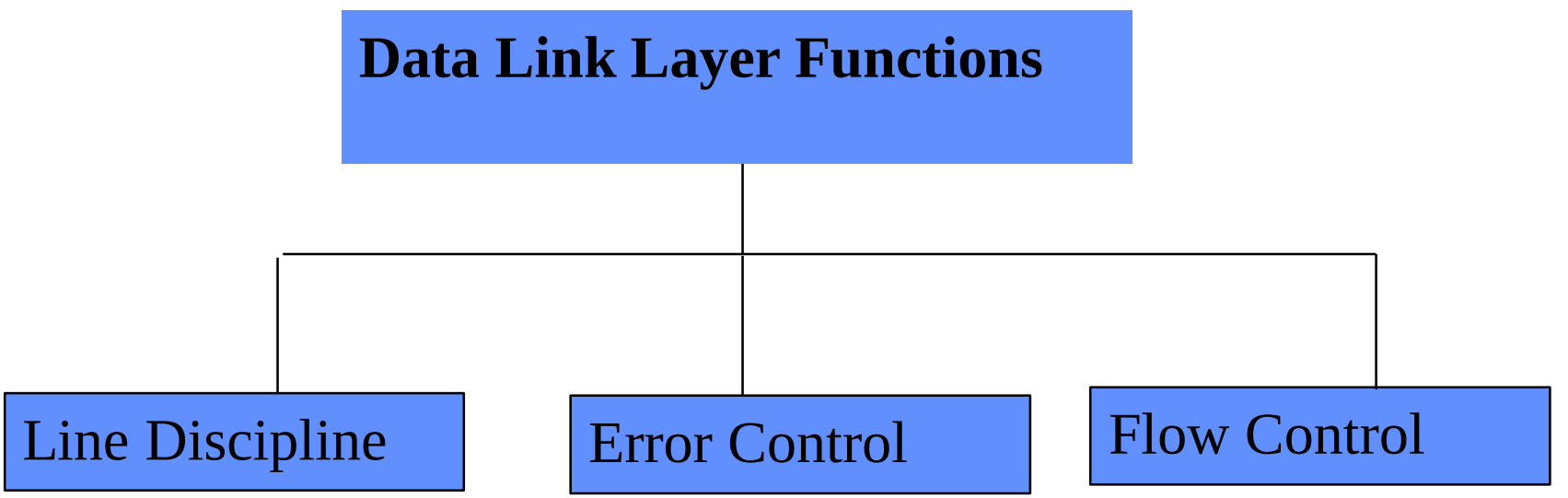
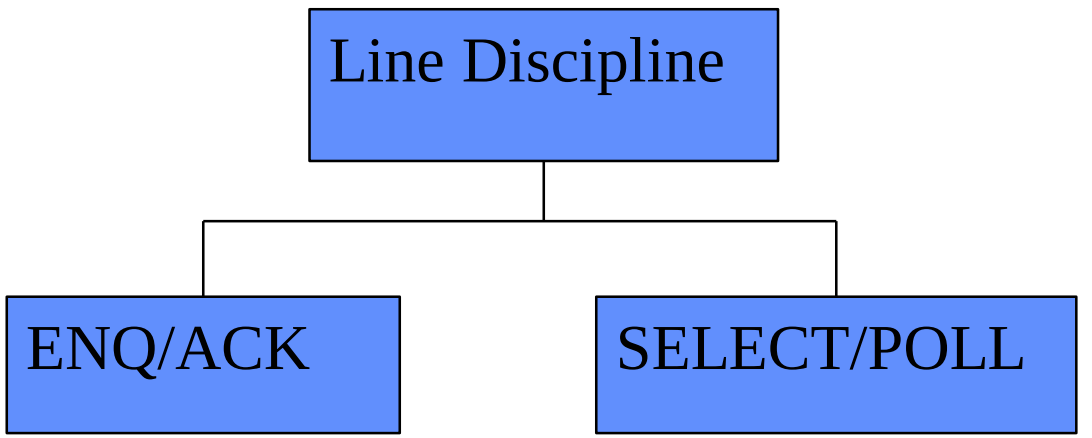
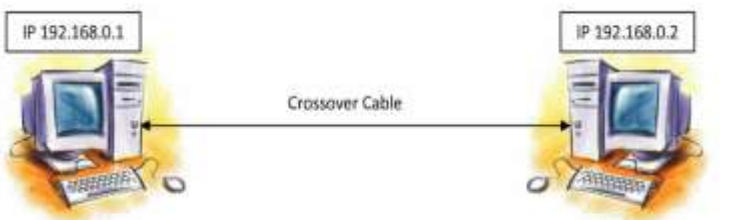
Data Link Layer

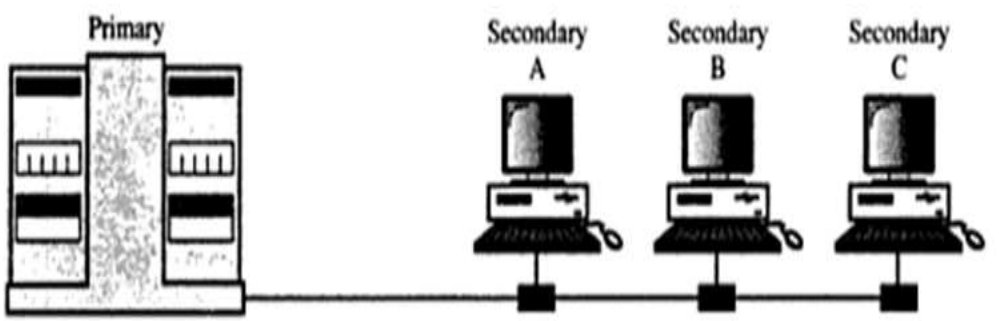


Line Discipline



ENQ/ACK:
Used in peer-to-peer.





SELECT/POLL:
Used in Primary ands Secondary Devices.

Line Discipline

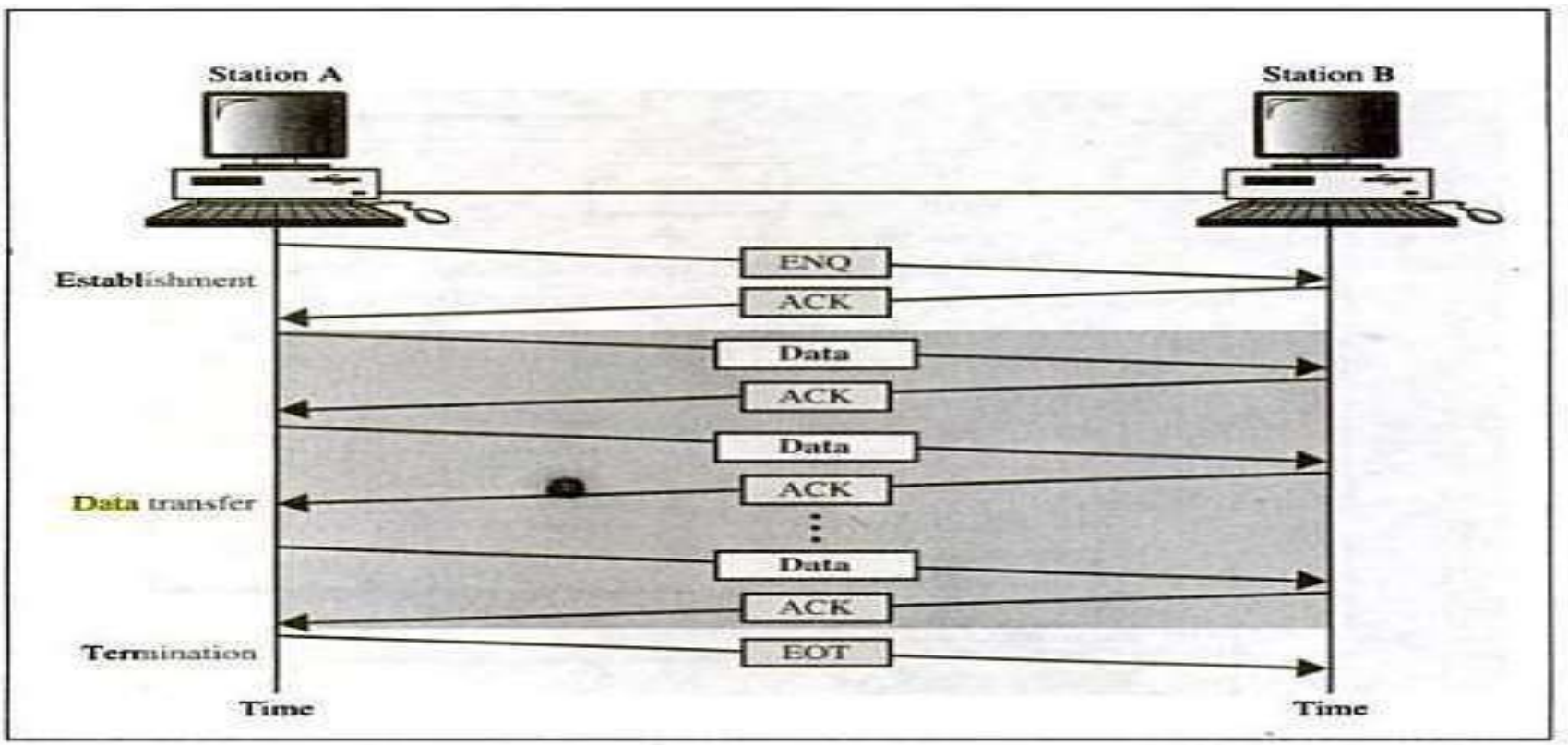
ENQ (Enquiry)

- Used in peer –peer communication
- Enquire whether there is a required link between two devices
- Check whether the intended device is capable to receive

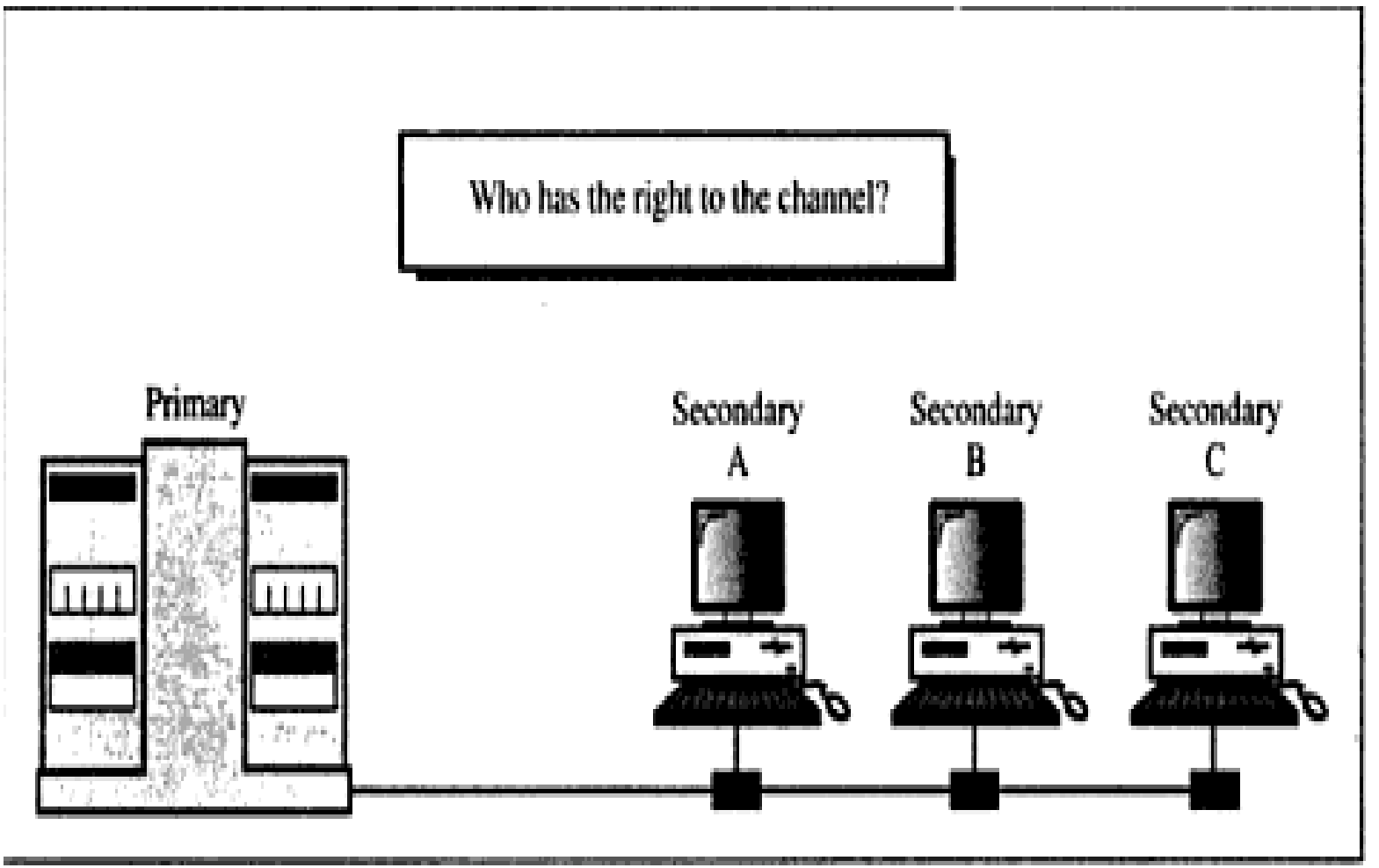
ACK (Acknowledgment)

- Used in Primary secondary communication
- The intended device will acknowledge about its status to the receiver

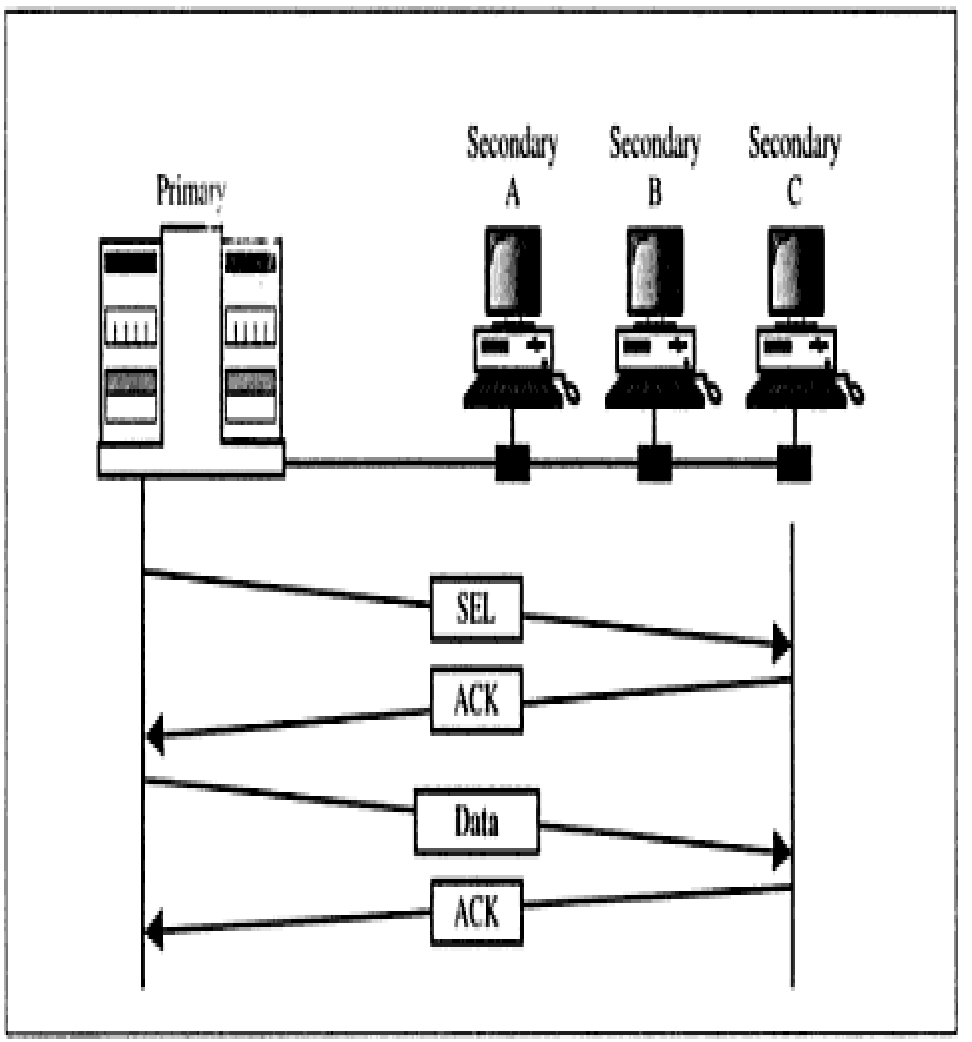
Line Discipline :ENQ/ACK



Line Discipline : Select/Poll



Line Discipline : Select



Select

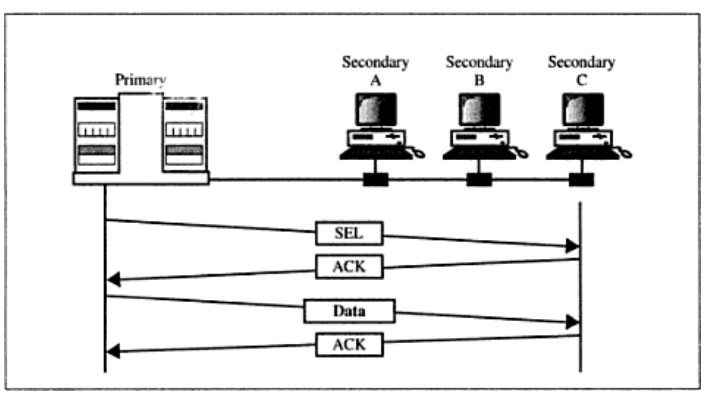
It is uses whenever the primary device with primary secondary relationship.

Line Discipline : Poll

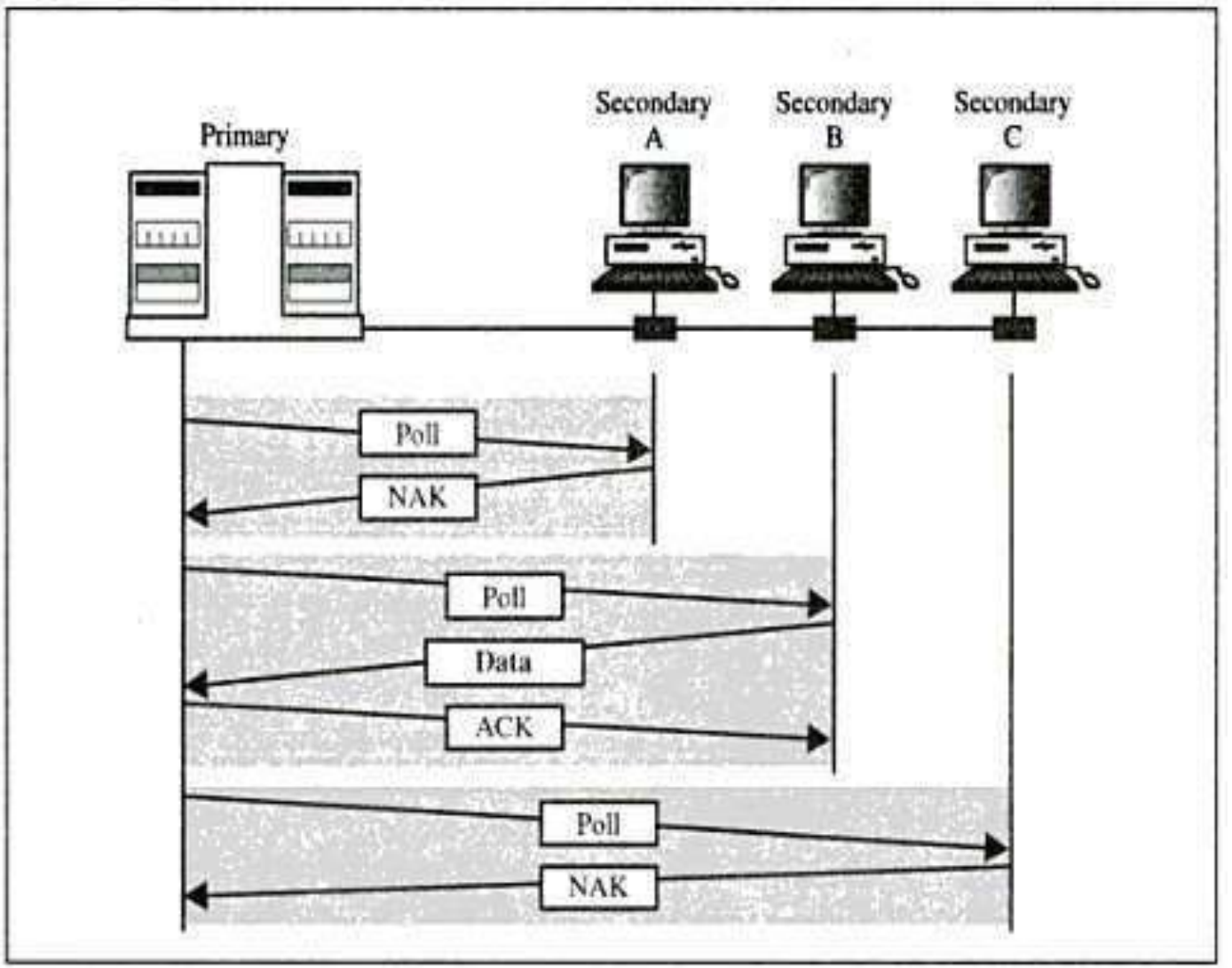
Poll

- The polling function is used by the primary device to Select transmissions from the secondary devices.
- If the primary device is ready to receive data , It ask each device in turn if it has anything to send.

.

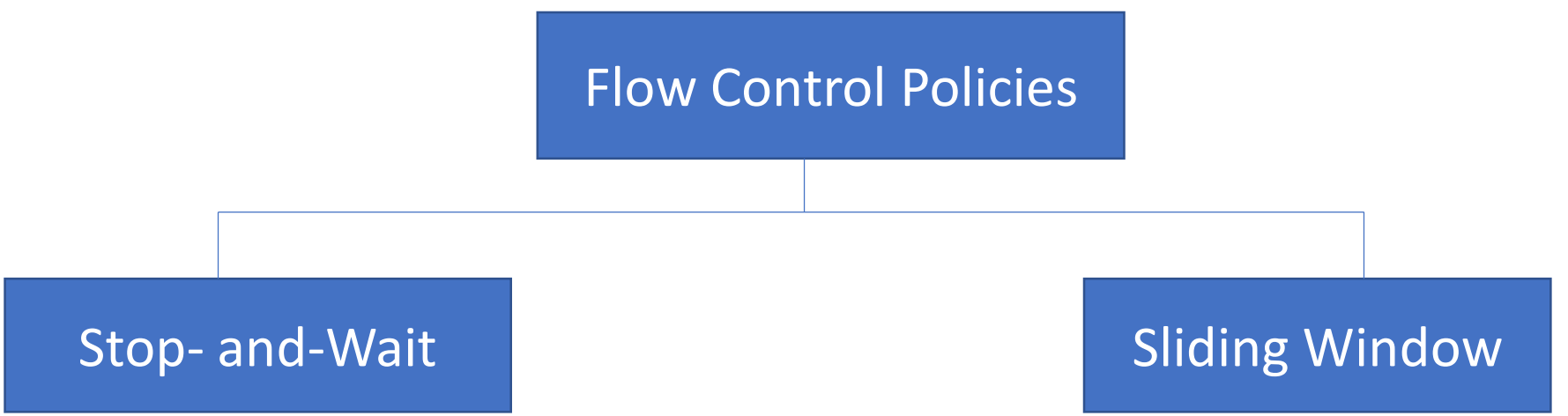


Poll


Primary
Secondary A
Secondary B
Secondary C
Poll
NAK
Poll
Data
ACK
Poll
NAK

Flow control

- It is a set of procedures to tell the sender how much data it can transmit and wait before receiving the acknowledgement from the receiver.

