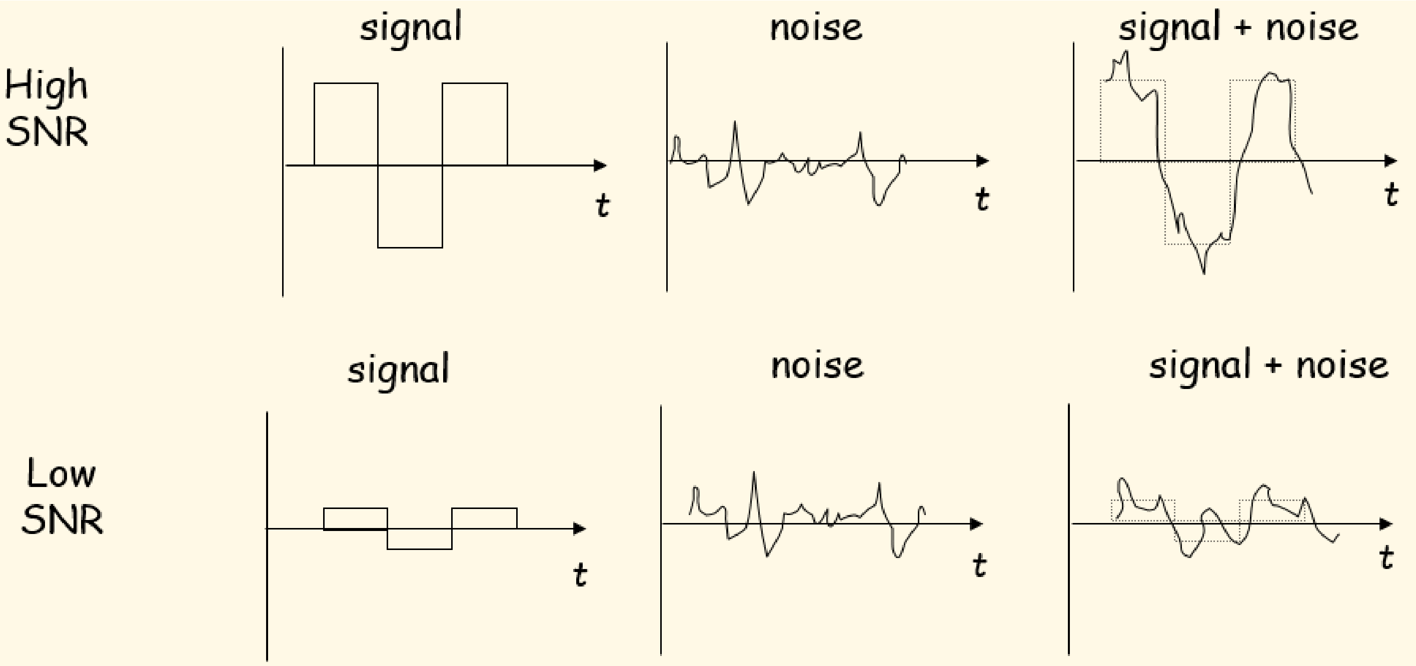
Shannon Capacity

- In reality, we can not have a noisless channel. In 1944, Claude Shannon introduced a formula, called the Shannon capacity, to determine the theoretical highest data rate for a noisy channel:

$$\text{Capacity} = \text{Bandwith} \times \log_2(1+\text{SNR})$$

- The Shannon capacity gives us the upper limit;
- the Nyquist formula tells us how many signal levels we need.

Signal to Noise Ratio (SNR)



$$\text{SNR} = \frac{\text{Average Signal Power}}{\text{Average Noise Power}}$$

$$\boxed{\text{SNR (dB)} = 10 \log_{10} \text{SNR}}$$

Leon-Garcia & Widjaja:
Communication Networks

Shannon's Channel Capacity Result

- For a noisy channel of bandwidth H Hz. and a signal-to-noise ratio SNR, the max. data rate::

$$C = H \log 2 (1 + SNR)$$

- Regardless of the number of signal levels used and the frequency of the sampling.

