MODULE 2

Data link layer - Data link layer design issues, Error detection and correction, Sliding window protocols, High-Level Data Link Control(HDLC)protocol. Medium Access Control (MAC) sublayer –Channel allocation problem, Multiple access protocols, Ethernet, Wireless LANs - 802.11, Bridges & switches - Bridges from 802.x to 802.y, Repeaters, Hubs, Bridges, Switches, Routers and Gateways.

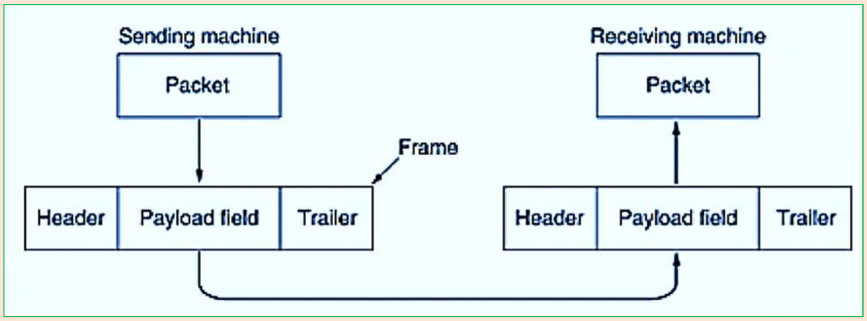
DLL DESIGN ISSUES

➢ Specific responsibilities of the data link layer include framing, addressing, flow control, error control, and media access control.

1. FRAMING

- The data link layer divides the stream of bits received from the network layer into manageable data units called frames.
- The data link layer adds a header to the frame to define the addresses of the sender and receiver of the frame.
- Each frame contains a frame header, a payload field for holding the packet, and a frame trailer

Relationship between packets and frames.



❖ **Fixed-Size Framing -** In fixed-size framing, there is no need for defining the boundaries of the frames; the size itself can be used as a delimiter.

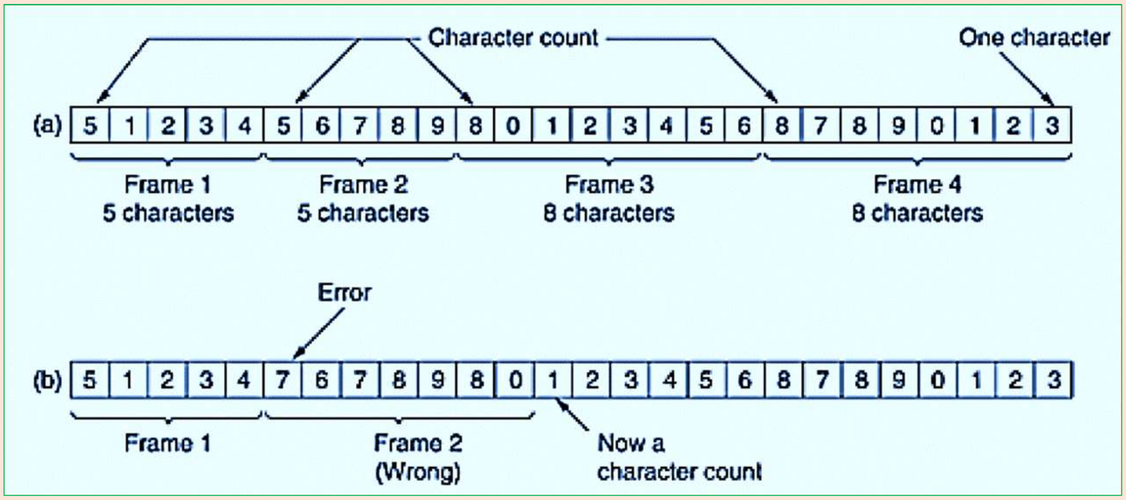
Eg: ATM wide-area network

❖ Variable-Size Framing

- In variable-size framing, we need a way to define the end of the frame and the beginning of the next.
- The approaches were used for this purpose are:

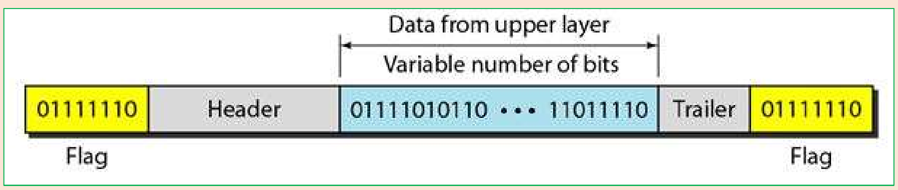
A. Character Count

- It uses a field in the header to specify the number of characters in the frame.
- When the data link layer at the destination sees the character count, it knows how many characters follow and hence where the end of the frame is.