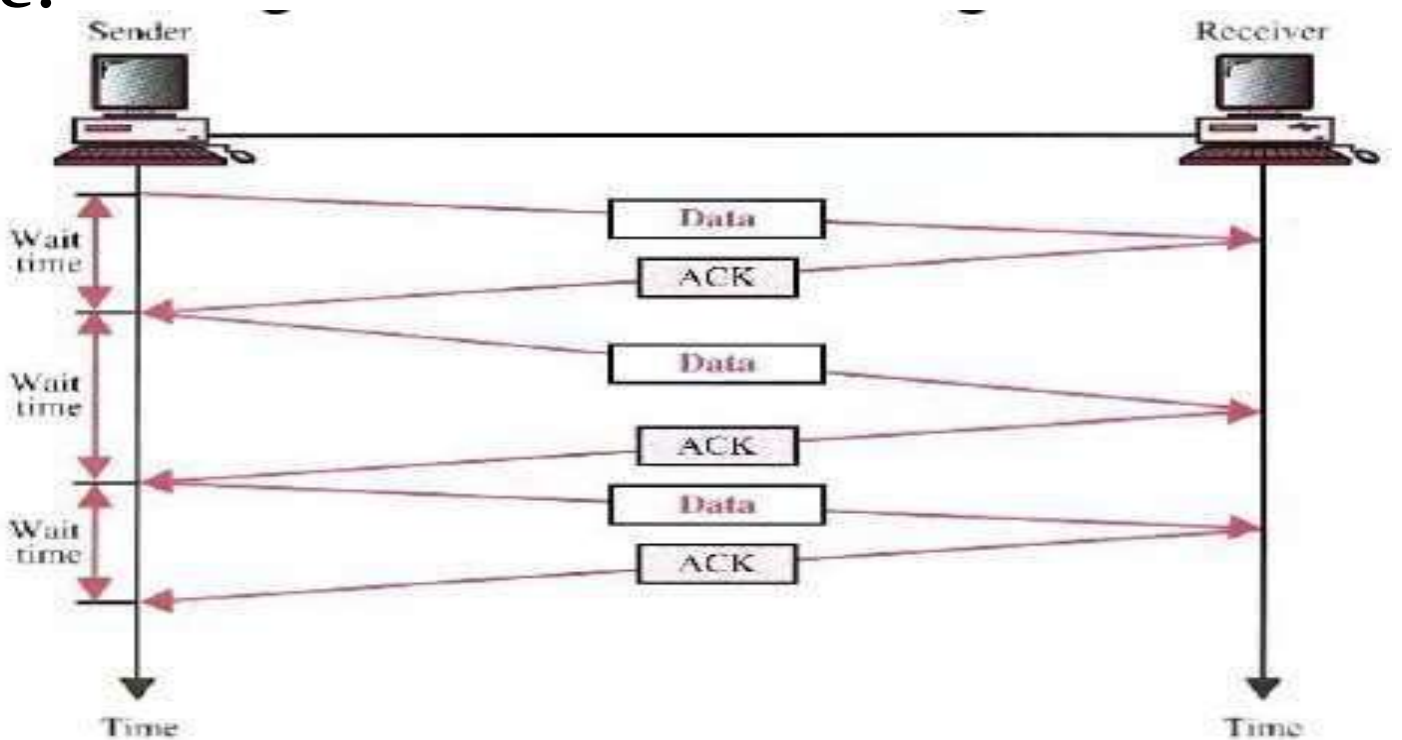


Send one frame at a time

Send several frames at a time

Flow control
Stop-and-wait

- Sender sends one frame and waits for an acknowledgement before sending the next frame.



Flow control
Stop-and-wait

- Advantages:

  Simplicity.

  Each frame is checked an acknowledged before the next frame is sent.

- Disadvantages:

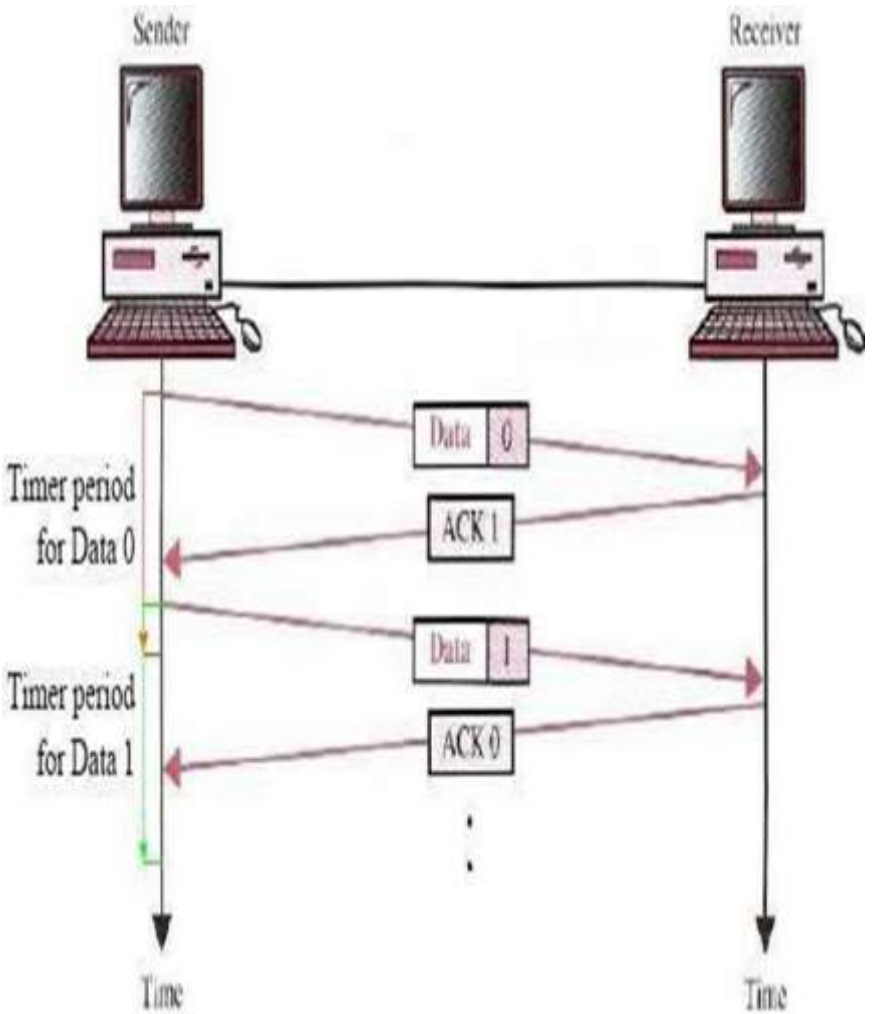
  Slow.

  - Can add significantly to the total transmission time if the distance between devices is long.

  Inefficiency

  - Each frame is alone on the line.

Flow control
Stop-and-wait ARQ



Stop-and-wait ARQ is for data retransmission.

ARQ needs to cater for three possible cases:
Corrupted Frame
Lost Frame
Lost of acknowledgement

Flow control
Stop-and-wait ARQ

Four features added to basic flow control

1. Sender keeps a copy of last transmitted frame until it receives an acknowledgement.
2. Sending device must equipped with timer

   Timer starts every time a frame is transmitted.

   If the timer times out and sender has not received an acknowledgement ,then it assumes that the frame is lost and retransmission is required.

Flow control
Stop-and-wait ARQ

3. Numbering of data frame and acknowledgement frames

   allows the receiver to identify duplicate transmission.

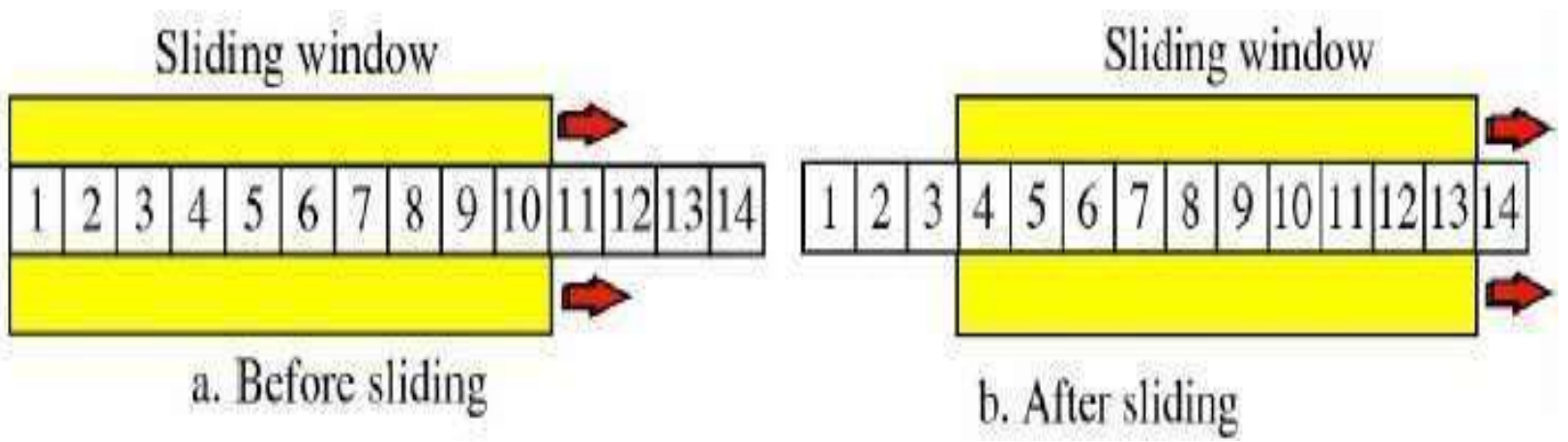
4. Receiving device returns a NAK frame if it detects an error in a data frame.

Flow control
SlidingWindow

- Sender can send several frames before needing an acknowledgement.

Advantages:

- The link can carry several frames at once.
- Its capacity can be used efficiently.



Flow control
Sliding Window ARQ

It is an extension of sliding window with retransmission

Two options are

1.Go-back-n ARQ

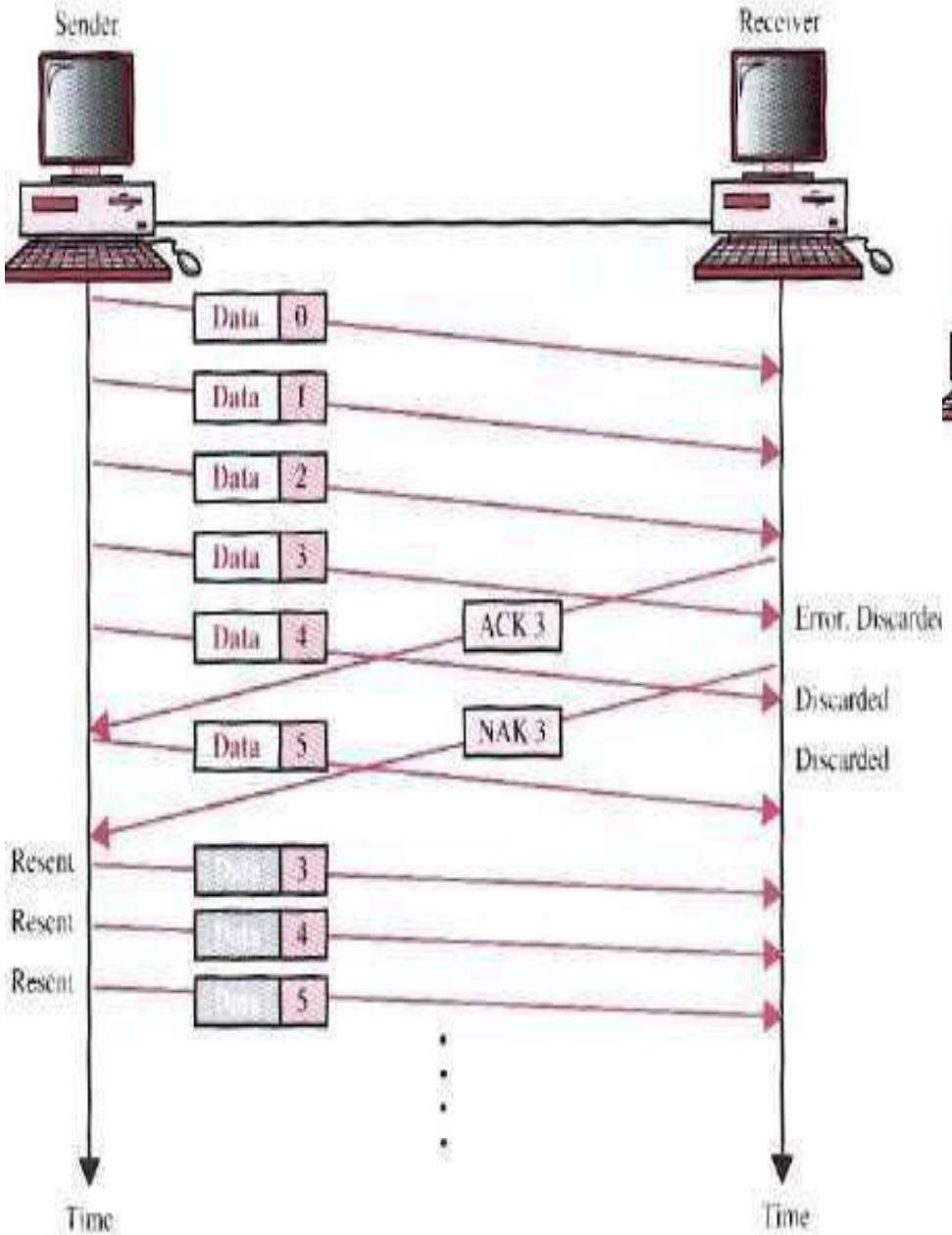
Retransmission of all the frames starting from corrupted frame

2.Selective-reject ARQ

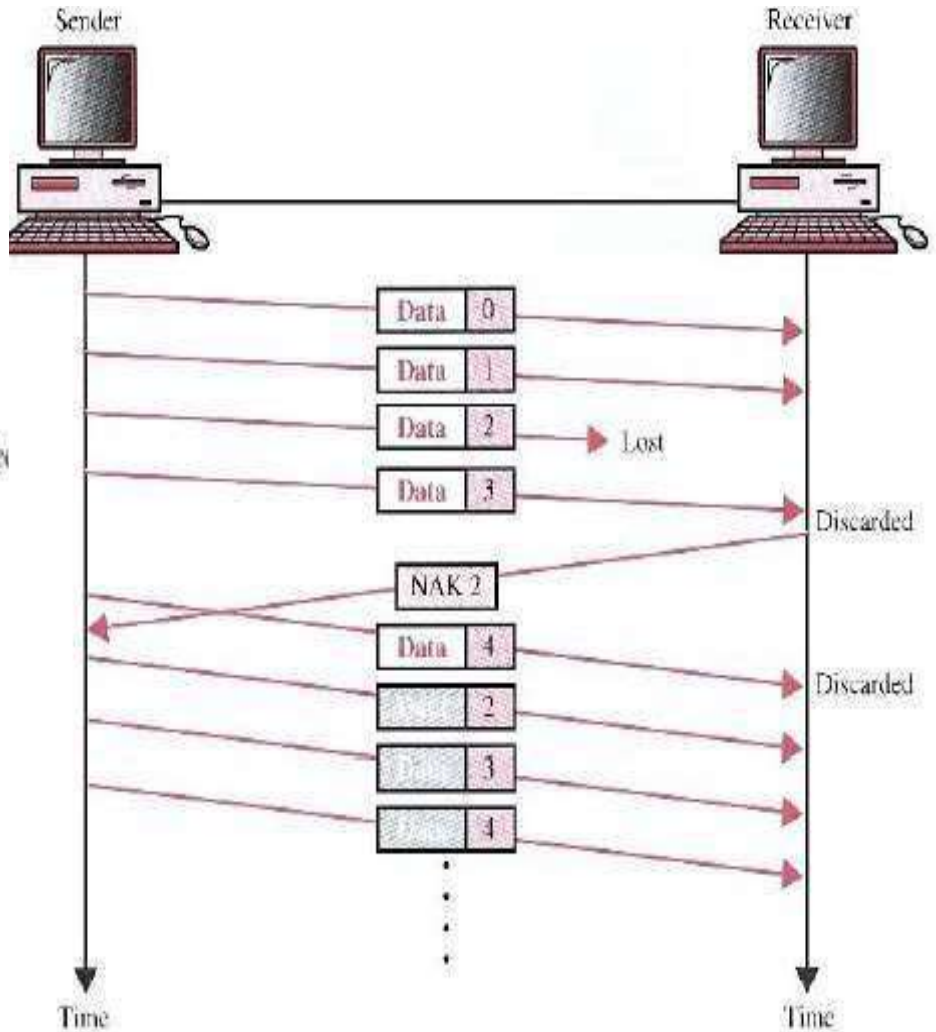
Retransmission of only corrupted frames.

Stop-and-wait protocol is a special case of sliding-window protocol with a window size of 1.

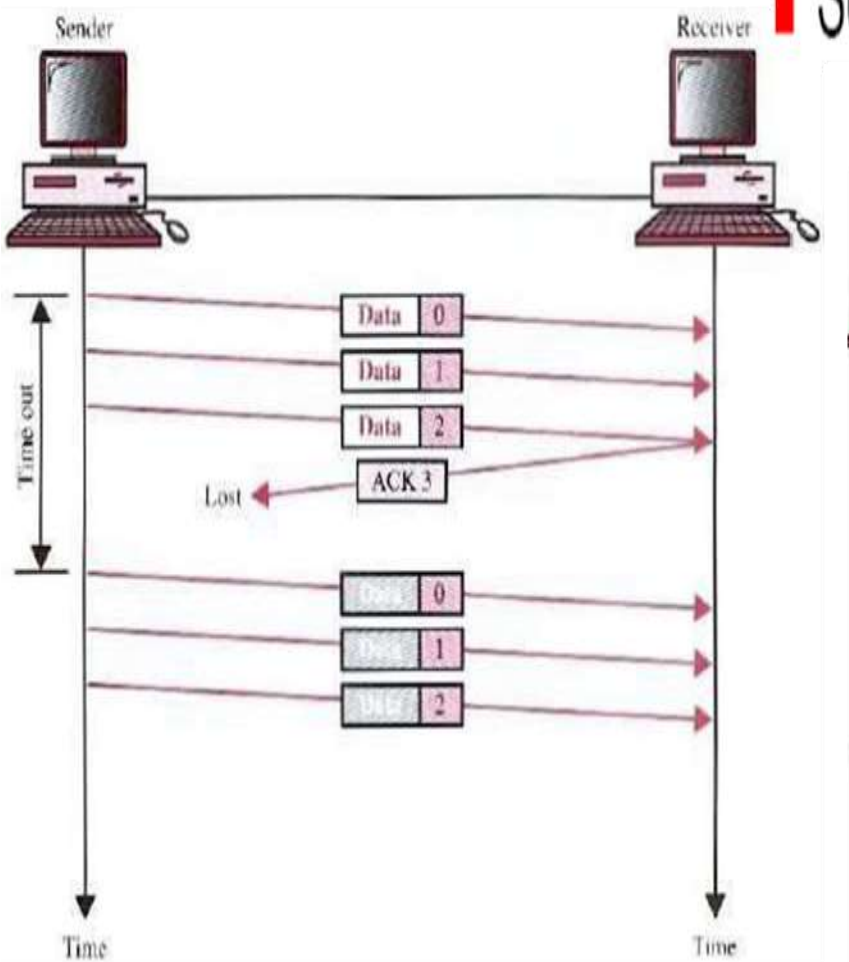
- Go-back-n, corrupted frame case:

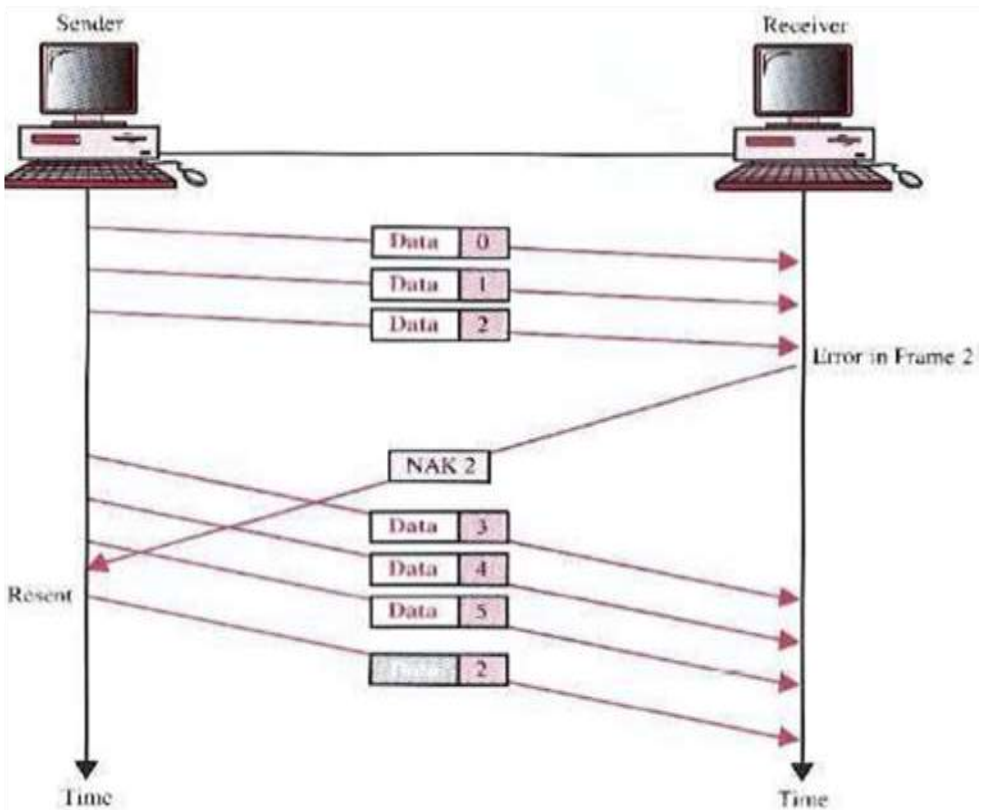


- Go-back-n, lost frame case:



- Selective-reject, corrupted frame case:





- Selective-reject, lost frame case:
  - Similar to corrupted frame case.
  - Lost frame is only detected after the next frame has been received correctly.
- Selective-reject, lost ACK case:
  - Similar to lost ACK case of go-back-n.
  - All frames from last ACK have to be retransmitted when the timer times out.
- Go-back-n vs. Selective-reject:
  - Selective-reject is expensive due to:
    - The need for extra logic to select specific frame for retransmission.
    - The need to buffer correctly received frames.
    - The complexity of sorting the received frames.
  - Go-back-n is much simpler and more commonly used.