Shannon Example – Noisy Channel

Telephone channel (3400 Hz) at 40 dB SNR

$C = H \log_2 (1+SNR)$ b/s

SNR (dB) = 40 dB

40 dB $= 10 \log_{10} (SNR)$

$4 = \log_{10} (SNR)$

SNR =10,000

$C = 3400 \log_2 (10001)$ = 44.8 kbps

Analog and Digital Signaling

- Digital data can be represented by analog signals using a modem (modulator/demodulator).

  The digital data is encoded on a carrier frequency.

- Analog data can be represented by digital signals using a codec (coder-decoder).

Analog vs Digital Signaling

Digital signaling is:

- Cheaper
- Less susceptible to noise interference
- Suffers more attenuation.

Transmission mode

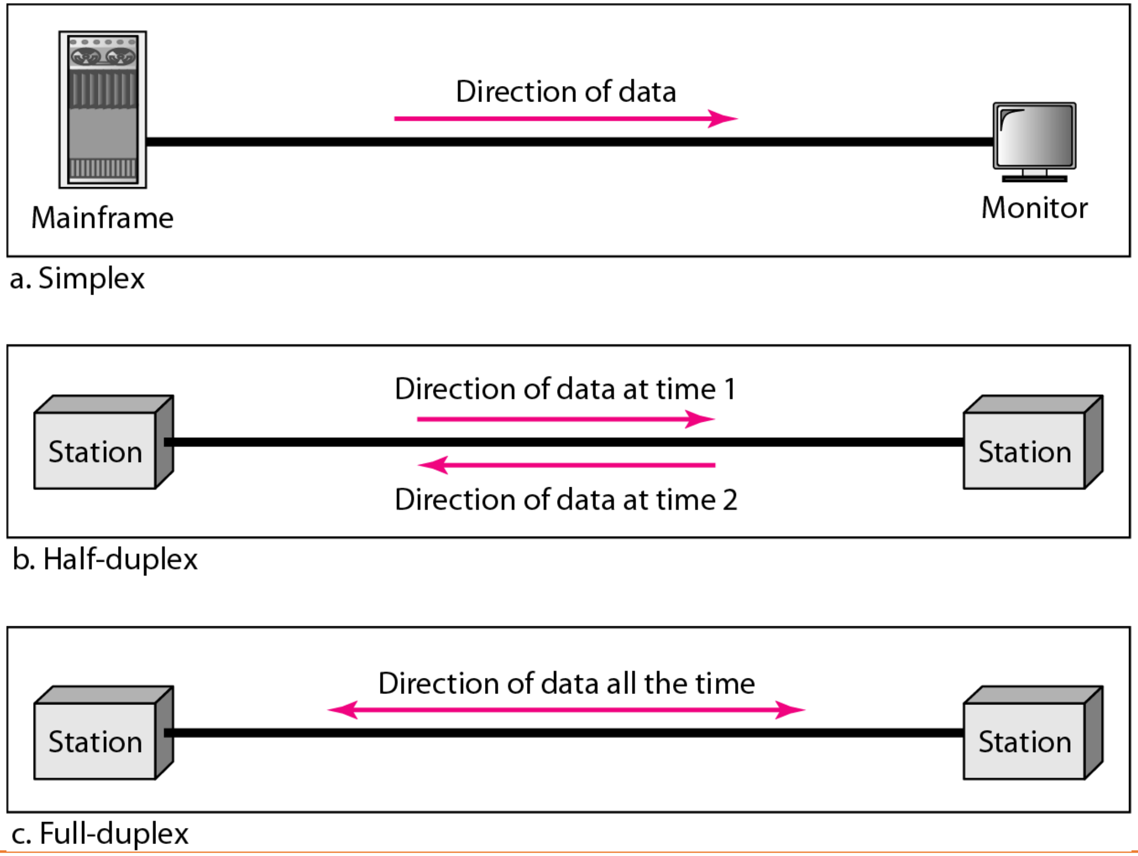
Transmission mode

How does data flow through the circuit

Configuration types:

- **Simplex**
- **Half-Duplex**
- **Full-Duplex**

Transmission mode



Line configuration

Circuit configuration

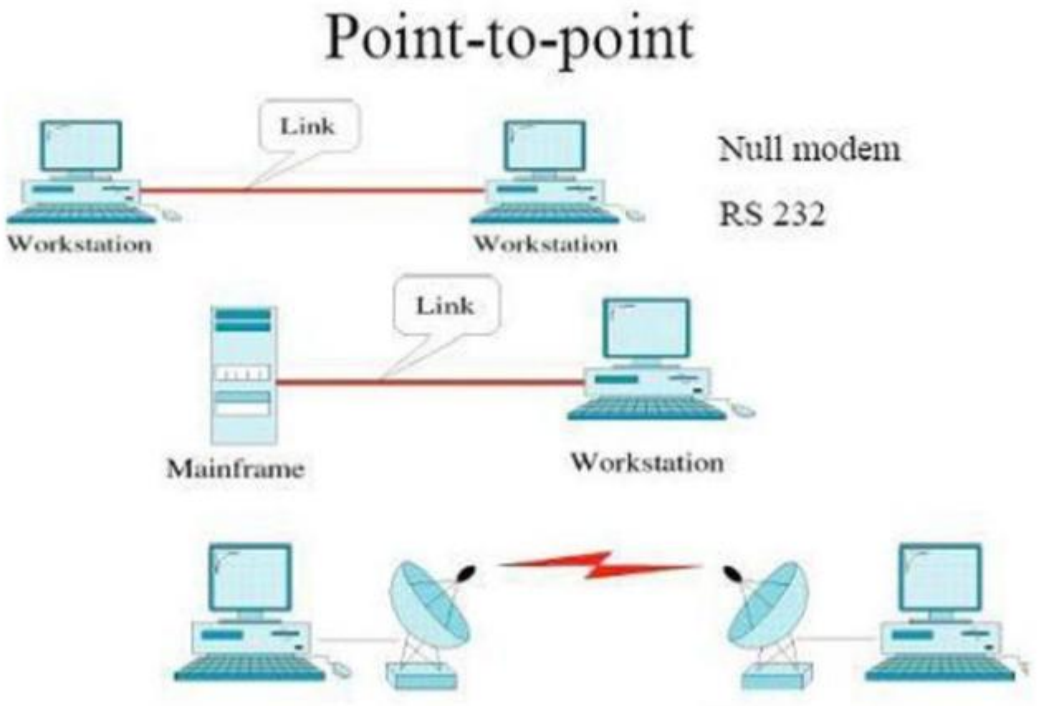
Basic physical layout of the circuit

Configuration types:

- **Point-to-Point Configuration**
- **Multipoint Configuration**

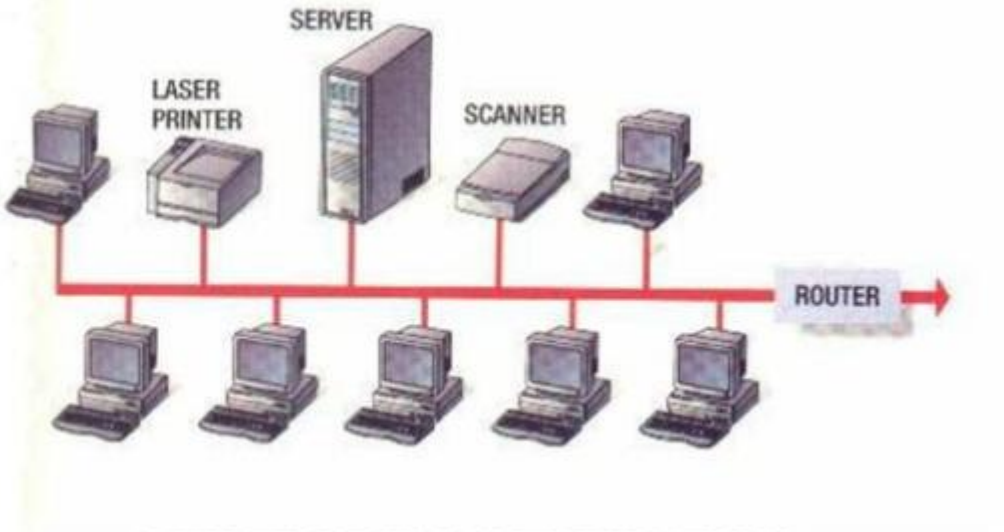
Point-to-point configuration

- Direct link,
- Single transmitter and receiver.



Multipoint configuration

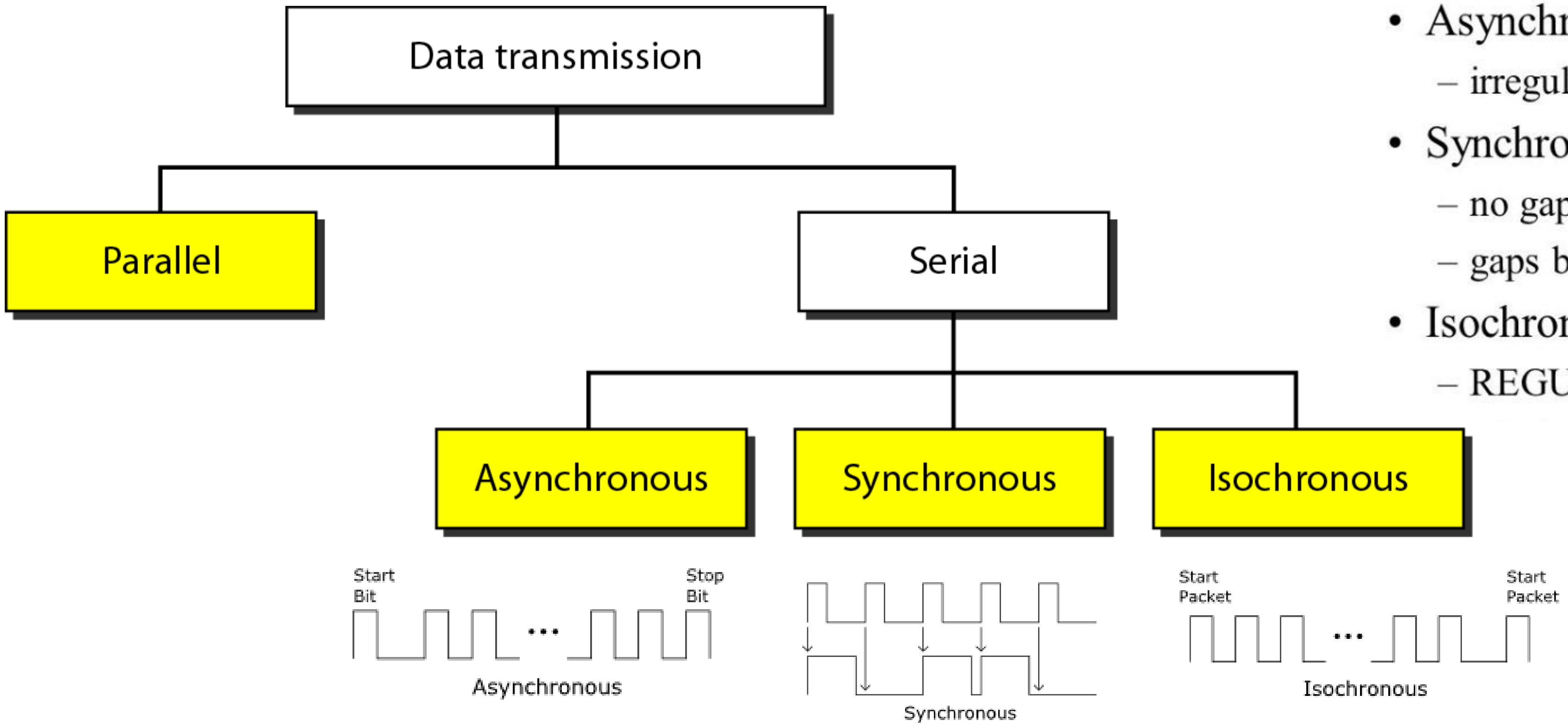
- direct link,
- More than two devices share the link,
- ex. multiple recipients of single transmission.