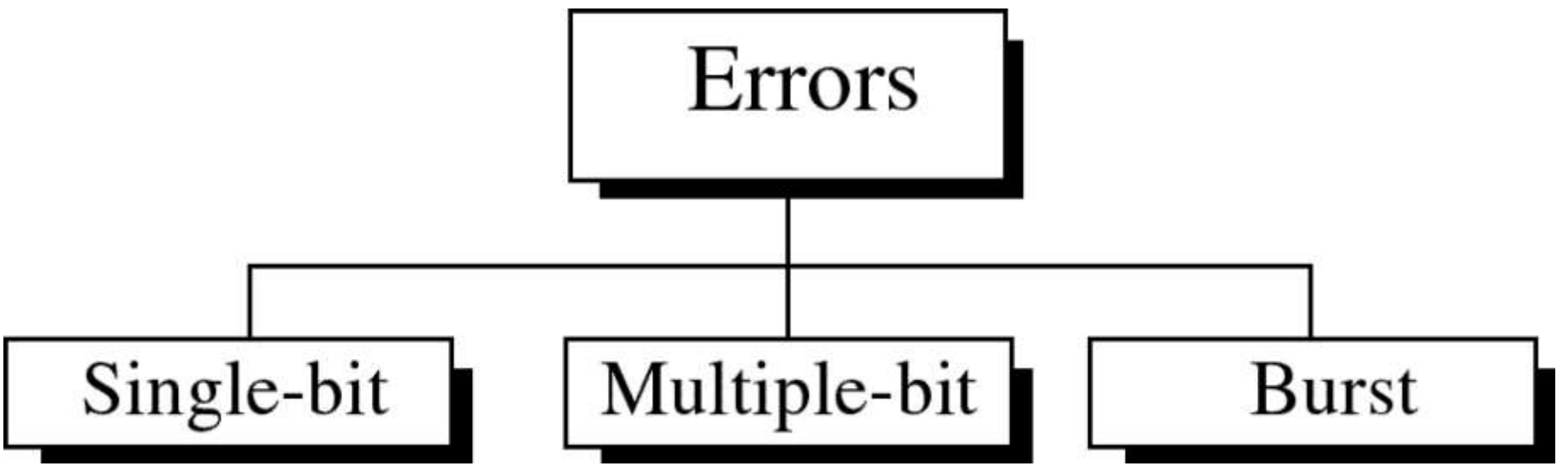
Error Detection and Correction

- **Error Detection**:
  Allows receiver to detect presence of error
- **Error Correction:**
  Allows receiver to correct the errors

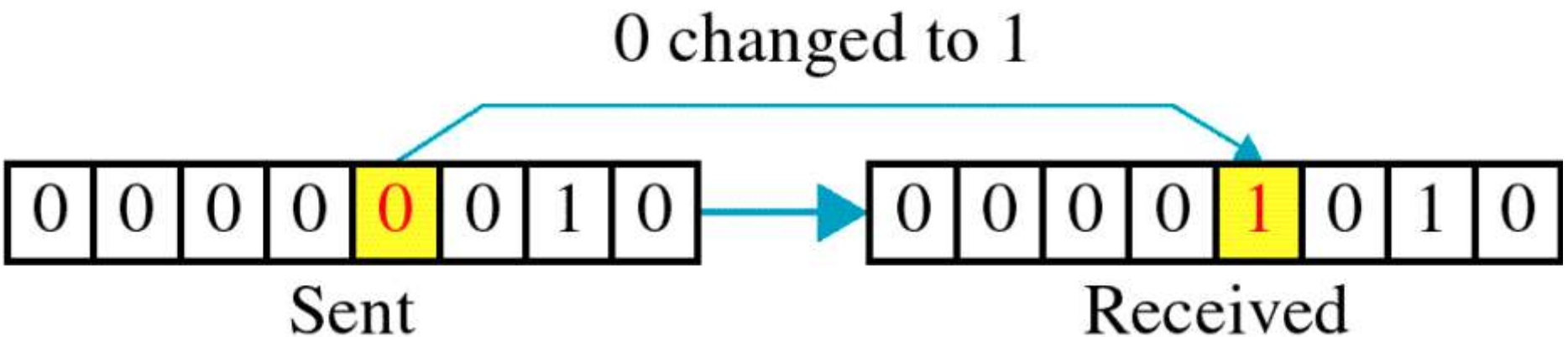
Basic concepts

- ★ Networks must be able to transfer data from one device to another with complete accuracy.
- ★ Data can be corrupted during transmission.
- ★ For reliable communication, errors must be detected and corrected.
- ★ **Error detection and correction** are implemented either at the **data link layer** or the **transport layer** of the OSI model.

Types of Errors



Single-bit error



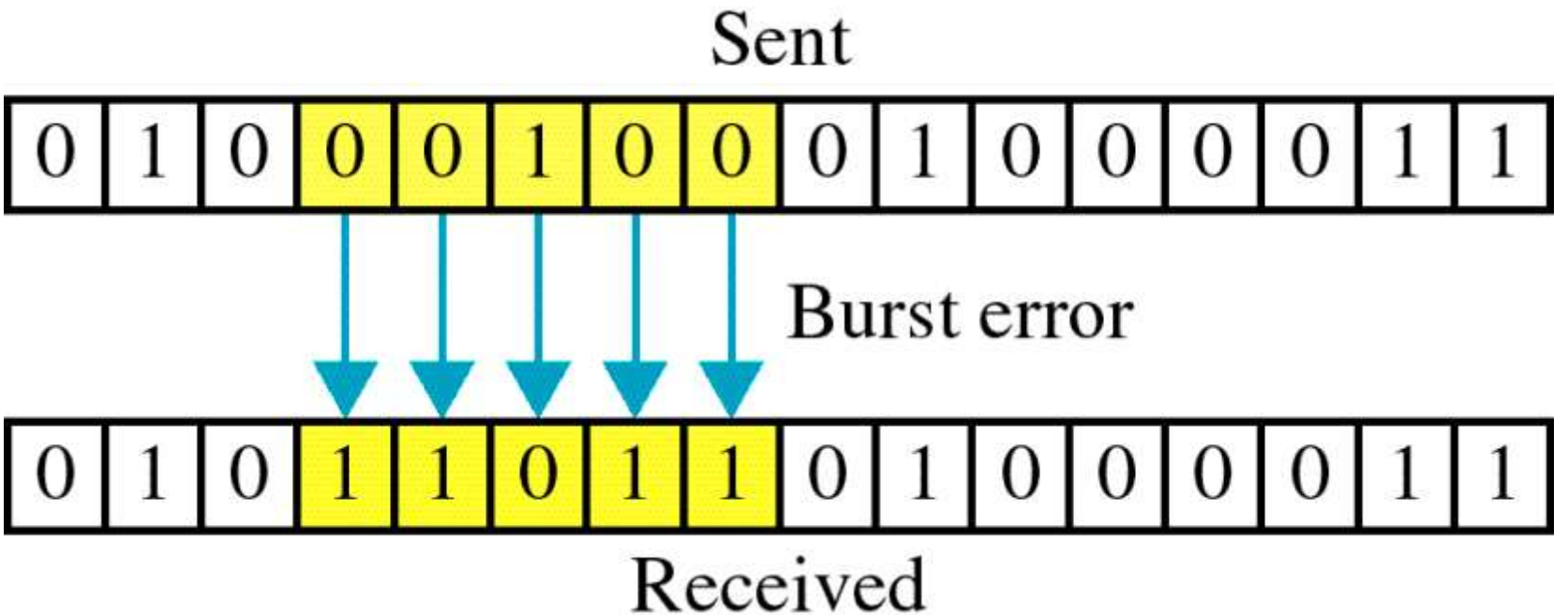
Single bit errors are the **least likely** type of errors in serial data transmission because the noise must have a very short duration which is very rare. However this kind of errors can happen in parallel transmission.

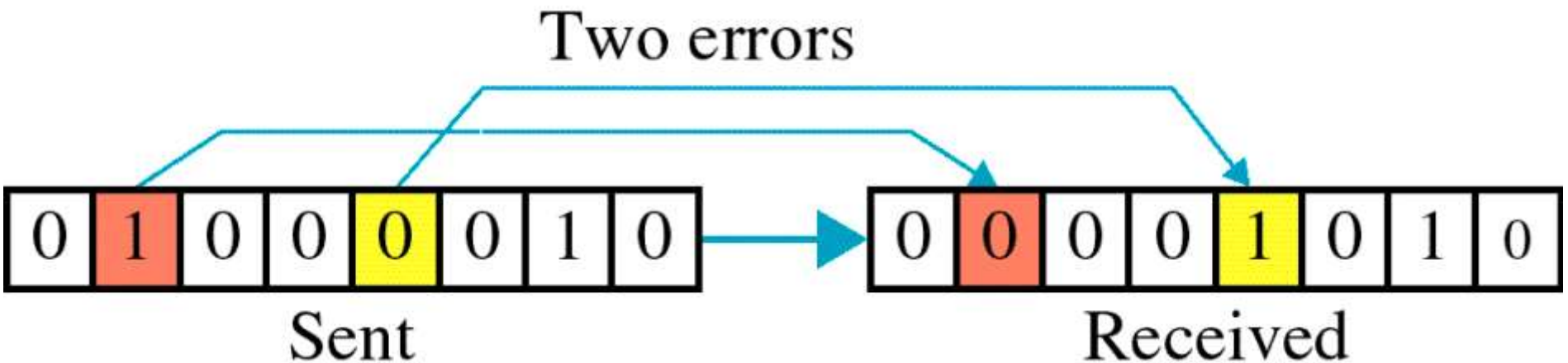
Example:

★If data is sent at 1Mbps then each bit lasts only 1/1,000,000 sec. or 1 μs.

★For a single-bit error to occur, the noise must have a duration of only 1 μs, which is very rare.

Burst error


Sent
0 1 0 0 0 1 0 0 0 1 0 0 0 0 1 1
Burst error
0 1 0 1 1 0 1 1 0 1 0 0 0 0 1 1
Received


Two errors
0 1 0 0 0 0 1 0
Sent
0 0 0 0 1 0 1 0
Received

The term **burst error** means that two or more bits in the data unit have changed from 1 to 0 or from 0 to 1.

Burst errors does not necessarily **mean that the errors occur in consecutive bits**, the length of the burst is measured from the first corrupted bit to the last corrupted bit. Some bits in between may not have been corrupted.

- ★**Burst error is most likely to happen in serial transmission** since the duration of noise is normally longer than the duration of a bit.
- ★The number of bits affected depends on the data rate and duration of noise.

Example:

- ➔If data is sent at rate = 1Kbps then a noise of 1/100 sec can affect 10 bits.(1/100*1000)

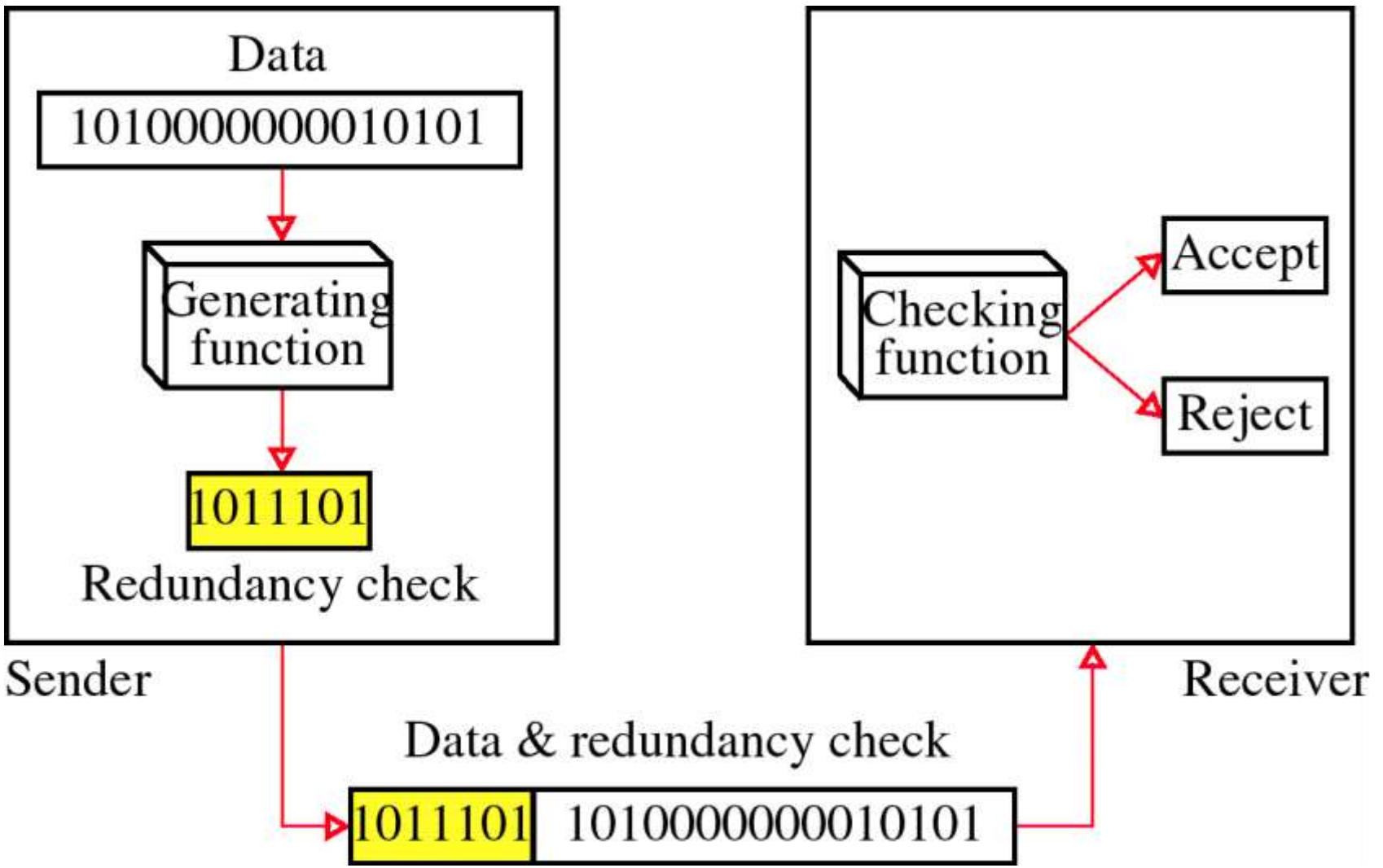
- ➔If same data is sent at rate = 1Mbps then a noise of 1/100 sec can affect 10,000 bits.($1/100*10^6$)

Error detection

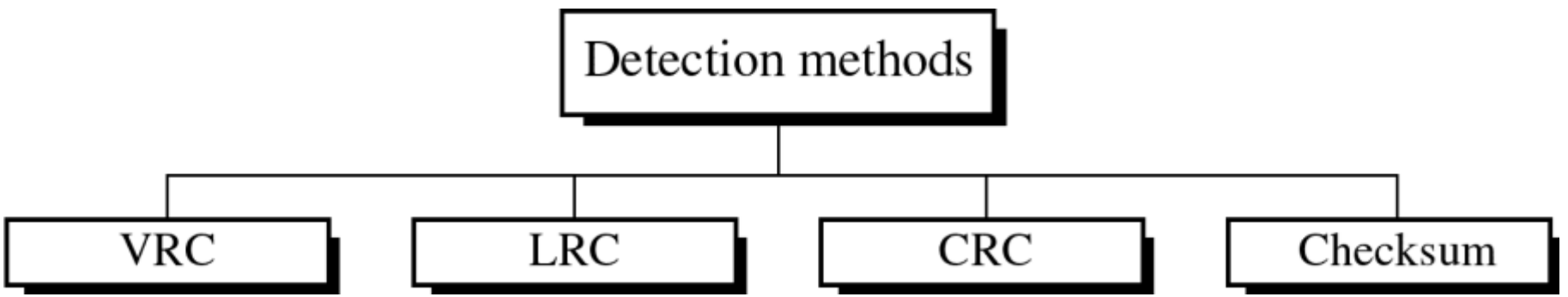
Error detection means to decide whether the received data is correct or not without having a copy of the original message.

Error detection **uses the concept of redundancy**, **which means** adding extra bits for detecting errors at the destination.

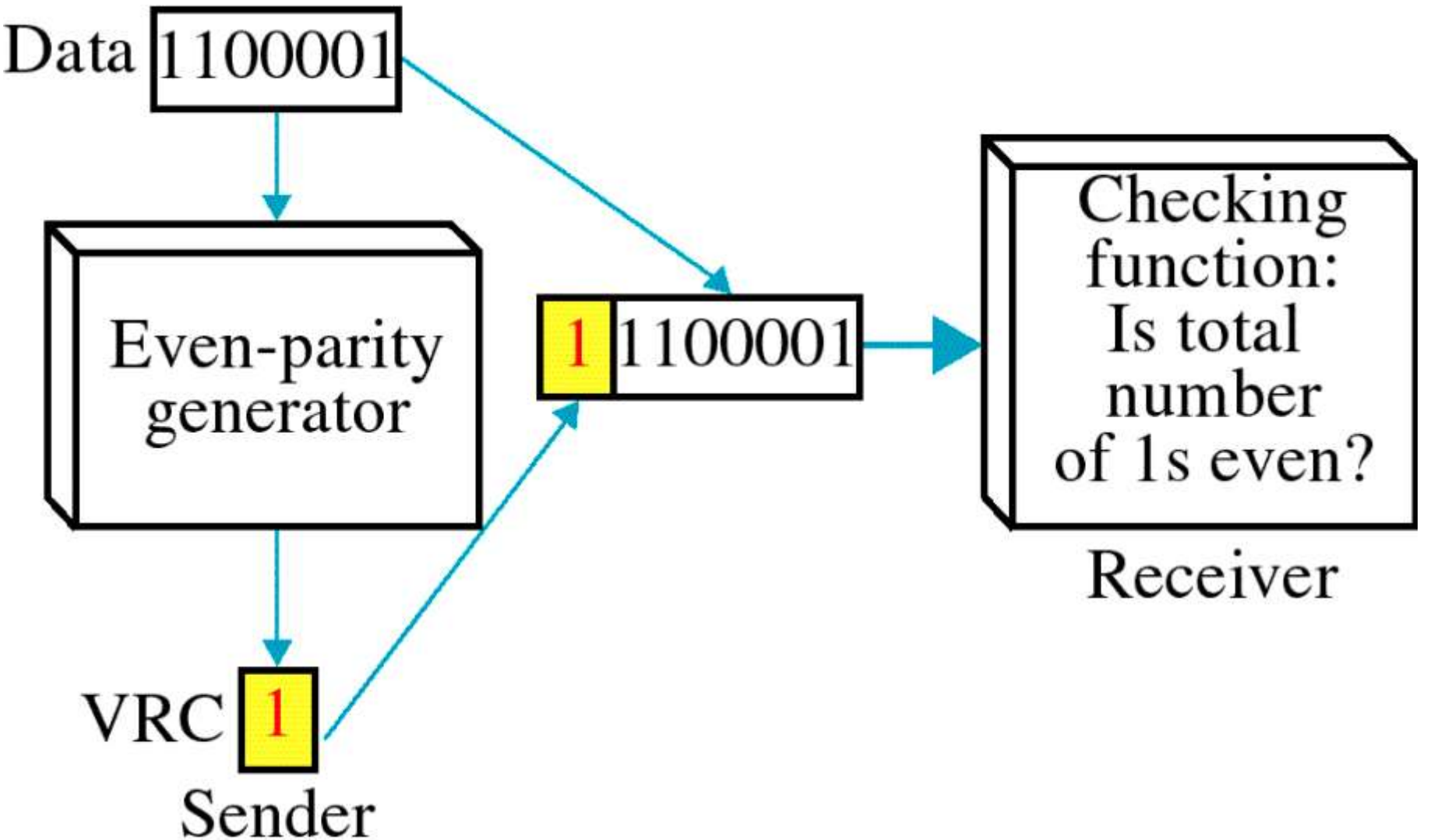
Redundancy


Data
1010000000010101
Generating function
1011101
Redundancy check
Sender
Checking function
Accept
Reject
Receiver
Data & redundancy check
1011101
1010000000010101

Four types of redundancy checks are used in data communications



Vertical Redundancy Check
VRC



Vertical Redundancy Check
VRC

- The word cute is coded in ASCII as

11000011	**1110101**	**1110100**	**1100101**
c	u	t	e

Using Even Parity checking the sender will send

110000110 **11101011** **11101000** **11001010**

If the word is not corrupted during transmission the receiver will count the 1s in each character and comes up with(4,6,4,4) - all even numbers.

Vertical Redundancy Check
VRC

- If the word is corrupted during transmission

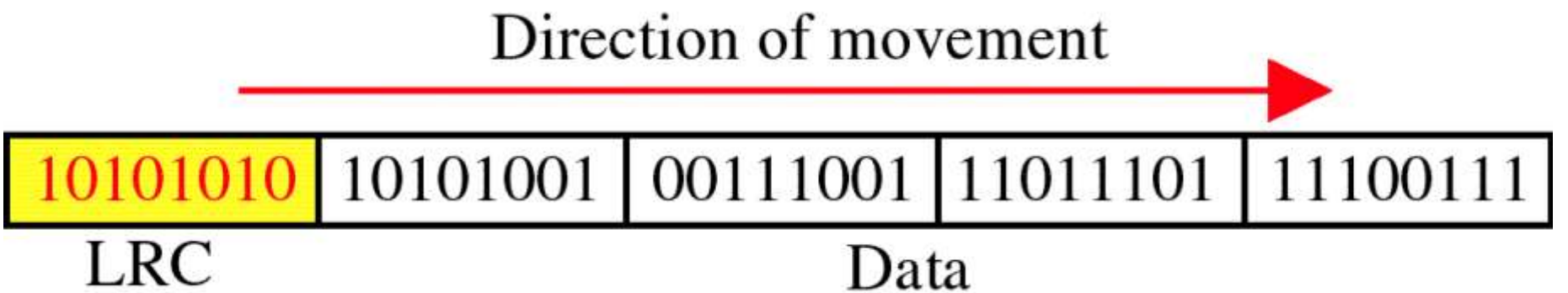
110010110 **11101011** **11101000** **11000010**

If the word is corrupted during transmission the receiver will count the 1s in each character and comes up with(5,6,4,3) - not all even numbers.