MULTIPOINT CONFIGURATION

Transmission modes

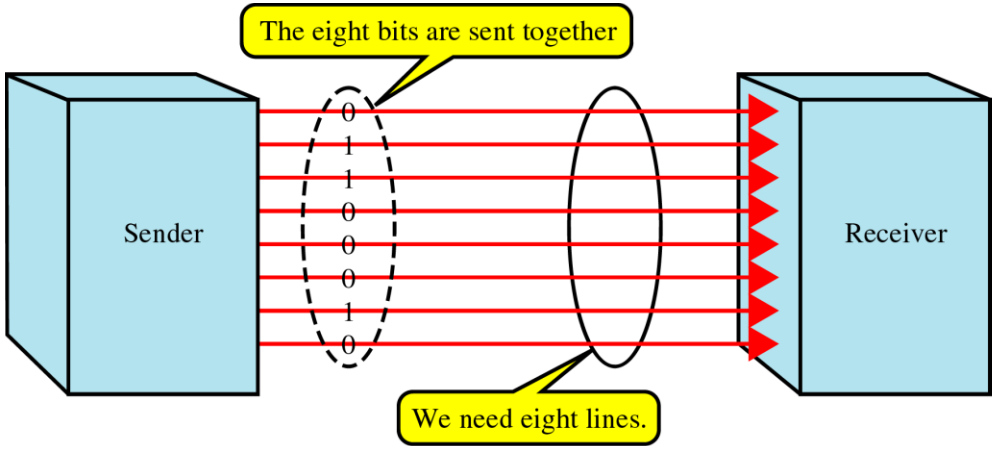
Transmission modes



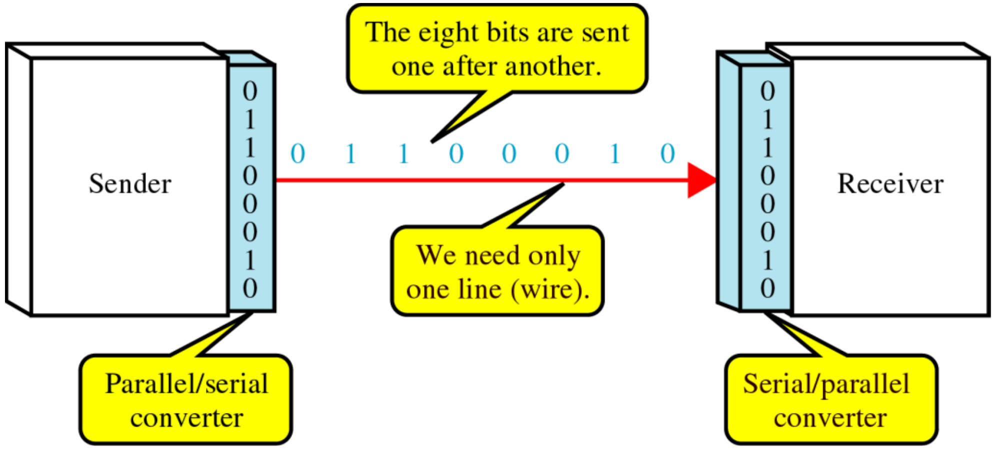
- Asynchronous
  - irregular gaps between bytes
- Synchronous
  - no gaps between bytes
  - gaps between blocks
- Isochronous
  - REGULAR gaps between blocks

Transmission Modes : Parallel & Serial

- **Parallel Transmission** use n wires to send n bits at one time synchronously.
  - Advantage: very speed.
  - Disadvantage: more expensive, limited to short distances.