Performance

- It can detect single bit error
- It can detect burst errors only if the total number of errors is odd.

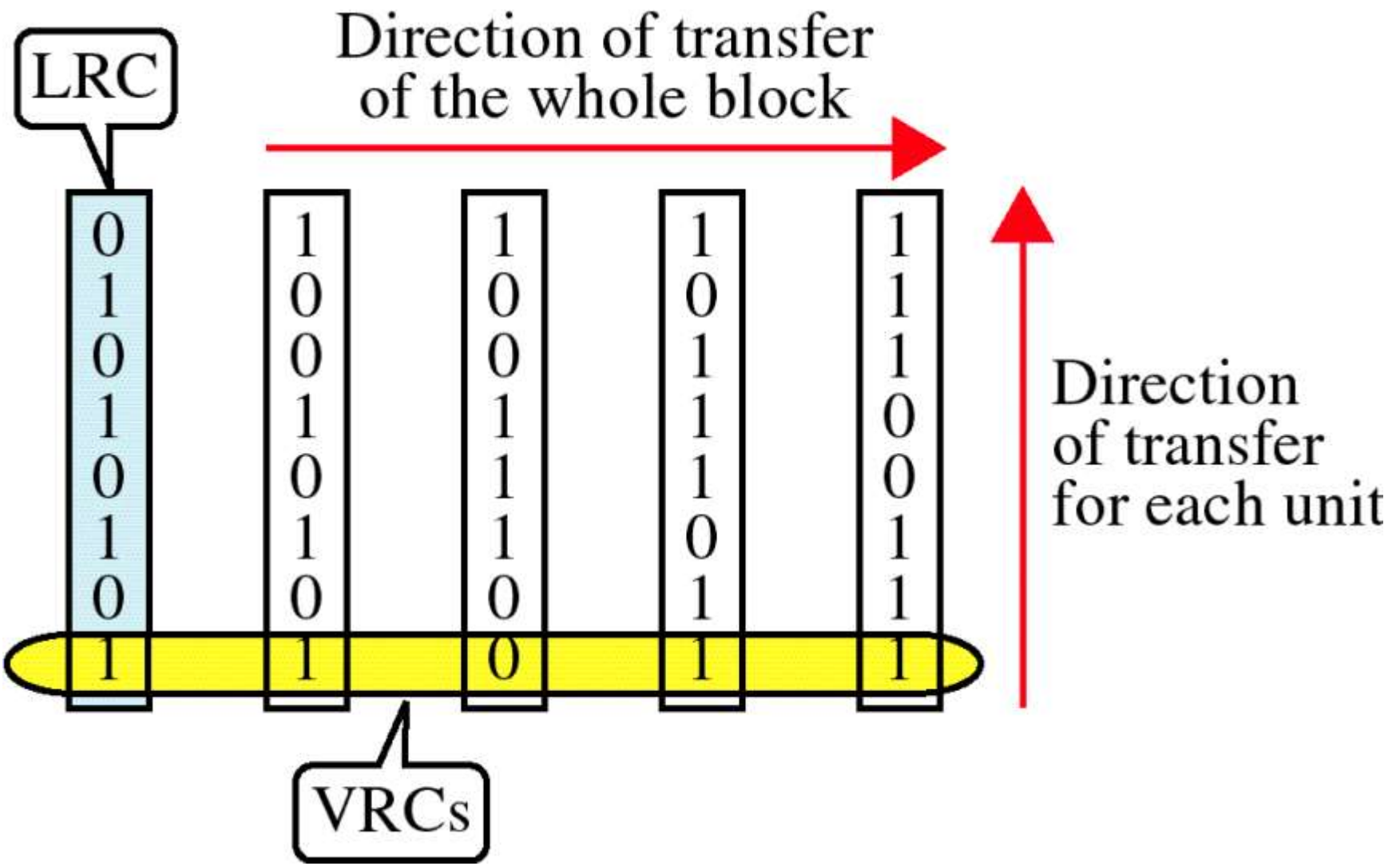
Longitudinal Redundancy Check
LRC



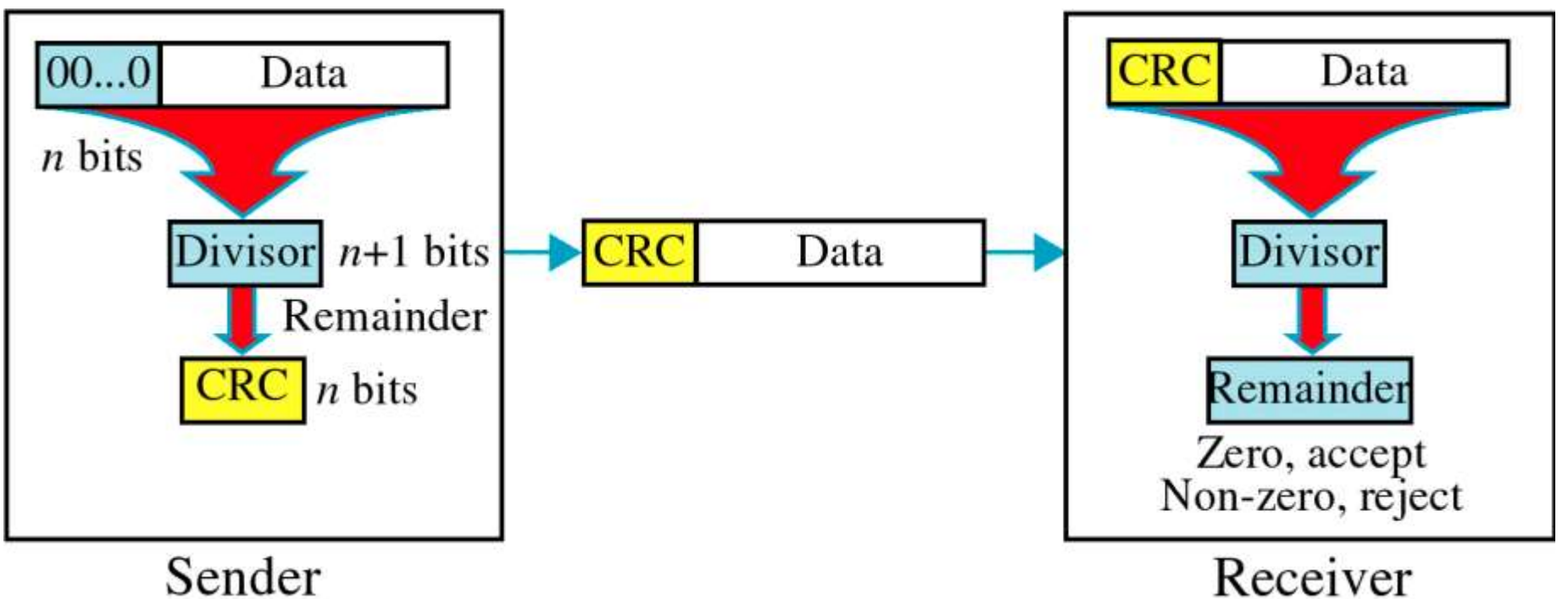
Performance

- LCR increases the likelihood of detecting burst errors.
- If two bits in one data units are damaged and two bits in exactly the same positions in another data unit are also damaged, the LRC checker will not detect an error.

VRC and LRC


LRC
Direction of transfer
of the whole block
0 1 0 1 0 1 0 1
1 0 0 1 0 1 0 0
1 0 0 1 1 1 1 0
1 0 1 1 1 0 1 1
1 1 1 0 0 1 1 1
1 1 0 1 1
Direction
of transfer
for each unit
VRCs

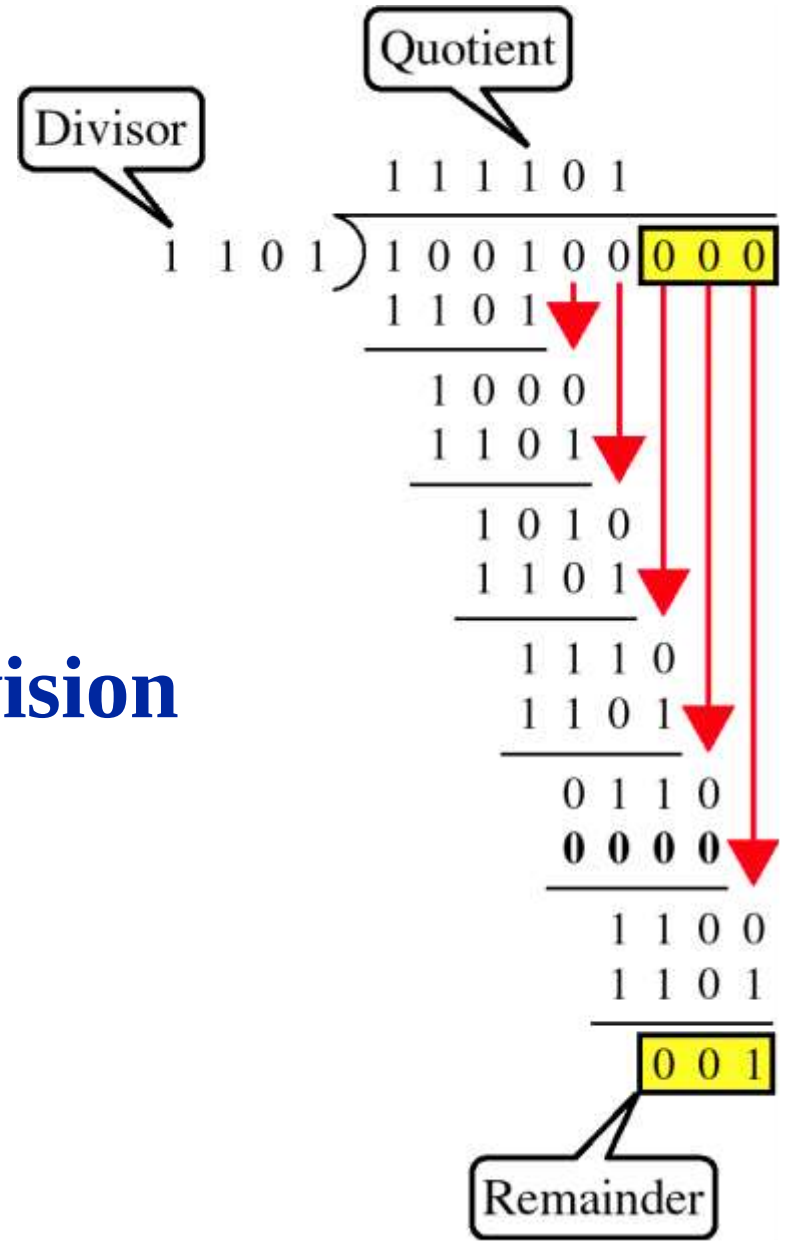
Cyclic Redundancy Check
CRC



Cyclic Redundancy Check

- Given a k-bit frame or message, the transmitter generates an n-bit sequence, known as a *frame check sequence (FCS)*, so that the resulting frame, consisting of (k+n) bits, is exactly divisible by some predetermined number.
- The receiver then divides the incoming frame by the same number and, if there is no remainder, assumes that there was no error.

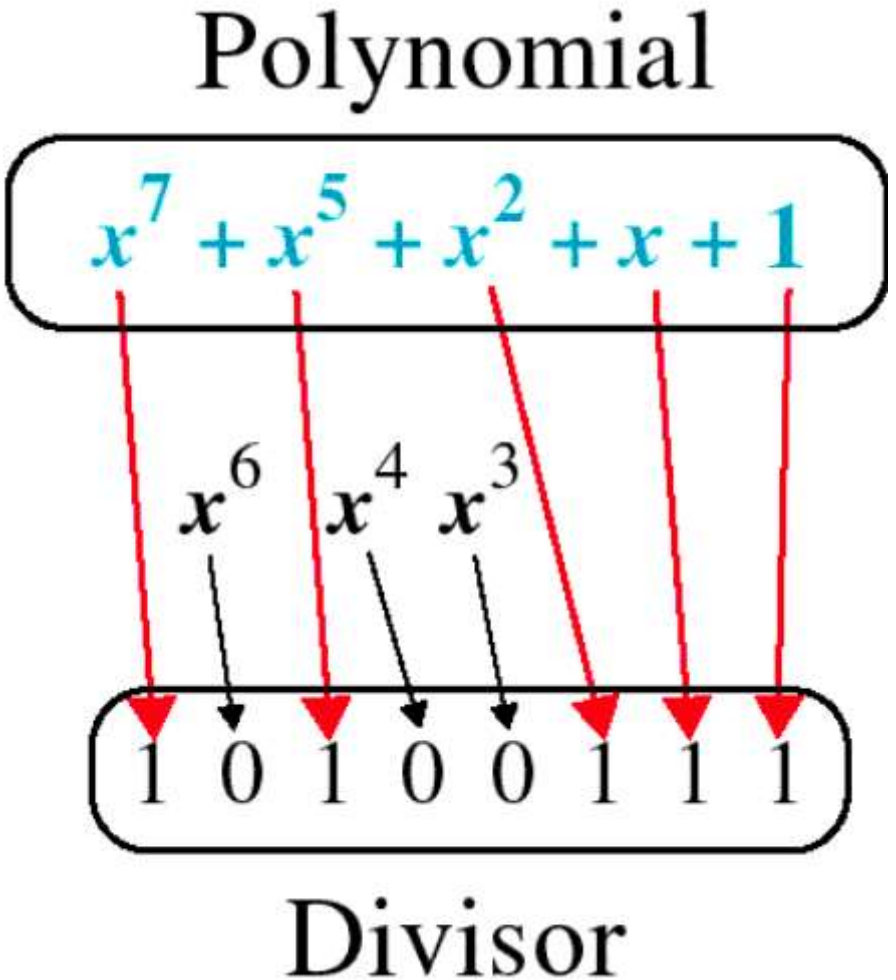
Binary Division



Polynomial

$$x^7 + x^5 + x^2 + x + 1$$

Polynomial and Divisor



Standard Polynomials

CRC-12

$$x^{12} + x^{11} + x^3 + x + 1$$

CRC-16

$$x^{16} + x^{15} + x^2 + 1$$

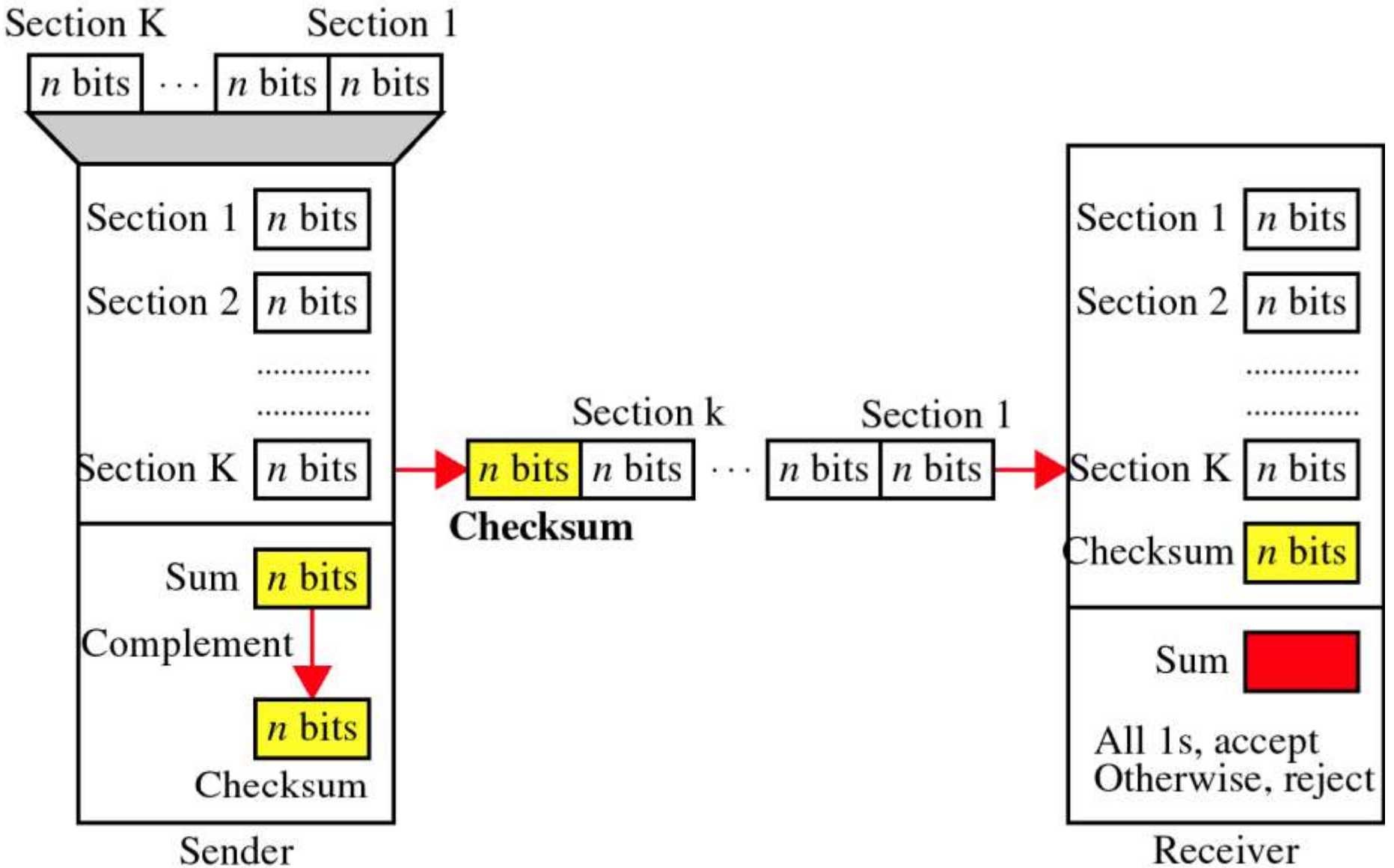
CRC-ITU

$$x^{16} + x^{12} + x^5 + 1$$

CRC-32

$$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$

Checksum


Section K
Section 1
n bits
n bits
n bits
n bits
Section 1
Section 2
Section K
Sum
Complement
Checksum
Sender
Section k
Section 1
Checksum
Receiver
All 1s, accept
Otherwise, reject

At the sender

- The unit is divided into *k* sections, each of *n* bits.
- All sections are added together using one's complement to get the sum.
- The sum is complemented and becomes the checksum.
- The checksum is sent with the data

At the receiver

- The unit is divided into *k* sections, each of *n* bits.
- All sections are added together using one's complement to get the sum.
- The sum is complemented.
- If the result is zero, the data are accepted: otherwise, they are rejected.

Performance

- The checksum detects all errors involving an odd number of bits.
- It detects most errors involving an even number of bits.
- If one or more bits of a segment are damaged and the corresponding bit or bits of opposite value in a second segment are also damaged, the sums of those columns will not change and the receiver will not detect a problem.

Error Correction

It can be handled in two ways:

1) receiver can have the sender retransmit the entire data unit.
2) The receiver can use an error-correcting code, which automatically corrects certain errors.

Single-bit error correction

To correct an error, the receiver reverses the value of the altered bit. To do so, it must know which bit is in error.

Number of redundancy bits needed

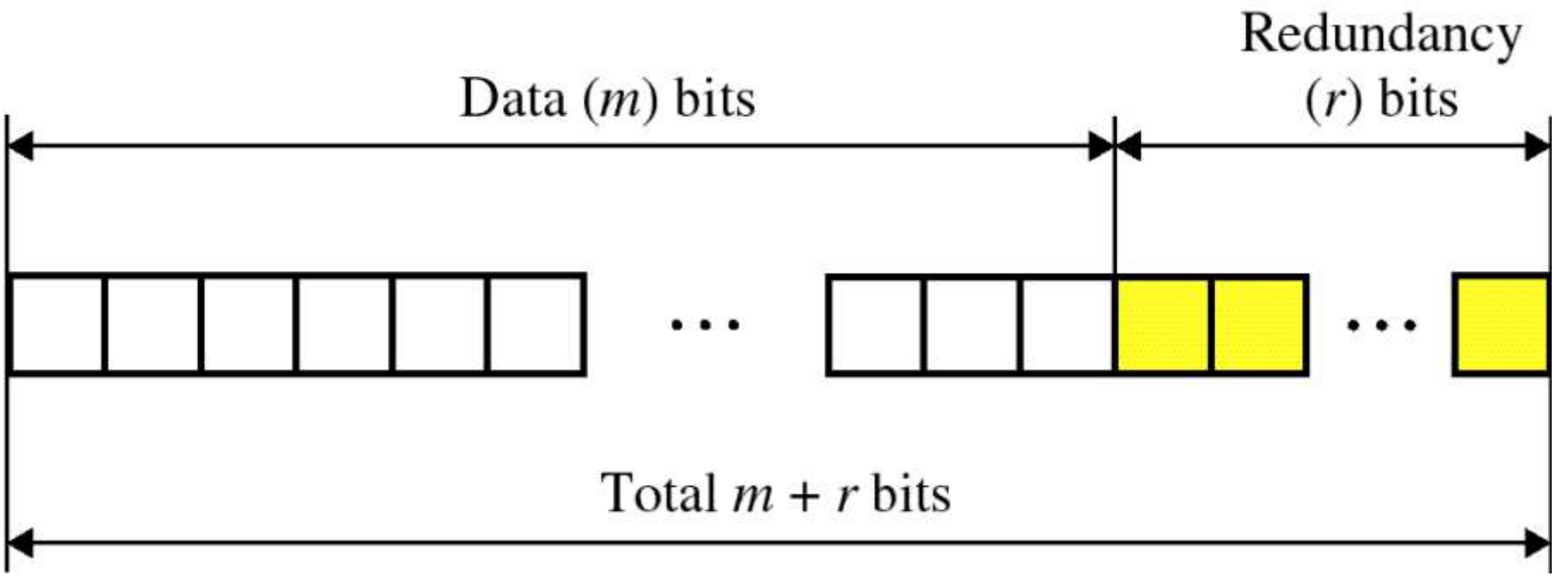
- Let data bits = m
- Redundancy bits = r

$\therefore$ Total message sent = $m+r$

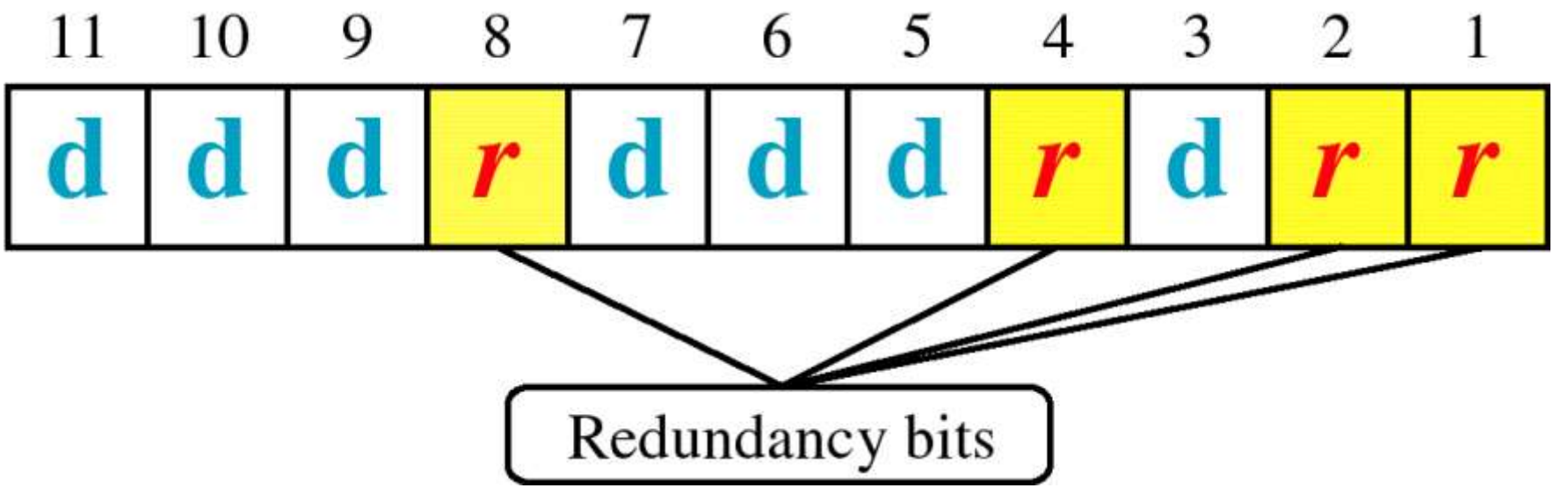
The value of r must satisfy the following relation:

$$2^r \geq m+r+1$$

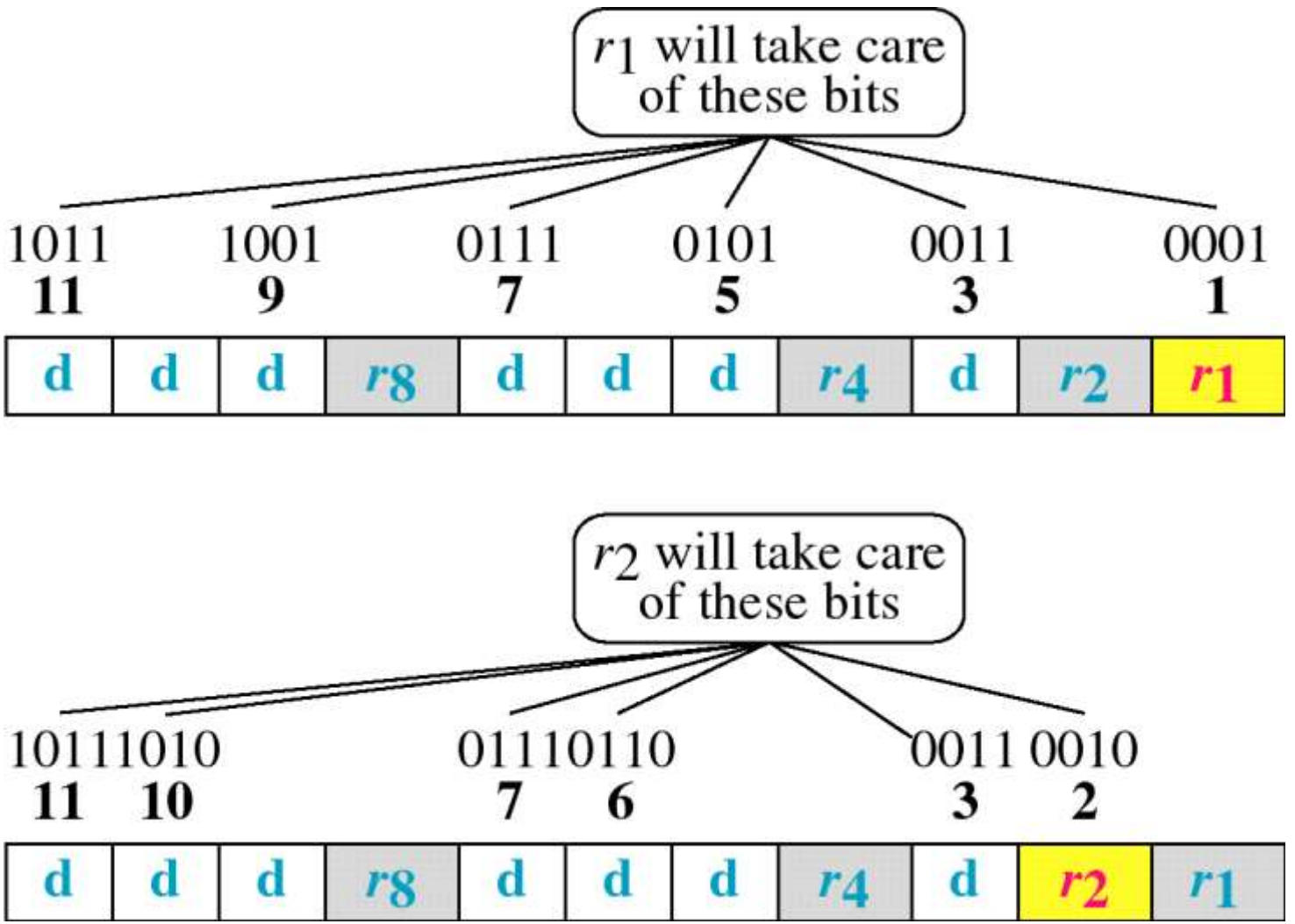
Error Correction



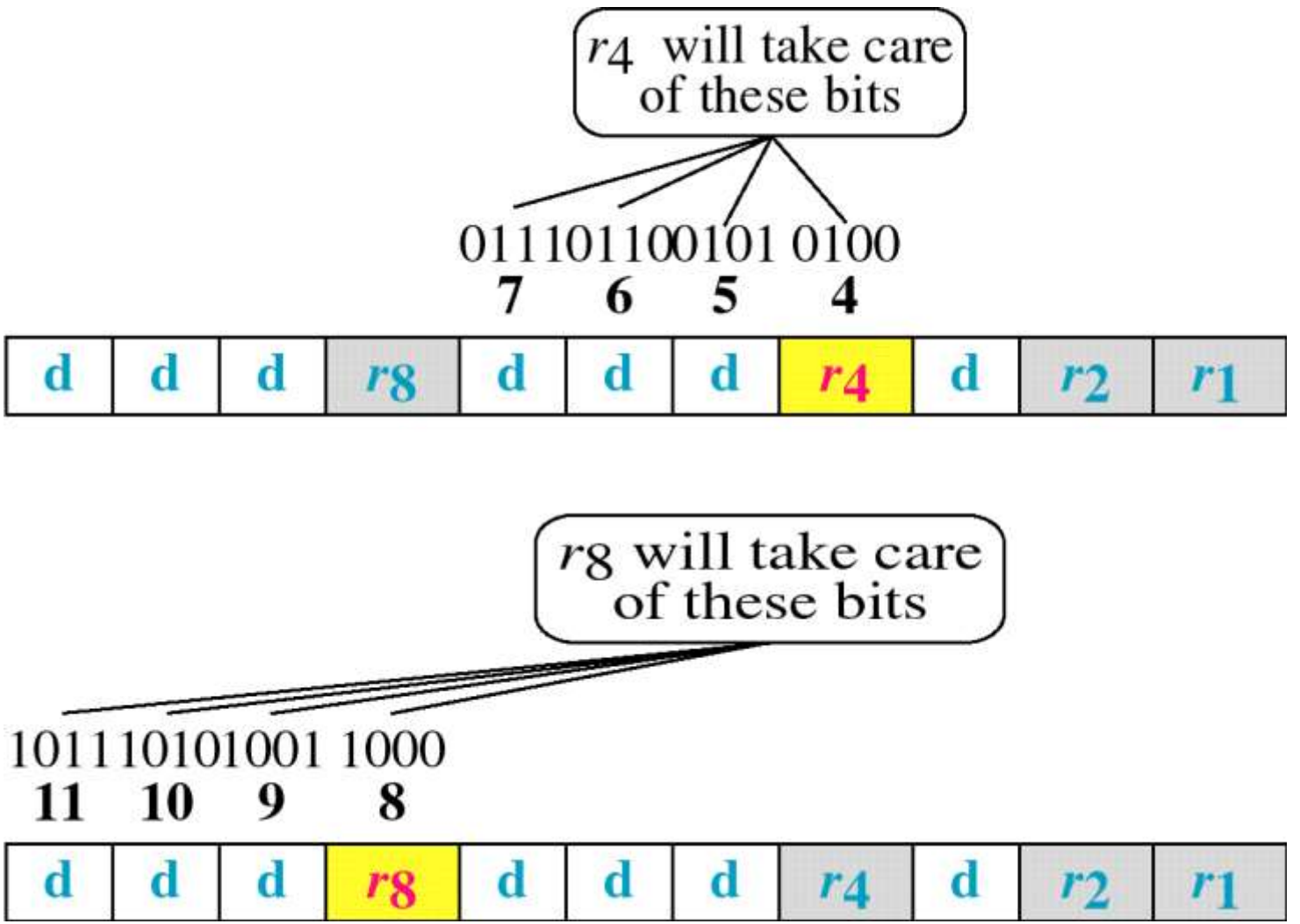
Hamming Code



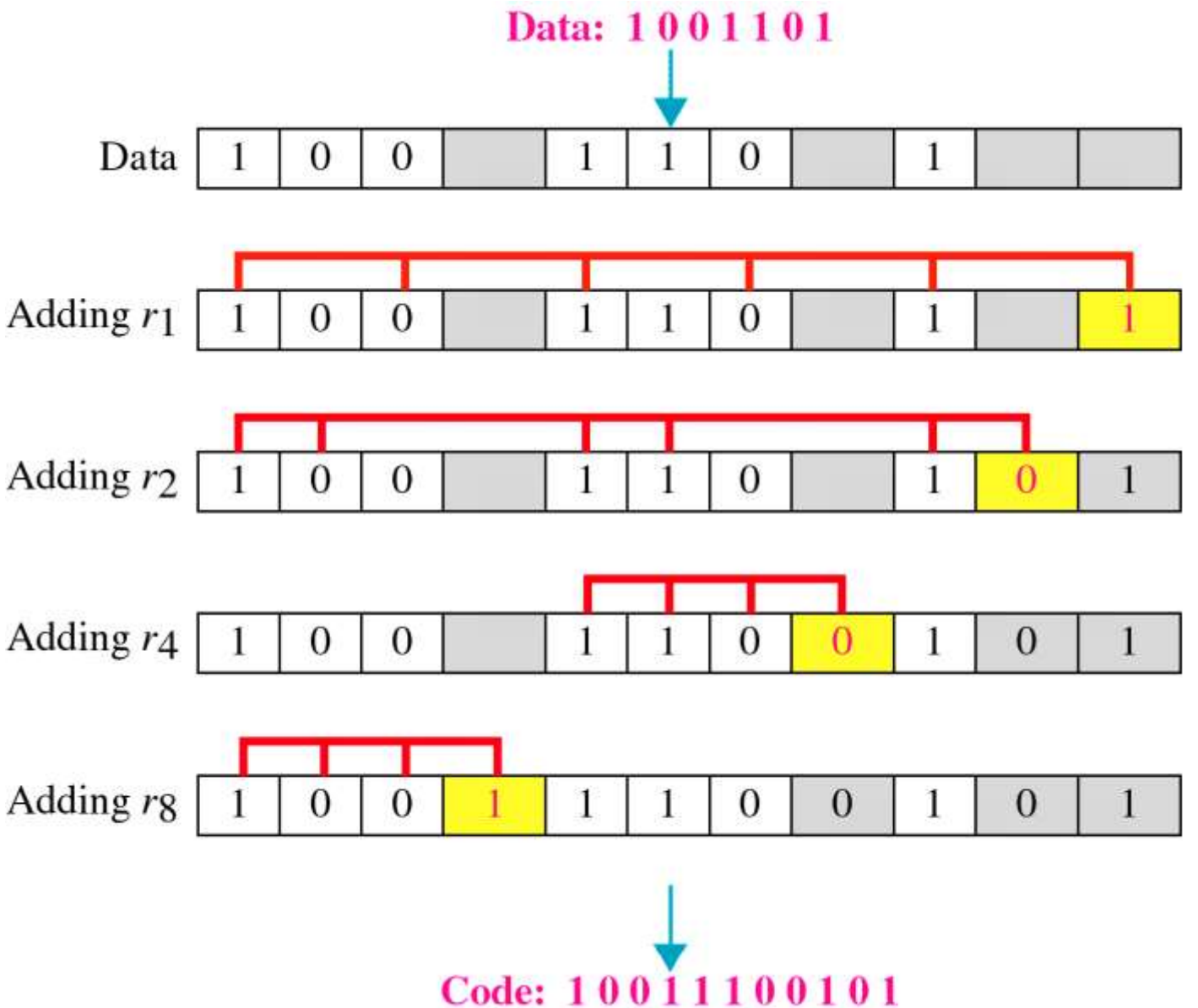
Hamming Code



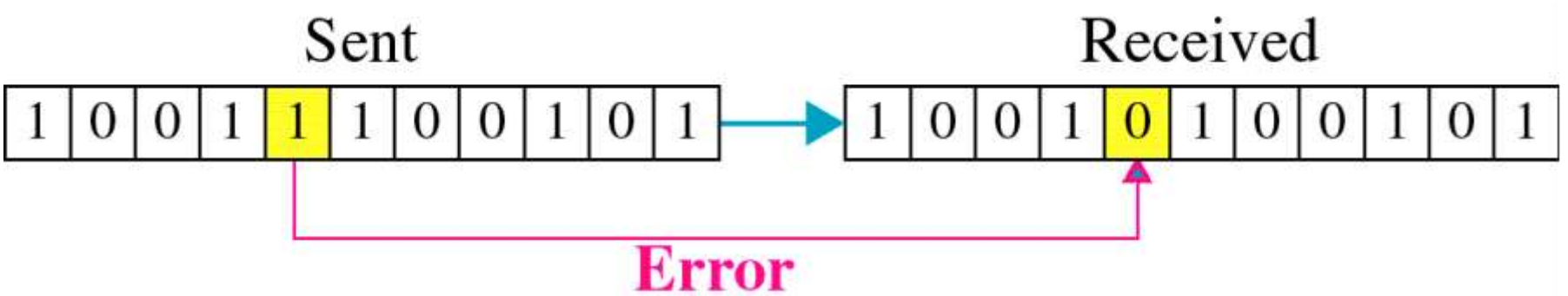
Hamming Code



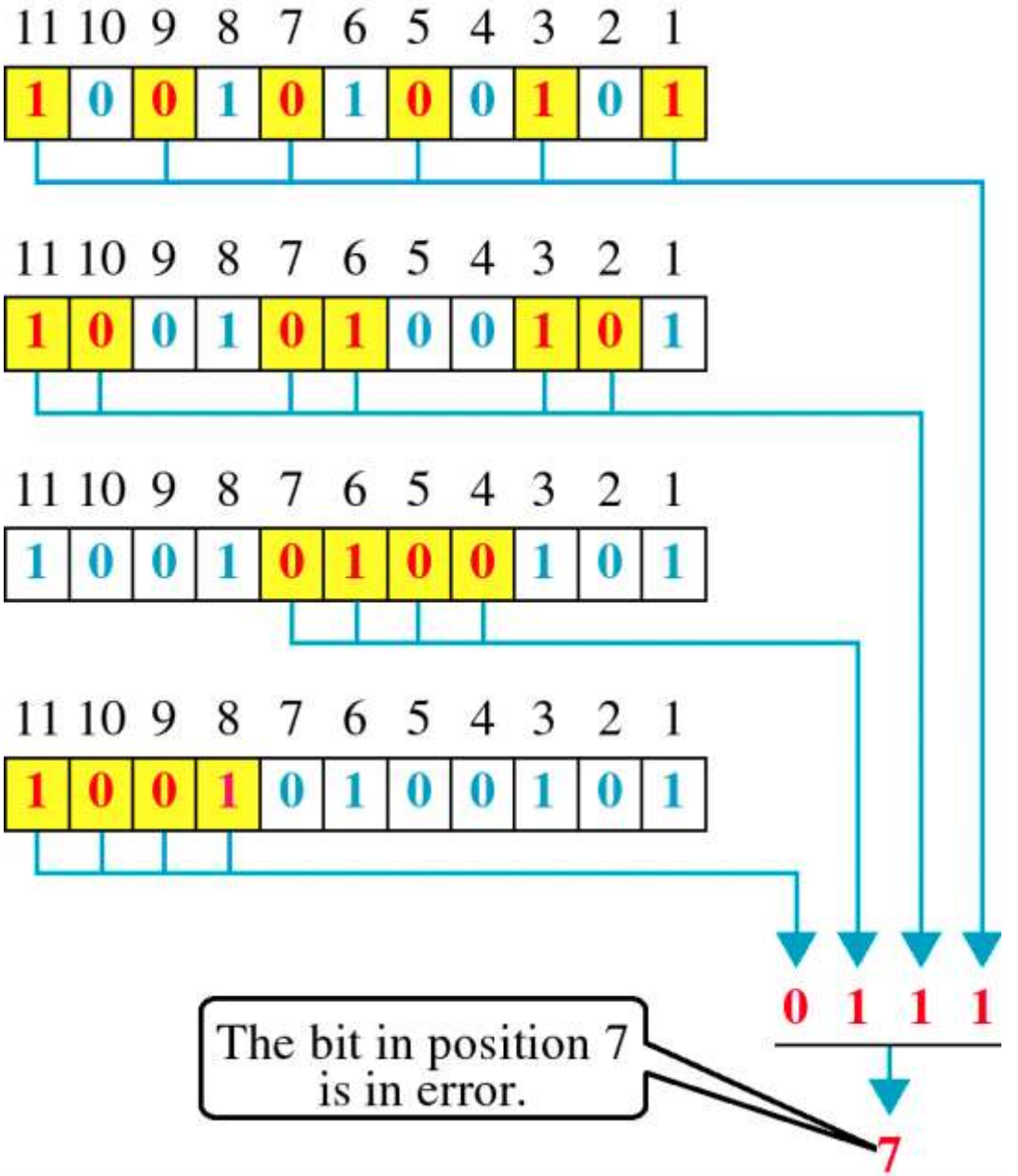
Example of Hamming Code



Single-bit error



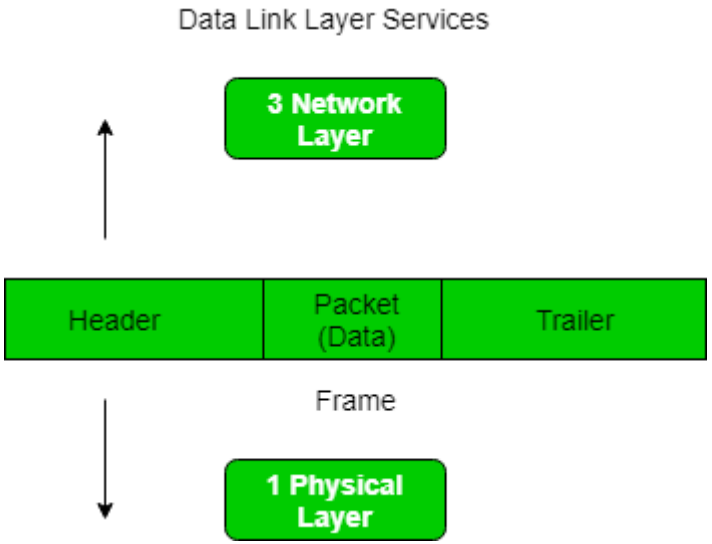
Error Detection



Data Link Layer

Framing

- Framing is a function of the data link layer.



- The advantage of using frames is that data is divided into recoverable chunks that can easily be checked for corruption.

Problems in Framing

- **Detecting start of the frame**

  Station detects frames by looking out for a special sequence of bits that marks the beginning of the frame i.e. ***SFD*** (Starting Frame Delimiter)

Problems in Framing

- **How does the station detect a frame?**

  Every station listens to link for SFD pattern through a sequential circuit. If SFD is detected, sequential circuit alerts station. Station checks destination address to accept or reject frame.

Problems in Framing

- **Detecting end of frame**

  When to stop reading the frame.

Types of framing

1. **Fixed size –** The frame is of fixed size
   no need to provide boundaries to the frame
   (length of the frame :delimiter)

Drawback:

It suffers from internal fragmentation if the data size is less than the frame size

Solution:

Padding

Types of framing

2. Variable size –

need to define the end of the frame as well as the beginning of the next frame.

This can be done in two ways:

- **Length field –** indicate the length of the frame.
  Used in **Ethernet(802.3)**.
  The problem with this is that sometimes the length field might get corrupted.
- **End Delimiter (ED) –** indicate the end of the frame.
  Used in **Token Ring**

The problem with this is that ED can occur in the data

Framing Approaches

There are mainly three types of framing approaches:

- Bit-Oriented Framing
- Byte-Oriented Framing
- Clock Based Framing

Methods of Framing

1. Character Count
2. Flag Byte with Character Stuffing
3. Starting and Ending Flags, with Bit Stuffing
4. Encoding Violations

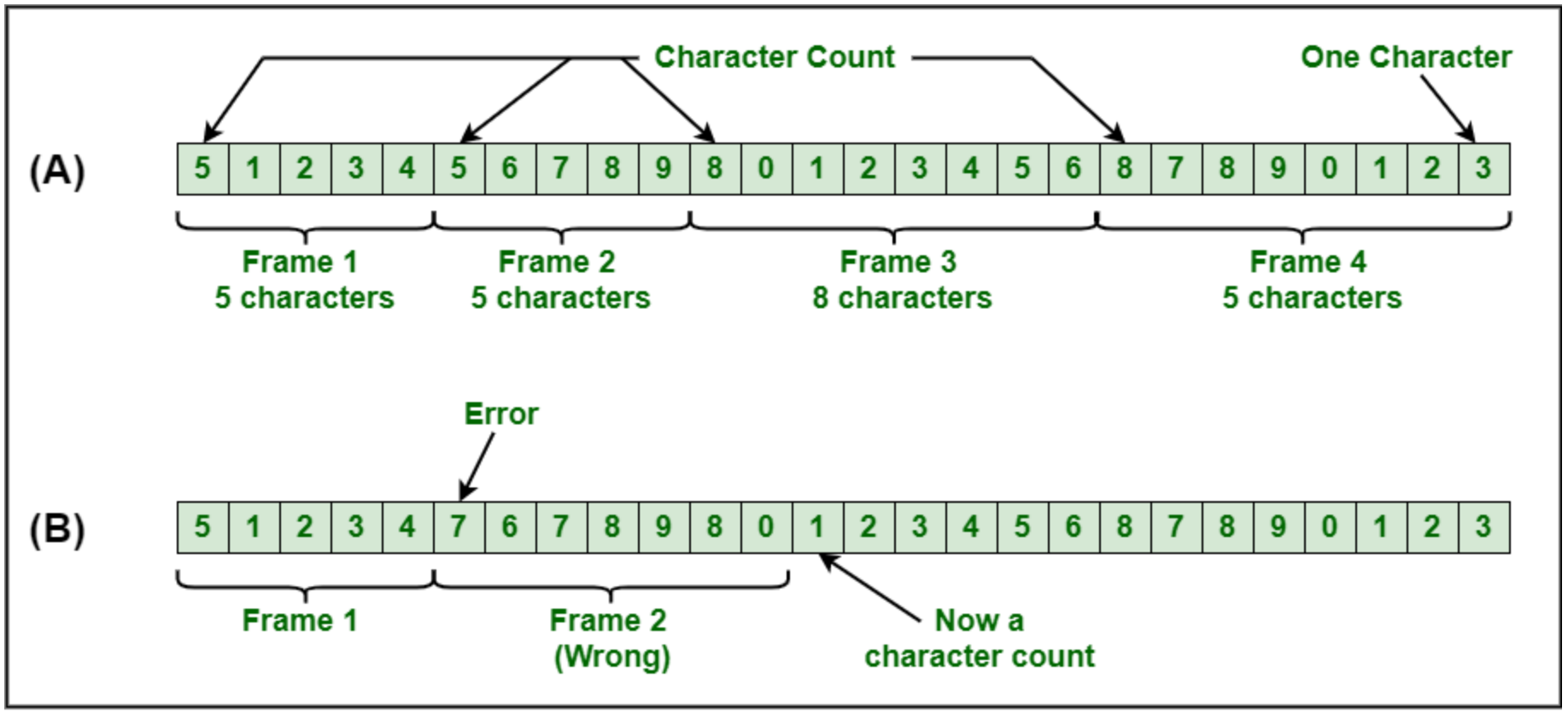
Character Count

- rarely used
- required to count total number of characters that are present in frame.
- This is be done by using field in header.
- Character count method ensures data link layer at the receiver or destination about total number of characters that follow, and about where the frame end.