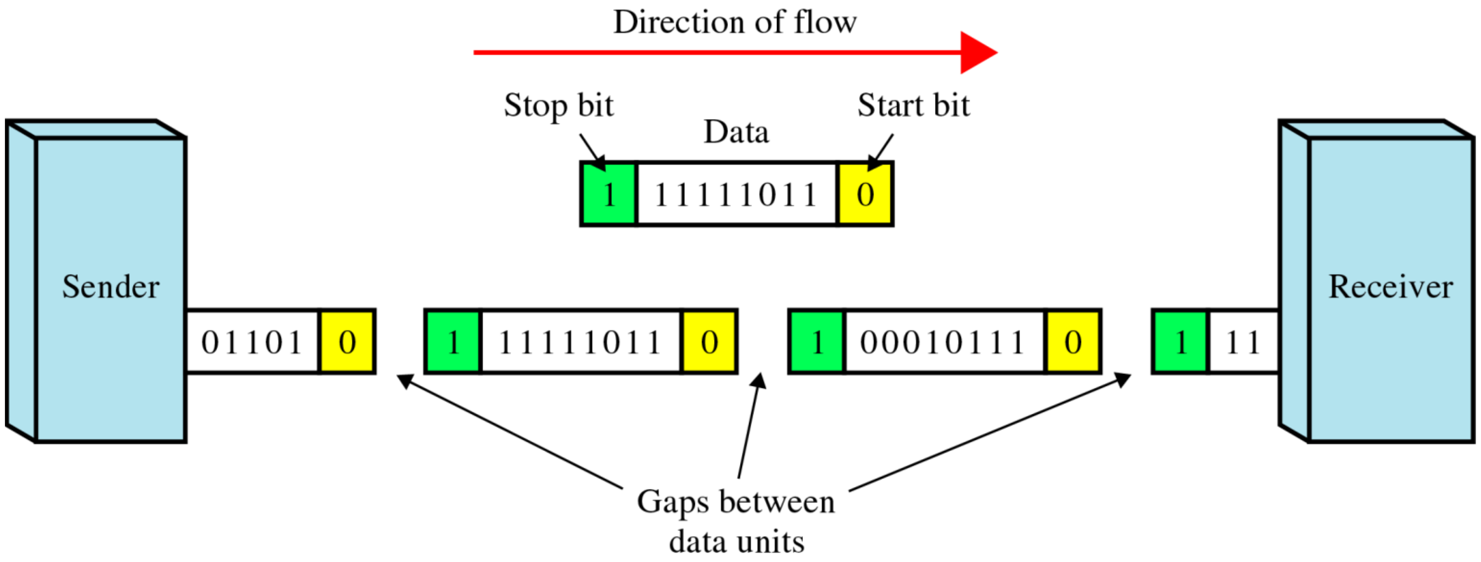


- **Serial Transmission** use one communication channel.
  - Advantage: less expensive, more distance.
  - Disadvantage: less speed.



Transmission Modes: Serial - Asynchronous Transmission

- uses start bit (0) and stop bits (1s)
- A gap between two bytes: idle state or stop bits
- It means asynchronous at byte level.
- Must still be synchronized at bit level
- Good for low-speed communications generate data at random time intervals (keyboard, modem, terminal). Example RS-232 interface.



Transmission Modes: Serial - Synchronous Transmission

- uses bit stream is combined into “frames”.
- Special sequence of 1/0 between frames: No gap.
- Timing is important in midstream.
- Byte synchronization in the data link layer.
- Advantage: speed (high-speed transmission).