Character Count

- There is disadvantage also of using this method i.e., if anyhow character count is disturbed or distorted by an error occurring during transmission, then destination or receiver might lose synchronization.
- The destination or receiver might also be not able to locate or identify beginning of next frame.

Character Count



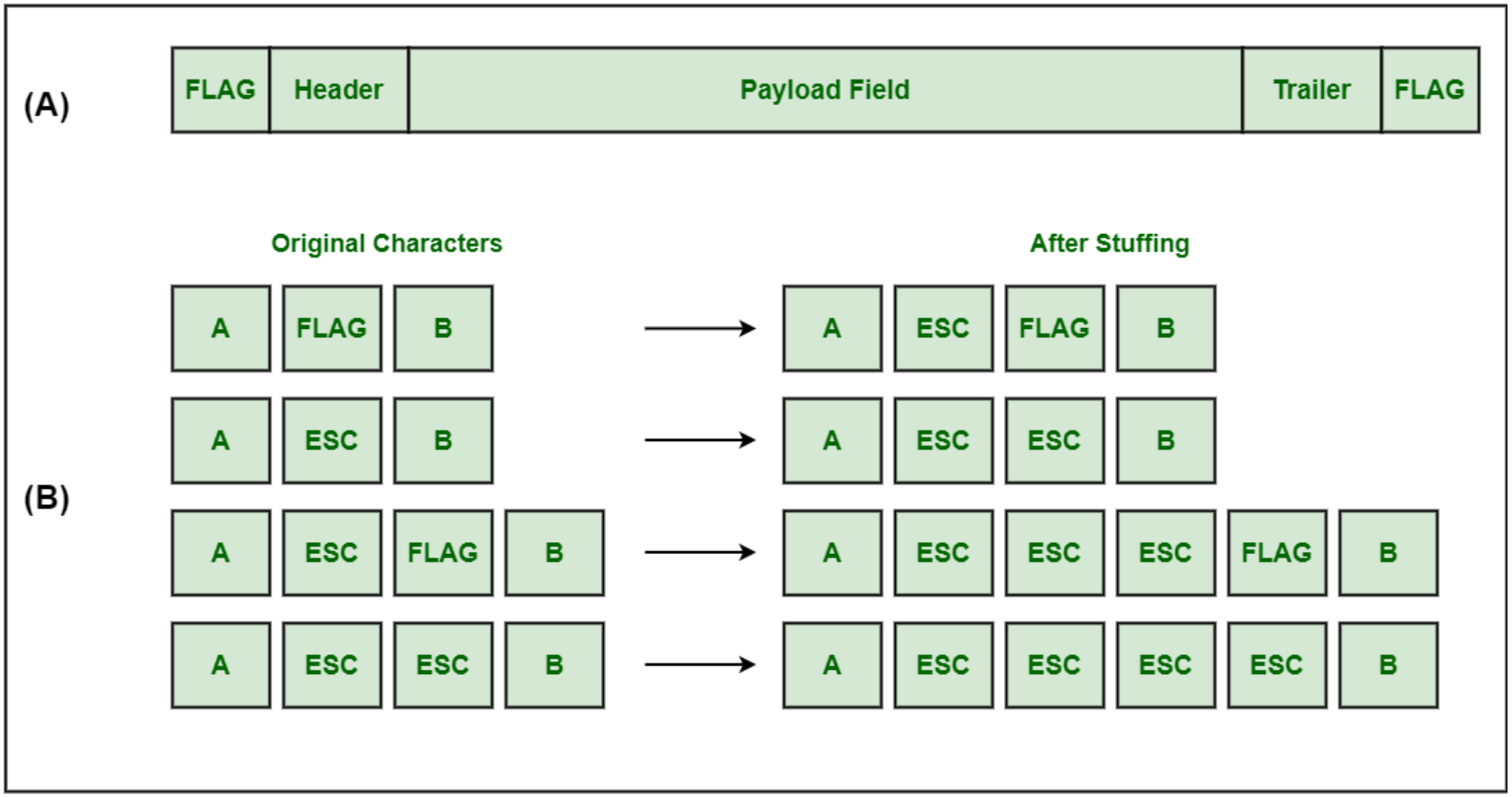
A Character Stream

(A) Without Errors
(B) With one Error

Character Stuffing

- also known as byte stuffing or character-oriented framing
- In byte stuffing, special byte that is basically known as ESC (Escape Character) added to data section of the data stream or frame
- when there is message or character that has same pattern as that of flag byte. But receiver removes this ESC and keeps data part that causes some problems or issues

Character Stuffing



A Character Stuffing

(A) A frame delimited by flag bytes

(B) Four examples of byte sequences before and after byte stuffing

Bit Stuffing

Bit stuffing is the insertion of non information bits into data. Note that stuffed bits should not be confused with overhead bits.

Overhead bits are non-data bits that are necessary for transmission (usually as part of headers, checksums etc.).

Example in upcoming slide

Bit-Oriented Framing

- frames as a collection of bits.
- The data is transmitted as a sequence of bits that can be interpreted as text and multimedia data in the upper layer.

Bit-Oriented Framing

- Consider the frame to be sent and received by the devices.
- The 8 bits added at the start and end of the frame in this protocol are 01111110
- Frame

  011111100101000111111**01111110**

Bit-Oriented Framing

011111100101000111111**01111110**

ie.

011111100101000**01111110****01111110**

In Bit Stuffing, stuff a pattern of bits of arbitrary length in the message to differentiate from the delimiter.

Bit-Oriented Framing

- whenever the sender device finds ***the frame consisting of five consecutive 1's***, **it will stuff a '0' bit.** 01111110010100011111010**01111110**. When the receiving device receives this frame and encounters a '0' after five consecutive bits, it will remove it to maintain the original frame.
- **High-Level Data Link Control (HDLC)** is a data link layer protocol.

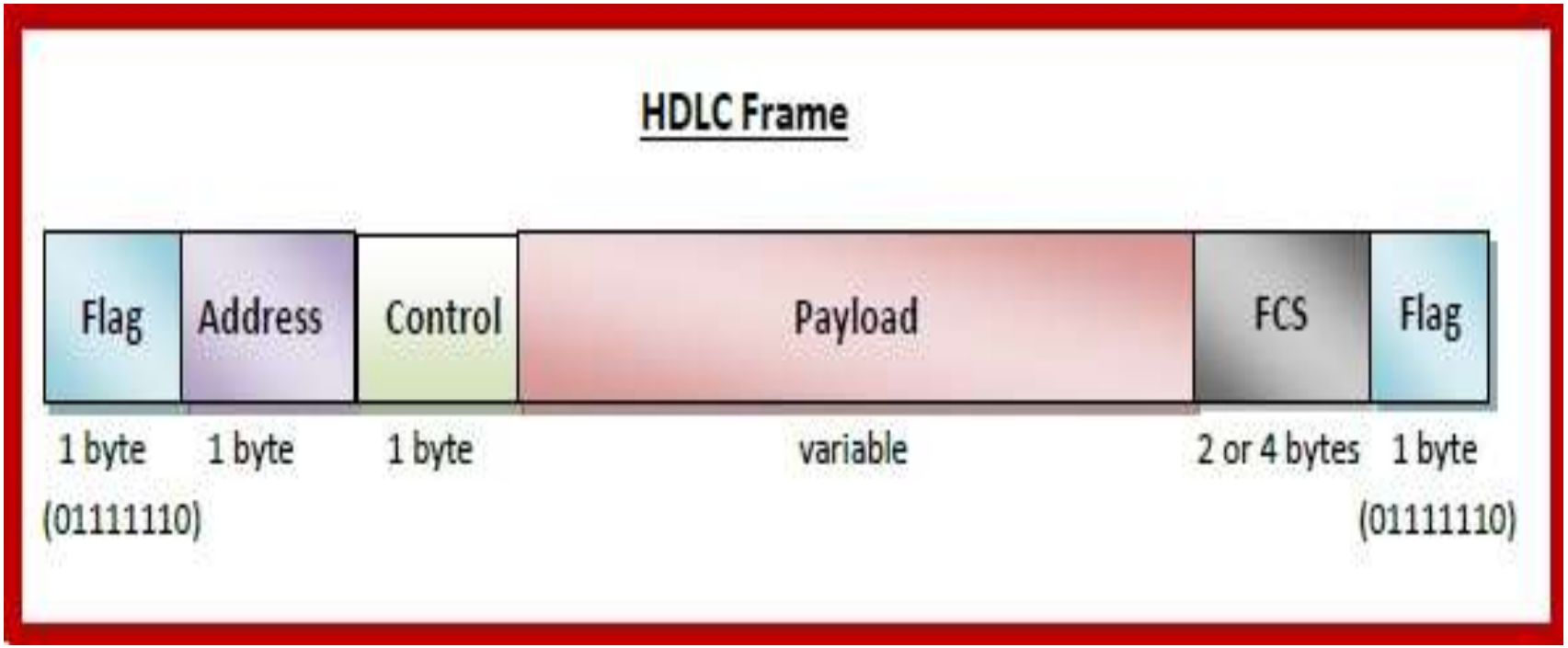
Byte-Oriented Approach

- frames as a collection of bytes(8 bits), also known as a character (Character Oriented Approach)

There are three Byte-Oriented Protocols:

1. Binary Synchronous Communication Protocol (BISYNC)
2. Digital Data Communication Message Protocol (DDCMP)
3. Point-to-Point Protocol(PPP).

HDLC


HDLC Frame
Flag
Address
Control
Payload
FCS
Flag
1 byte
1 byte
1 byte
variable
2 or 4 bytes
1 byte
(01111110)
(01111110)

High-Level Data Link Control (HDLC)

- a bit - oriented protocol
- each frame contains up to six fields.

The fields of a HDLC frame are −

- **Flag** − It is an 8-bit sequence that marks the beginning and the end of the frame. The bit pattern of the flag is 01111110.
- **Address** − It contains the address of the receiver.

  If the frame is sent by the primary station, it contains the address(es) of the secondary station(s).

  If it is sent by the secondary station, it contains the address of the primary station.

  The address field may be from 1 byte to several bytes.

High-Level Data Link Control (HDLC)

- **Control** – It is 1 or 2 bytes containing flow and error control information.
- **Payload** – This carries the data from the network layer. Its length may vary from one network to another.
- **FCS** – It is a 2 byte or 4 bytes frame check sequence for error detection. The standard code used is CRC (cyclic redundancy code)

Types of HDLC Frames

- There are three types of HDLC frames. The type of frame is determined by the control field of the frame.

  1.I-Frame

  2.S-Frame

  3.U-Frame

- **I-frame** –

  Information frames carry user data from the network layer.

  include flow and error control information

  The first bit of control field of I-frame is 0.

- **S-frame** –

  Supervisory frames do not contain information field.

  used for flow and error

  The first two bits of control field of S-frame is 10.

- **U-frame** –

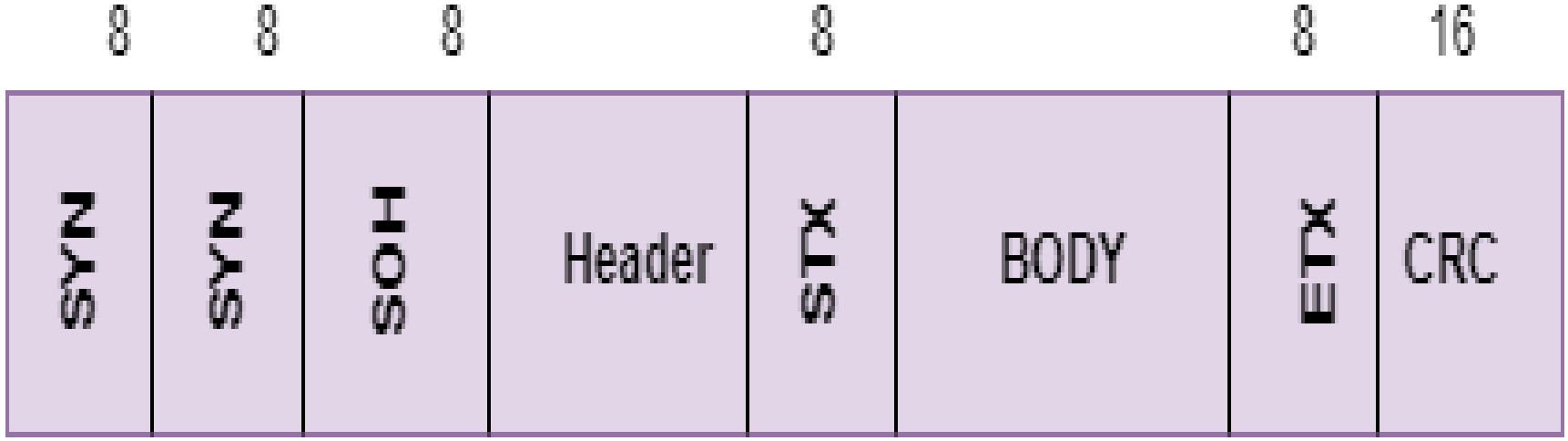
  Un-numbered frames are used for miscellaneous functions, like link management.

  It may contain an information field, if required.

  The first two bits of control field of U-frame is 11.

Binary Synchronous Communication Protocol (BISYNC)

- Also known as Basic Mode Protocol
- It is a sentinel approach.
- frame format



Binary Synchronous Communication Protocol (BISYNC)

- **SYN**: Special starting character,
- **SOH**: Start of the Header,
- **STX**: Start of the text,
- **ETX**: End of the text.

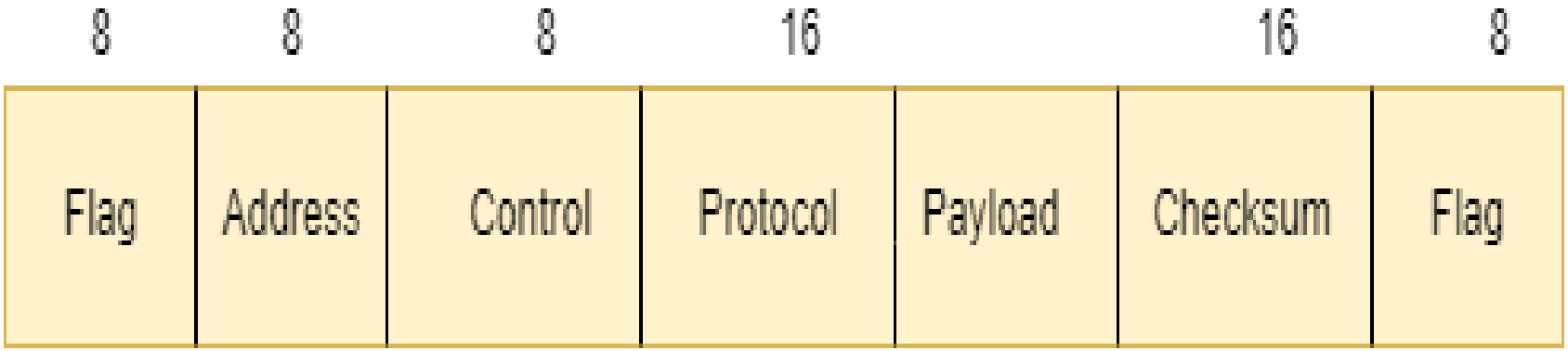
The STX and ETX guard the data part of the portion.

To avoid the framing error problem, Byte Stuffing is used. This is used when the frames consist of characters.

A byte is stuffed in the message to differentiate from the delimiter.

Point-to-Point Protocol(PPP)

- It is a wide area network protocol that runs over internet links.
- mainly used in broadband communication that deals with high speed and heavy loads.
- The frame format

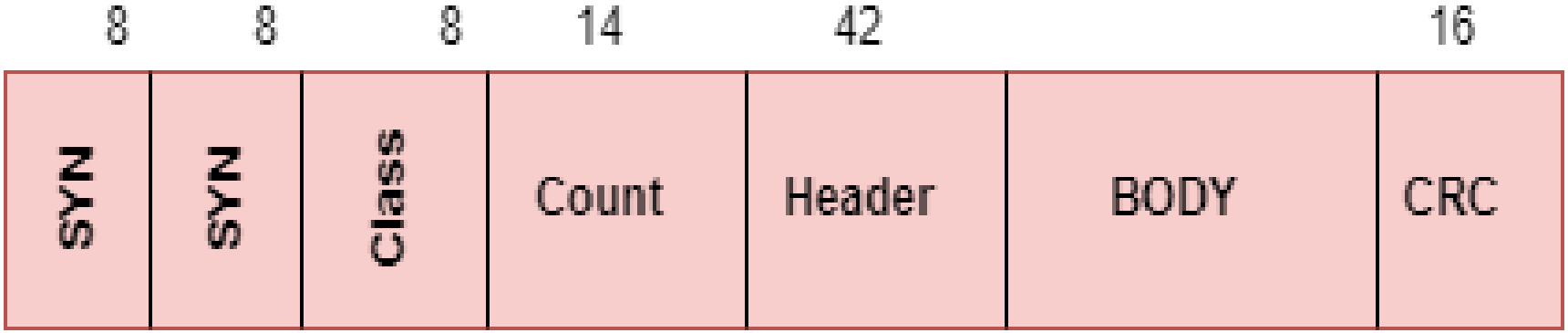


Point-to-Point Protocol(PPP)

- The bit pattern for the flag is 01111110.
- The address field is set to 11111111 in case of broadcast.
- The control value is set to a constant value of 11000000.
- The protocol consists of 1 or 2 bytes that define the type of data in the payload section.
- The maximum length of this field is 1500 bytes.
- The checksum field is used for error detection.
- if the flag bits appear in the payload part, the situation is overcome using character/byte stuffing.

Digital Data Communication Message Protocol (DDCMP)

- A new count field is introduced in this protocol.
- The frame format



Digital Data Communication Message Protocol (DDCMP)

- if transmission error corrupts the count field, then the end of the frame will not be detected by the receiver correctly.

Clock Based Framing

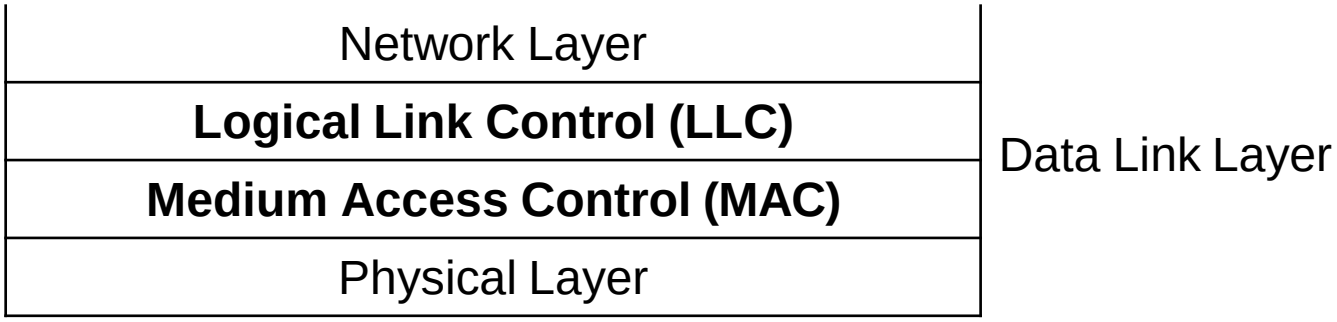
- mainly used for Optical Networks such as SONET.
- In this approach, a series of repetitive pulses maintain a constant bit rate and keep the digital bits aligned in the data stream.

Medium Access Control Sublayer

Introduction

- In broadcast networks, several stations share a single communication channel.
- The major issue in these networks is, which station should transmit data at a given time.
- This process of deciding the turn of different stations is known as **Channel Allocation**.
- To coordinate the access to the channel, **multiple access** protocols are required.
- All these protocols belong to the MAC sublayer.

Introduction



- Data Link layer is divided into two sublayers:
  - Logical Link Control (LLC)
  - Medium Access Control (MAC)
- **LCC** is responsible for error control & flow control.
- **MAC** is responsible for multiple access resolutions.

Channel Allocation Problem

- In broadcast networks, single channel is shared by several stations.
- This channel can be allocated to only one transmitting user at a time.
- There are two different methods of channel allocations:
  - Static Channel Allocation
  - Dynamic Channel Allocation

Static Channel Allocations

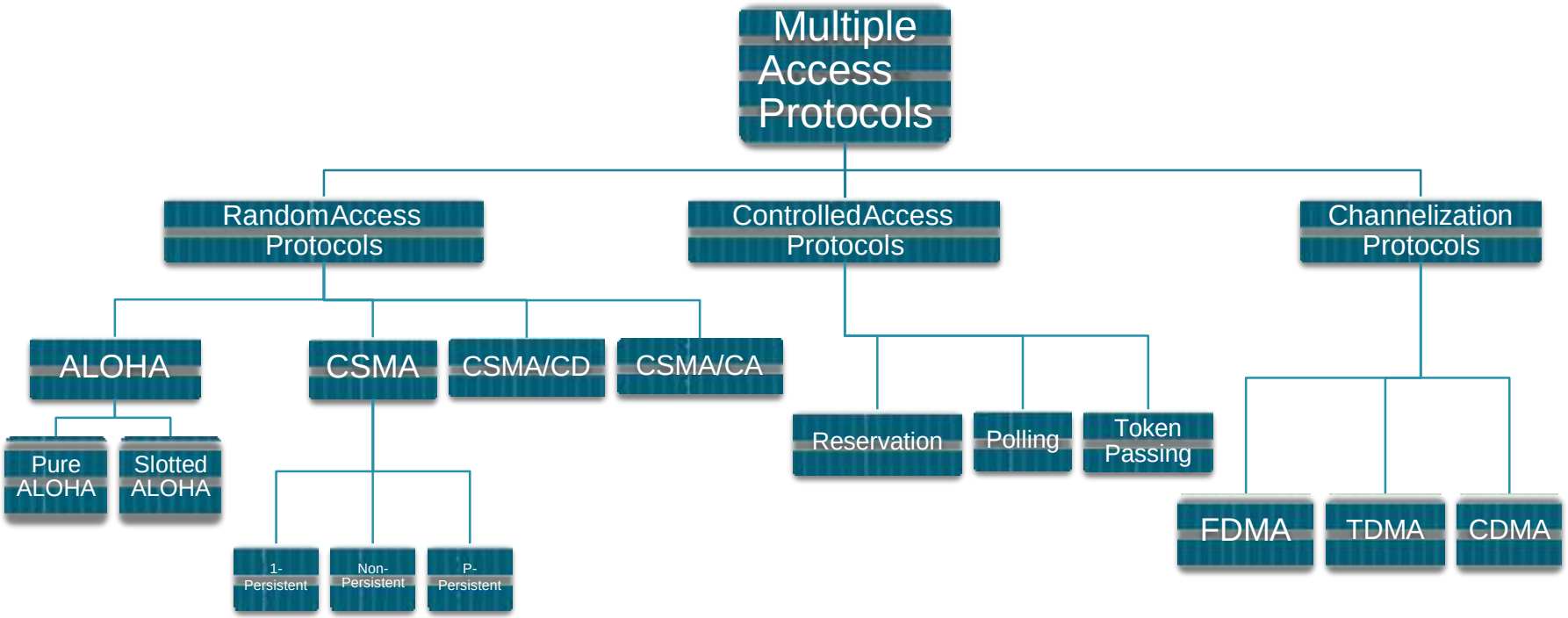
- In this method, a single channel is divided among various users either on the basis of frequency or on the basis of time.
- It either uses FDM (Frequency Division Multiplexing) or TDM (Time Division Multiplexing).
- In FDM, fixed frequency is assigned to each user, whereas, in TDM, fixed time slot is assigned to each user.

Dynamic Channel Allocation

- In this method, no user is assigned fixed frequency or fixed time slot.
- All users are dynamically assigned frequency or time slot, depending upon the requirements of the user.

Multiple Access Protocols

- Many protocols have been defined to handle the access to shared link.
- These protocols are organized in three different groups.:
  - Random Access Protocols
  - Controlled Access Protocols
  - Channelization Protocols


Multiple Access Protocols
Random Access Protocols
Controlled Access Protocols
Channelization Protocols
ALOHA
CSMA
CSMA/CD
CSMA/CA
Pure ALOHA
Slotted ALOHA
1-Persistent
Non-Persistent
P-Persistent
Reservation
Polling
Token Passing
FDMA
TDMA
CDMA

Random Access Protocols

- It is also called **Contention Method**.
- In this method, there is no control station.
- Any station can send the data.
- The station can make a decision on whether or not to send data. This decision depends on the state of the channel, i.e. channel is busy or idle.
- There is no scheduled time for a stations to transmit. They can transmit in random order.

Random Access Protocols

- There is no rule that decides which station should send next.
- If two stations transmit at the same time, there is collision and the frames are lost.
- The various random access methods are:
  - ALOHA
  - CSMA (Carrier Sense Multiple Access)
  - CSMA/CD (Carrier Sense Multiple Access with Collision Detection)
  - CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance)

ALOHA

- ALOHA was developed at University of Hawaii in early 1970s by Norman Abramson.
- It was used for ground based radio broadcasting.
- In this method, stations share a common channel.
- When two stations transmit simultaneously, collision occurs and frames are lost.
- There are two different versions of ALOHA:
  - Pure ALOHA
  - Slotted ALOHA

Pure ALOHA

- In pure ALOHA, stations transmit frames whenever they have data to send.
- When two stations transmit simultaneously, there is collision and frames are lost.
- In pure ALOHA, whenever any station transmits a frame, it expects an acknowledgement from the receiver.
- If acknowledgement is not received within specified time, the station assumes that the frame has been lost.

Pure ALOHA

- If the frame is lost, station waits for a random amount of time and sends it again.
- This waiting time must be random, otherwise, same frames will collide again and again.
- Whenever two frames try to occupy the channel at the same time, there will be collision and both the frames will be lost.
- If first bit of a new frame overlaps with the last bit of a frame almost finished, both frames will be lost and both will have to be retransmitted.

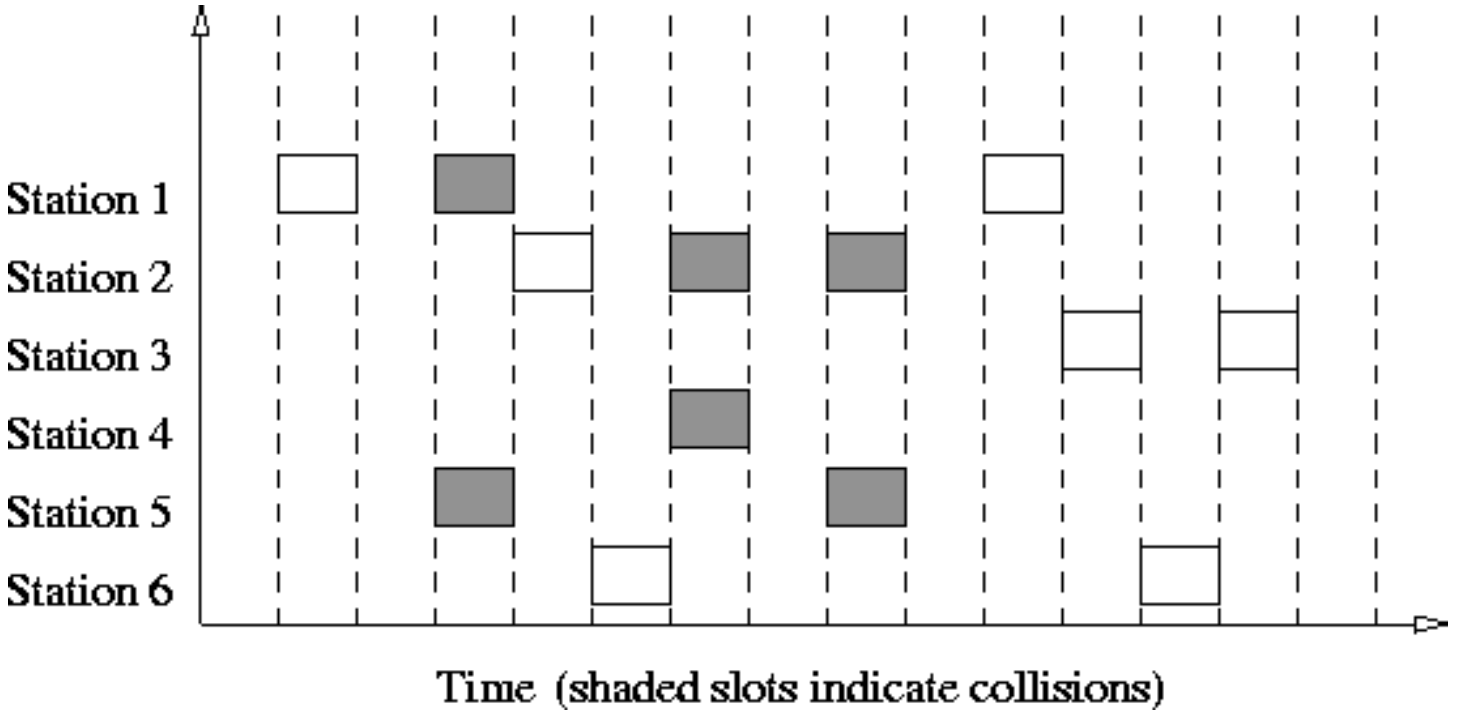
Pure ALOHA


Station 1
Station 2
Station 3
Station 4
Station 5
Station 6
Time (shaded slots indicate collisions)

Slotted ALOHA

- Slotted ALOHA was invented to improve the efficiency of pure ALOHA.
- In slotted ALOHA, time of the channel is divided into intervals called slots.
- The station can send a frame only at the beginning of the slot and only one frame is sent in each slot.
- If any station is not able to place the frame onto the channel at the beginning of the slot, it has to wait until the next time slot.
- There is still a possibility of collision if two stations try to send at the beginning of the same time slot.

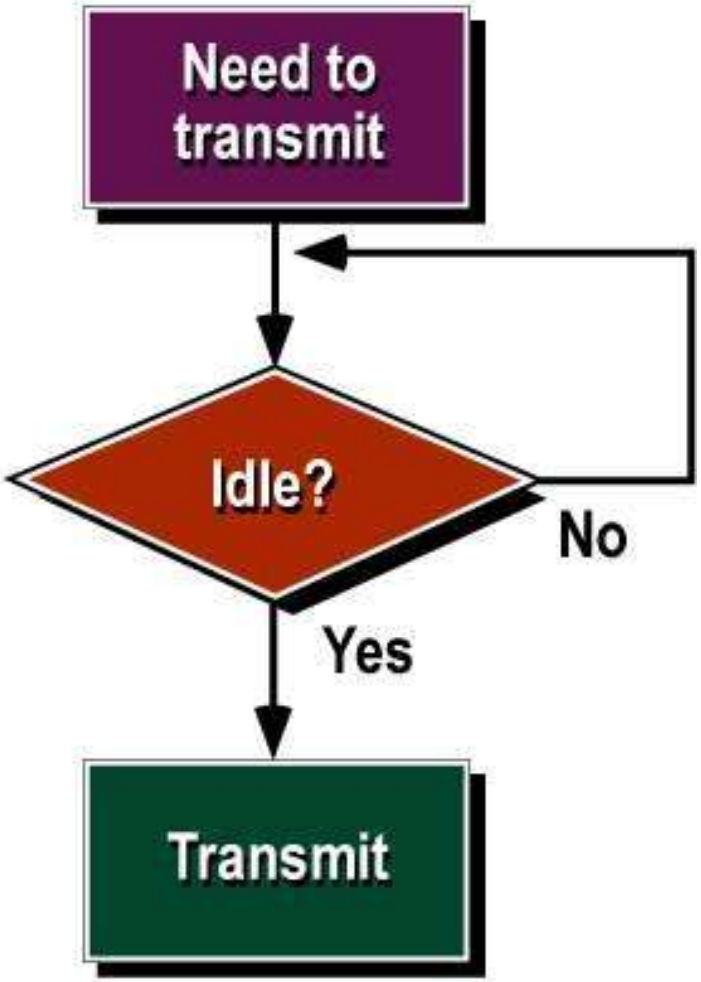
Slotted ALOHA



Carrier Sense Multiple Access (CSMA)

- CSMA was developed to overcome the problems of ALOHA i.e. to minimize the chances of collision.
- CSMA is based on the principle of “carrier sense”.
- The station sense the carrier or channel before transmitting a frame.
- It means the station checks whether the channel is idle or busy.
- The chances of collision reduces to a great extent if a station checks the channel before trying to use it.

Carrier Sense Multiple Access (CSMA)



Carrier Sense Multiple Access (CSMA)

- The chances of collision still exists because of propagation delay.
- The frame transmitted by one station takes some time to reach the other station.
- In the meantime, other station may sense the channel to be idle and transmit its frames.
- This results in the collision.

Carrier Sense Multiple Access (CSMA)

- There are three different types of CSMA protocols:
  - 1-Persistent CSMA
  - Non-Persistent CSMA
  - P-Persistent CSMA

1-Persistent CSMA

- In this method, station that wants to transmit data, continuously senses the channel to check whether he channel is idle or busy.
- If the channel is busy, station waits until it becomes idle.
- When the station detects an idle channel, it immediately transmits the frame.
- This method has the highest chance of collision because two or more stations may find channel to be idle at the same time and transmit their frames.

Non-Persistent CSMA

- A station that has a frame to send, senses the channel.
- If the channel is idle, it sends immediately.
- If the channel is busy, it waits a random amount of time and then senses the channel again.
- It reduces the chance of collision because the stations wait for a random amount of time .
- It is unlikely that two or more stations will wait for the same amount of time and will retransmit at the same time.

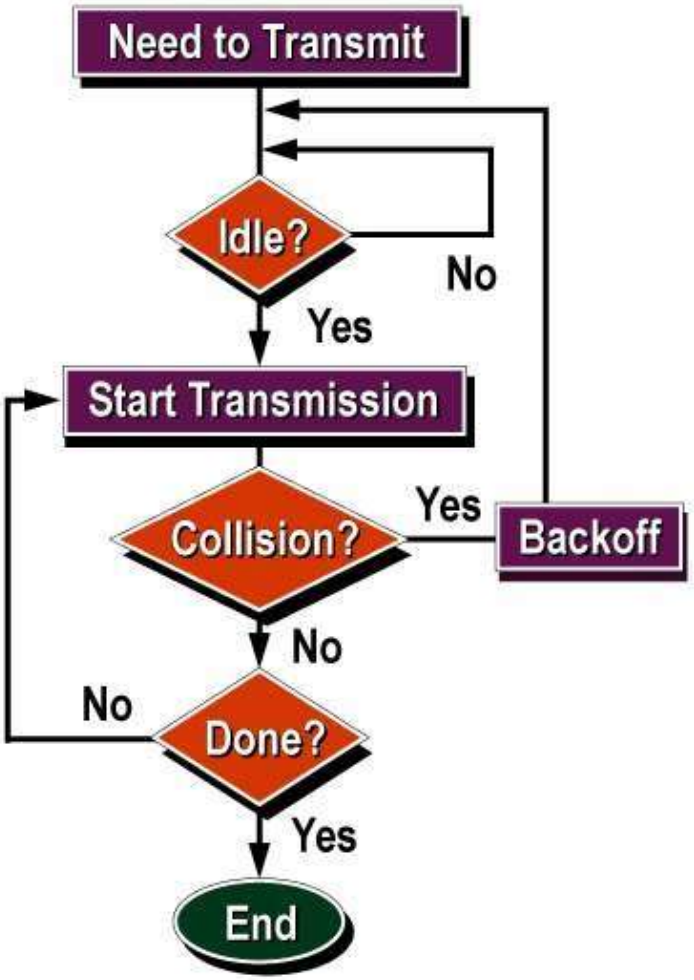
P-Persistent CSMA

- In this method, the channel has time slots such that the time slot duration is equal to or greater than the maximum propagation delay time.
- When a station is ready to send, it senses the channel.
- If the channel is busy, station waits until next slot.
- If the channel is idle, it transmits the frame.
- It reduces the chance of collision and improves the efficiency of the network.

CSMA with Collision Detection (CSMA/CD)

- In this protocol, the station senses the channel before transmitting the frame. If the channel is busy, the station waits.
- Additional feature in CSMA/CD is that the stations can detect collisions.
- The stations abort their transmission as soon as they detect collision.
- This feature is not present in CSMA.
- The stations continue to transmit even though they find that collision has occurred.

CSMA with Collision Detection (CSMA/CD)



CSMA with Collision Detection (CSMA/CD)

- In CSMA/CD, the station that sends its data on the channel, continues to sense the channel even after data transmission.
- If collision is detected, the station aborts its transmission and waits for a random amount of time & sends its data again.
- As soon as a collision is detected, the transmitting station release a ***jam*** signal.
- Jam signal alerts other stations. Stations are not supposed to transmit immediately after the collision has occurred.

CSMA with Collision Avoidance (CSMA/CA)

- This protocol is used in wireless networks because they cannot detect the collision.
- So, the only solution is collision avoidance.
- It avoids the collision by using three basic techniques:
  - Interframe Space
  - Contention Window
  - Acknowledgements

CSMA with Collision Avoidance (CSMA/CA)

Interframe Space

- Whenever the channel is found idle, the station does not transmit immediately.
- It waits for a period of time called Interframe Space (IFS).
- When channel is sensed idle, it may be possible that some distant station may have already started transmitting.
- Therefore, the purpose of IFS time is to allow this transmitted signal to reach its destination.
- If after this IFS time, channel is still idle, the station can send the frames.

Contention Window

- Contention window is the amount of time divided into slots.
- Station that is ready to send chooses a random number of slots as its waiting time.
- The number of slots in the window changes with time.
- It means that it is set of one slot for the first time, and then doubles each time the station cannot detect an idle channel after the IFS time.
- In contention window, the station needs to sense the channel after each time slot.

Acknowledgment

- Despite all the precautions, collisions may occur and destroy the data.
- Positive acknowledgement and the time-out timer helps guarantee that the receiver has received the frame.

Controlled Access Protocol

- In this method, the stations consult each other to find which station has a right to send.
- A station cannot send unless it has been authorized by other station.
- The different controlled access methods are:
  - Reservation
  - Polling
  - Token Passing

Reservation

- In this method, a station needs to make a reservation before sending data.
- The time is divided into intervals. In each interval, a reservation frame precedes the data frames sent in that interval.
- If there are N stations, then there are exactly N reservation slots in the reservation frame.
- Each slot belongs to a station.
- When a station needs to send a frame, it makes a reservation in its own slot.
- The stations that have made reservations can send their frames after the reservation frame.

Polling

- Polling method works in those networks where primary and secondary stations exist.
- All data exchanges are made through primary device even when the final destination is a secondary device.
- Primary device controls the link and secondary device follow the instructions.

Token Passing

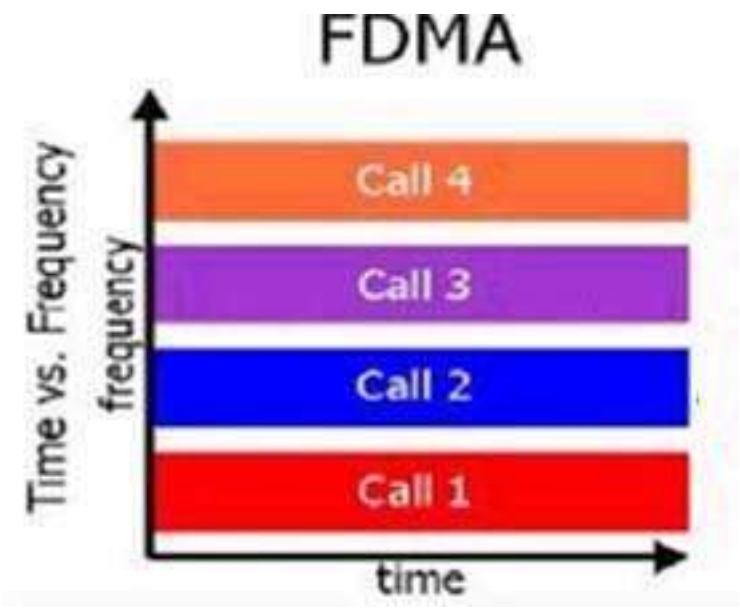
- Token passing method is used in those networks where the stations are organized in a logical ring.
- In such networks, a special packet called token is circulated through the ring.
- Station that possesses the token has the right to access the channel.
- Whenever any station has some data to send, it waits for the token. It transmits data only after it gets the possession of token.
- After transmitting the data, the station releases the token and passes it to the next station in th ring.
- If any station that receives the token has no data to send, it simply passes the token to the next station in the ring.

Channelization Protocol

- Channelization is a multiple access method in which the available bandwidth of a link is shared in ***time***, ***frequency*** or ***code*** between different stations.
- There are three basic channelization protocols:
  - Frequency Division Multiple Access (FDMA)
  - Time Division Multiple Access (TDMA)
  - Code Division Multiple Access (CDMA)

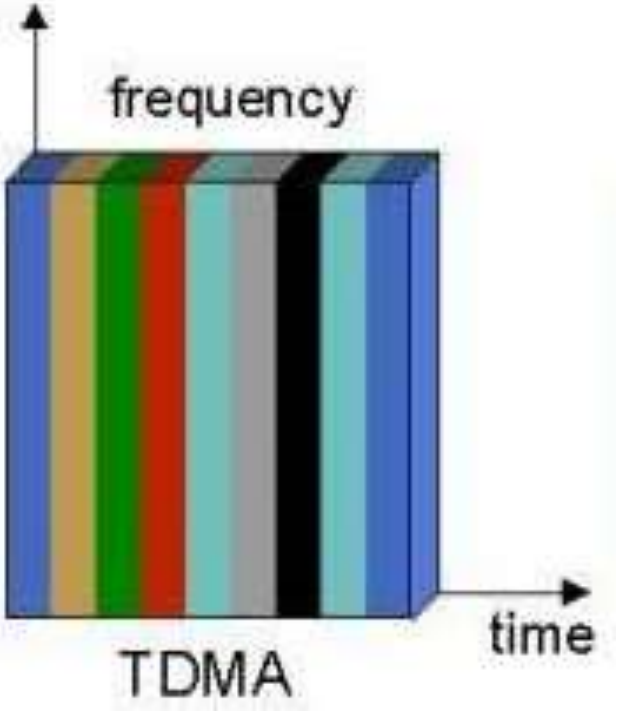
FDMA

- In FDMA, the available bandwidth is divided into frequency bands.
- Each station is allocated a band to send its data.
- This band is reserved for that station for all the time.
- The frequency bands of different stations are separated by small bands of unused frequency.
- These unused bands are called ***guard bands*** that prevent station interferences.
- FDMA is different from FDM (Frequency Division Multiplexing).
- FDM is a physical layer technique, whereas, FDMA is an access method in the data link layer.


FDMA
Time vs. Frequency
frequency
Call 4
Call 3
Call 2
Call 1
time

TDMA

- In TDMA, the bandwidth of channel is divided among various stations on the basis of time.
- Each station is allocated a time slot during which it can send its data.
- Each station must know the beginning of its time slot.
- TDMA requires synchronization between different stations.
- Synchronization is achieved by using some synchronization bits at the beginning of each slot.
- TDMA is also different from TDM. TDM is a physical layer technique, whereas, TDMA is an access method in data link layer.


frequency
time
TDMA

CDMA

- Unlike TDMA, in CDMA all stations can transmit data simultaneously.
- CDMA allows each station to transmit over the entire frequency spectrum all the time.
- Multiple simultaneous transmissions are separated using coding theory.
- In CDMA, each user is given a unique code sequence.

Working of CDMA

- Let us assume that we have four stations: 1, 2, 3 and 4 that are connected to the same channel.
- The data from station 1 is d_1, from station 2 is d_2 and so on.
- The code assigned to station 1 is c_1, station 2 is c_2 and so on.
- These assigned codes have two properties:
  - If we multiply each code by another, we get 0.
  - If we multiply each code by itself, we get 4, (no. of stations).

Working of CDMA

- When these four stations send data on the same channel, then station 1 multiplies its data by its code i.e. $d_1.c_1$, station 2 multiplies its data by its code i.e. $d_2.c_2$ and so on.
- The data that goes on the channel is the sum of all these terms:

$$\mathbf{d_1.c_1 + d_2.c_2 + d_3.c_3 + d_4.c_4}$$

- Any station that wants to receive data from the channel multiplies the data on the channel by the code of the sender.

Working of CDMA

- For e.g.: suppose station 2 wants to receive data from station1.
- It multiplies the data on the channel by c_1, (code of station 1).
- Because $(c_1.c_1)$ is 4, but $(c_2.c_1)$, $(c_3.c_1)$ and $(c_4.c_1)$ are all 0s, station 2 divides the result by 4 to get the data from station 1.

$$\text{data} = (d_1.c_1 + d_2.c_2 + d_3.c_3 + d_4.c_4).c1$$

$$= d_1.c_1.c_1 + d_2.c_2.c_1 + d_3.c_3.c_1 + d_4.c_4.c_1$$

$$= d_1.4 + 0 + 0 + 0$$

$$= (d_1.4) / 4 = d_1$$

Working of CDMA

- The code assigned to each station is a sequence of numbers called chips.
- These chips are called orthogonal sequences.
- Each sequence is made of N elements, where N is the number of stations.

c_1	c_2	c_3	c_4
[+1 +1 +1 +1]	[+1 -1 +1 -1]	[+1 +1 -1 -1]	[+1 -1 -1 +1]

Working of CDMA

c_1	c_2	c_3	c_4
[+1 +1 +1 +1]	[+1 -1 +1 -1]	[+1 +1 -1 -1]	[+1 -1 -1 +1]

- This sequence has following properties:
  - If we multiply two equal sequences, element by element, and add the result, we get N, where N is the number of elements in the sequence.
  - This is called ***inner product of two equal sequence***.

[+1 +1 -1 -1] . [+1 +1 -1 -1] = 1 + 1 + 1 + 1 = 4

Working of CDMA

c_1	c_2	c_3	c_4
[+1 +1 +1 +1]	[+1 -1 +1 -1]	[+1 +1 -1 -1]	[+1 -1 -1 +1]

- If we multiply two different sequences, element by element, and add the result, we get 0.
- This is called ***inner product of two different sequence***.

[+1 +1 -1 -1] . [+1 +1 +1 +1] = 1 + 1 - 1 - 1 = 0

Data Link Layer

Sliding Window Protocol

Sliding Window Protocol

- Data link layer protocols for reliable and sequential delivery of data frames.
- The sliding window is also used in **Transmission Control Protocol.**
- In this protocol, multiple frames can be sent by a sender at a time before receiving an acknowledgment from the receiver.
- The term sliding window refers to the imaginary boxes to hold frames.
- Sliding window method is also known as windowing.

Sliding Window Protocol

- The **sender has a buffer called the sending window** and the **receiver has buffer called the receiving window.**
- If the sequence number of the frames is an n-bit field, then the range of sequence numbers that can be assigned is 0 to 2^n–1.
- Consequently, the size of the sending window is 2^n–1.

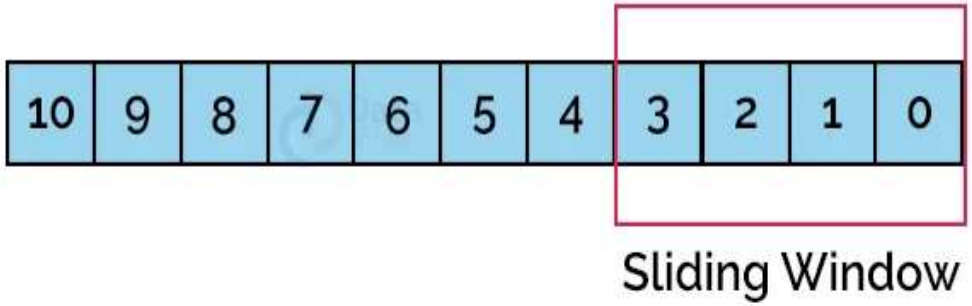
Sliding Window Protocol

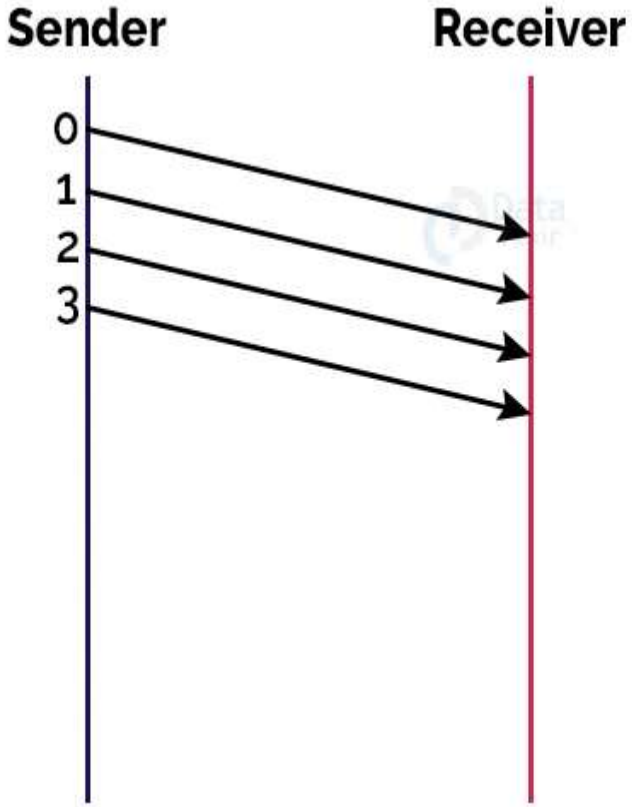
- if the sending window size is 4, then the sequence numbers will be 0, 1, 2, 3, 0, 1, 2, 3, 0, 1, and so on.
- The number of bits in the sequence number is 2 to generate the binary sequence 00, 01, 10, 11.
- The size of the receiving window is the maximum number of frames that the receiver can accept at a time.
- It determines the maximum number of frames that the sender can send before receiving acknowledgment.

- sender window = receiver window = 4.
- So the sequence numbering of both the windows will be 0,1,2,3,0,1,2

- The size of the sender window is at most 2k-1.
- For example; if 4 bits are allowed by the frame then the size of the window is 2 raised to the power 4 -1 16-1=15.
- The buffer is provided to the sender that has the size equal to the size of the window.

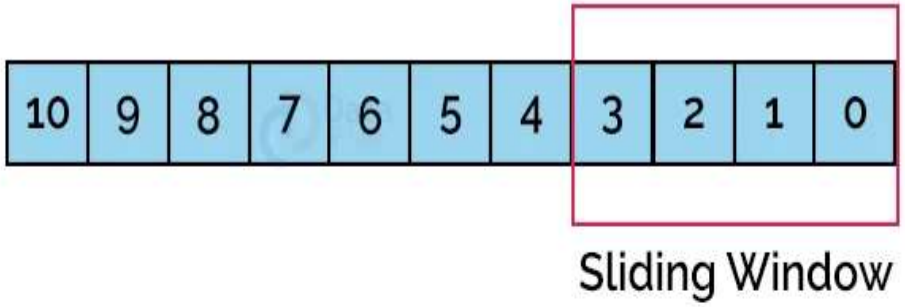
First, the sender sends the first four frames in the window (here the window size is 4).



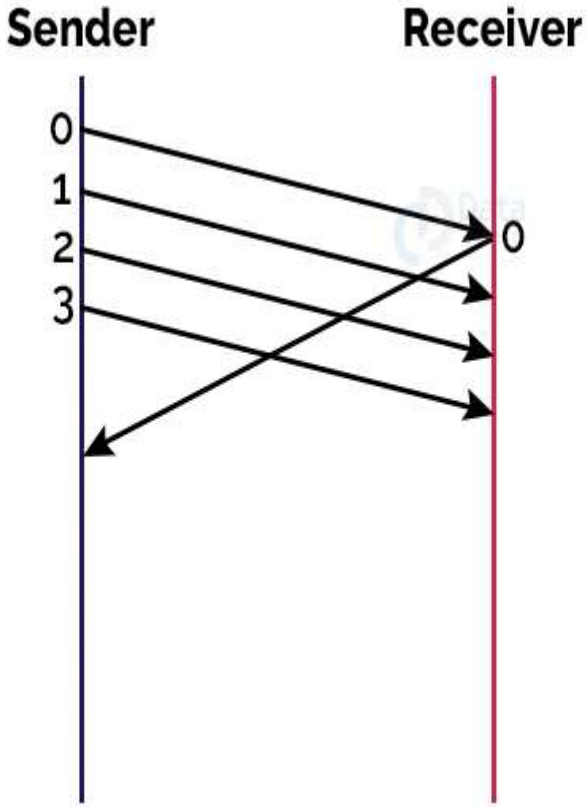


Window Size : 4

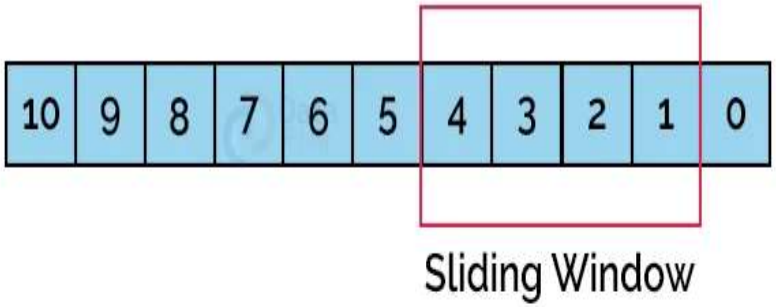
Then, the receiver sends the acknowledgment for the 0th frame.



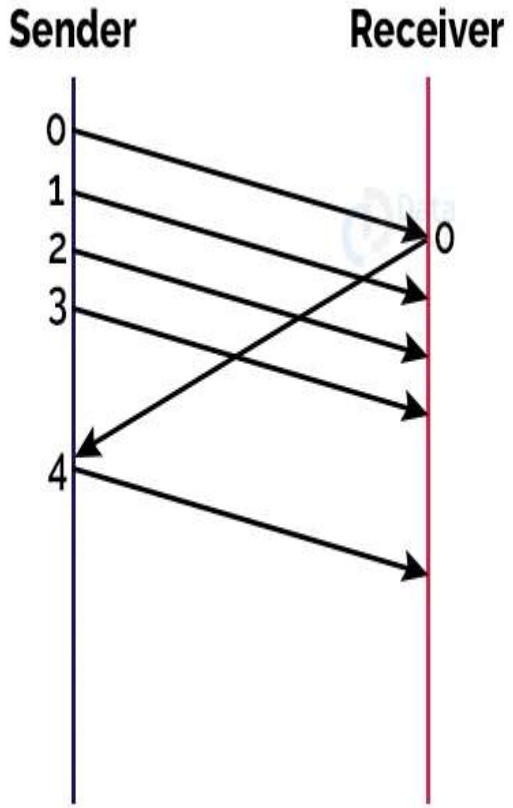
Window Size : 4

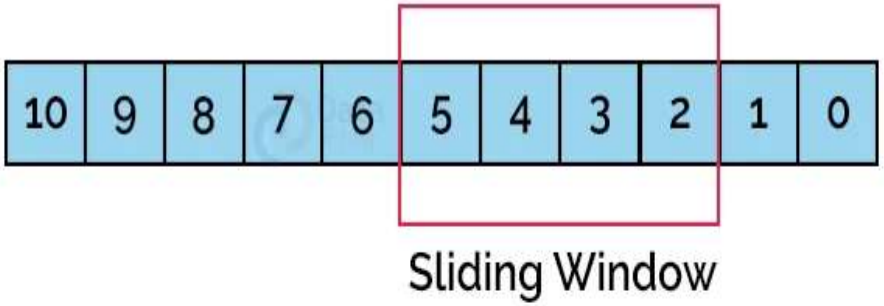


The receiver then slides the window over and sends the next frame in the queue.

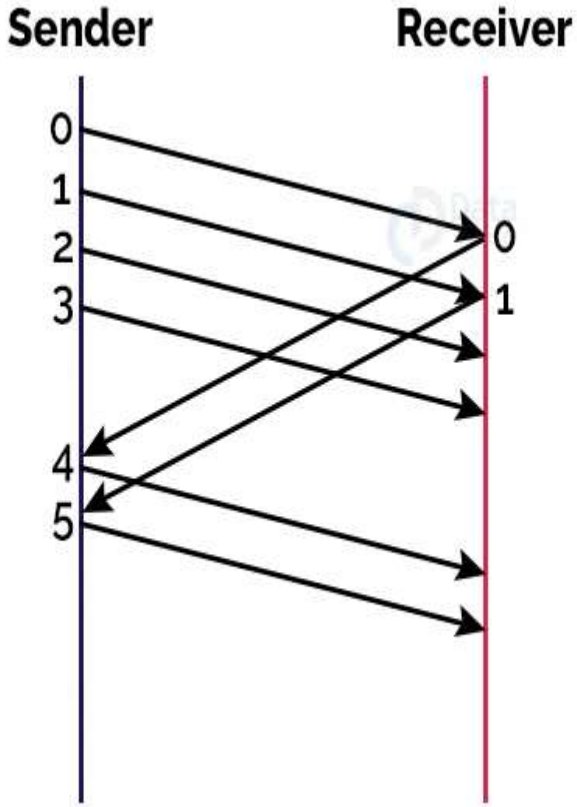


Window Size : 4





Window Size : 4



Accordingly, the receiver sends the acknowledgement for the 1st frame, and upon receiving that, the sender slides the window again and sends the next frame. This process keeps on happening until all the frames are sent successfully.

Types of Sliding Window Protocols:

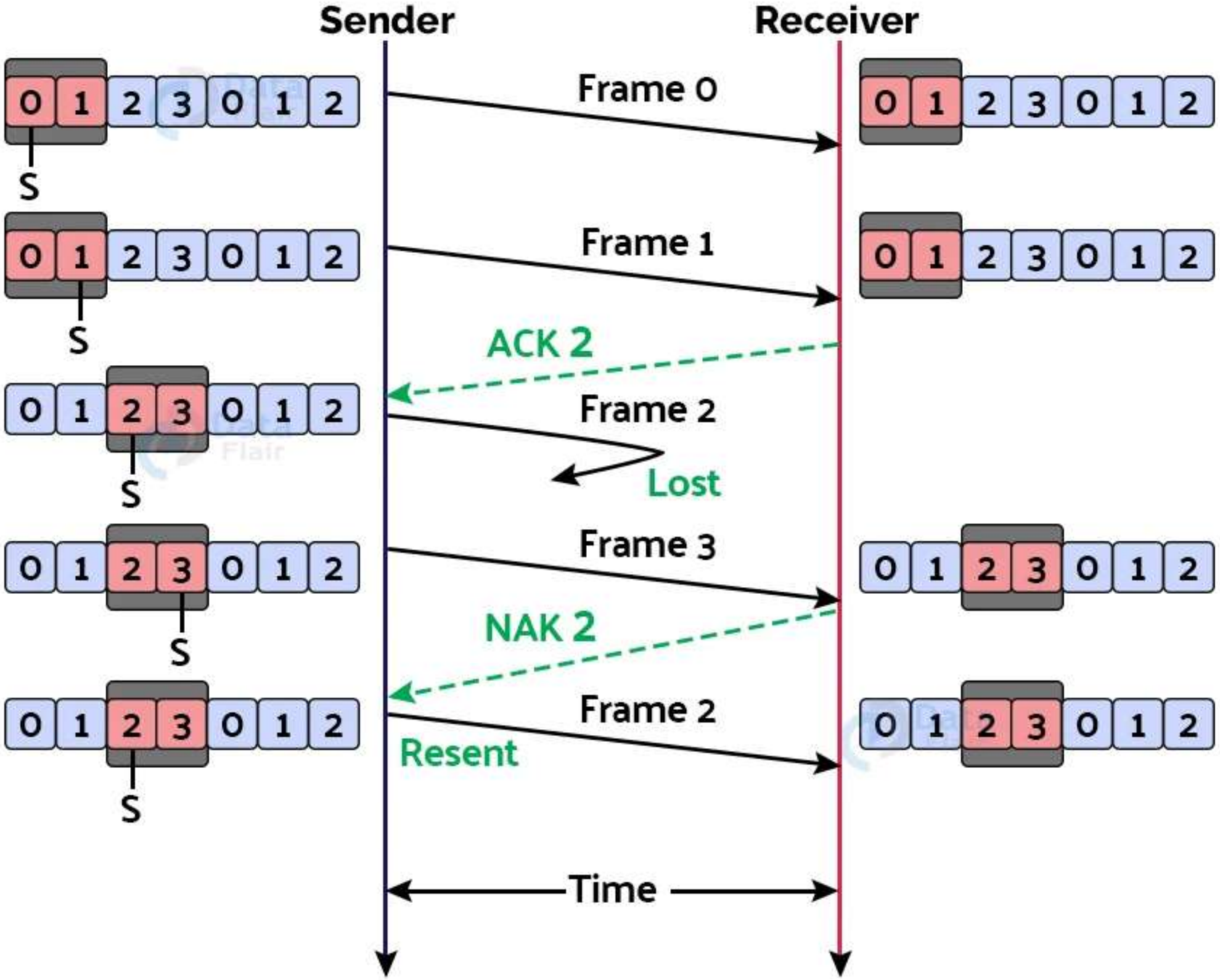
1. Go-Back-N ARQ:

The Go-Back-N Automatic Repeat Request protocol is also known as the Go-Back-N ARQ protocol.

In the event of corruption or loss of frames, all subsequent frames must be sent again.

In this protocol, the sender window size is N. The size of the receiver window is always one.

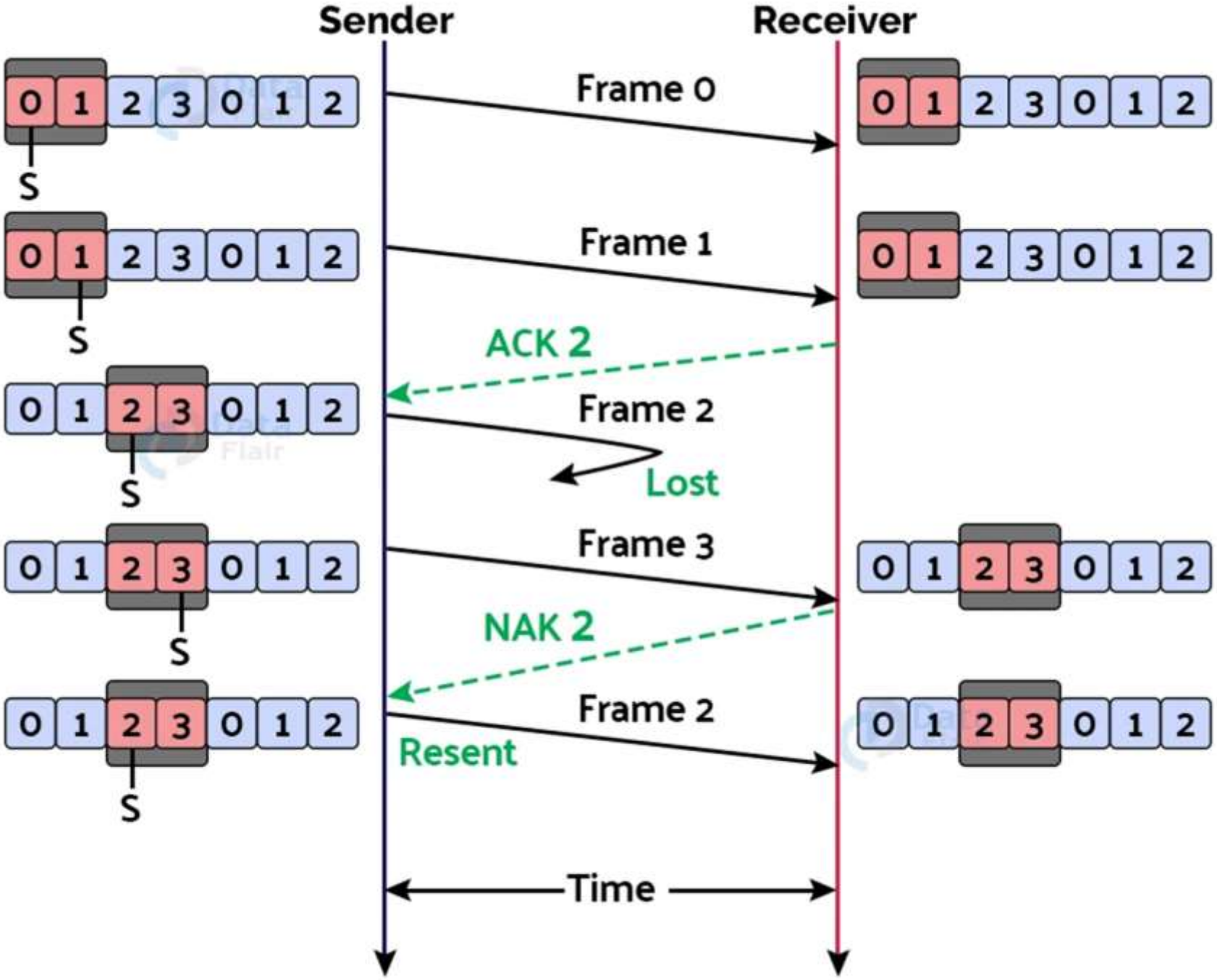
In the event of transmission of a corrupted frame, the receiver cancels it. The receiver does not accept a corrupted frame. The sender sends the correct frame again when the timer expires.


Sender
Receiver
0 1 2 3 0 1 2
S
Frame 0
Frame 1
ACK 2
Frame 2
Lost
Frame 3
NAK 2
Frame 2
Resent
Time

2. Selective Repeat ARQ:

Also known as Selective Repeat Automatic Repeat Request.

The size of the sender window is always equal to the size of the receiver window in this protocol. The sliding window's size is always greater than 1.


Sender
Receiver
0 1 2 3 0 1 2
S
Frame 0
Frame 1
ACK 2
Frame 2
Lost
Frame 3
NAK 2
Frame 2
Resent
Time

2. Selective Repeat ARQ:

- First, the sender sends the contents of the first window, which are frames 0 and 1 (because the window size is 2).

- b. When the receiver receives the frames sent above, it sends an acknowledgment for frame 2 (because frame 2 is the frame it expects to receive next).

- c. The sender then sends frames 2 and 3, however, frame 2 is lost on the way. The receiver thus sends back a “NAK” signal or a non-acknowledgment to let the sender know that frame 2 has been lost, and thus the sender retransmits frame 2.