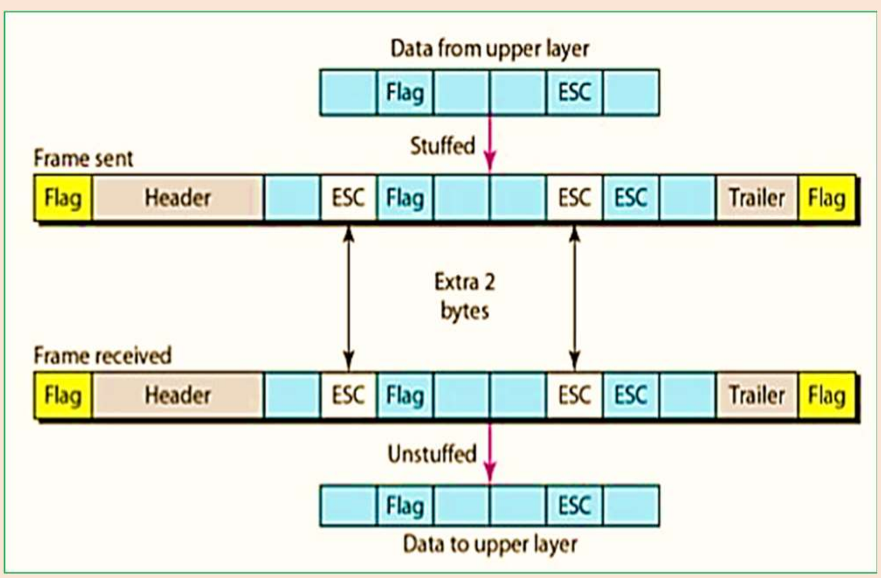
B. Flag bytes with byte stuffing

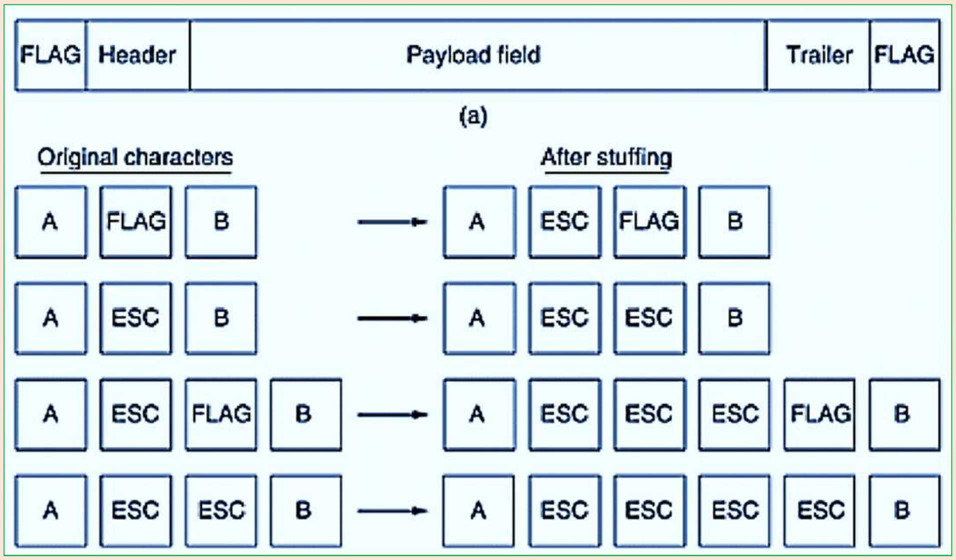
- To separate one frame from the next, an 8-bit (1-byte) flag is added at the beginning and the end of a frame.



- Any pattern used for the flag could also be part of the information. If this happens, the receiver, when it encounters this pattern in the middle of the data, thinks it has reached the end of the frame.
- To fix this problem, In byte stuffing (or character stuffing), a special byte is added to the data section of the frame when there is a character with the same pattern as the flag.

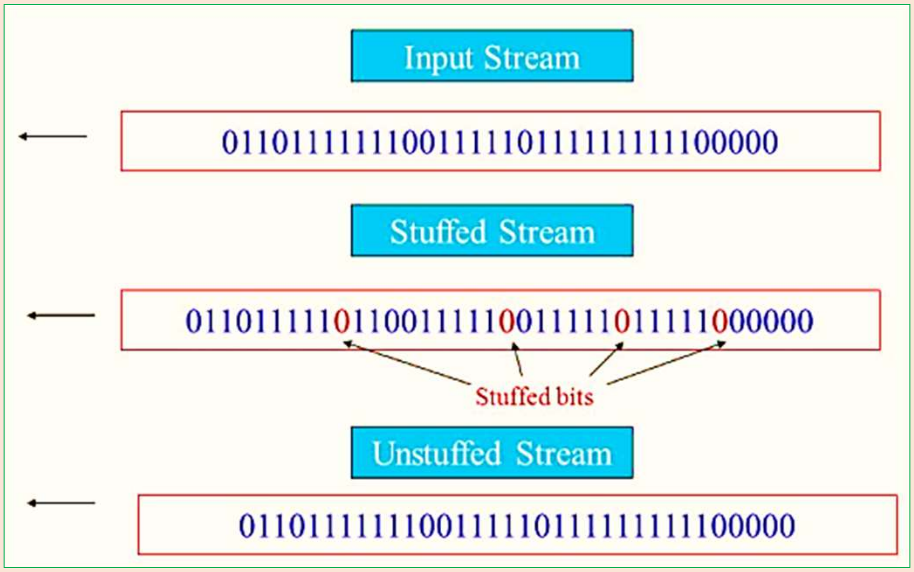
- This byte is usually called the escape character (ESC) which has a predefined bit pattern.
- Whenever the receiver encounters the ESC character, it removes it from the data section and treats the next character as data, not a delimiting flag.
- If the text contains one or more escape characters followed by a flag, the receiver removes the escape character, but keeps the flag, which is incorrectly interpreted as the end of the frame.
- To solve this problem, the escape characters that are part of the text must also be marked by another escape character.
- In other words, if the escape character is part of the text, an extra one is added to show that the second one is part of the text.





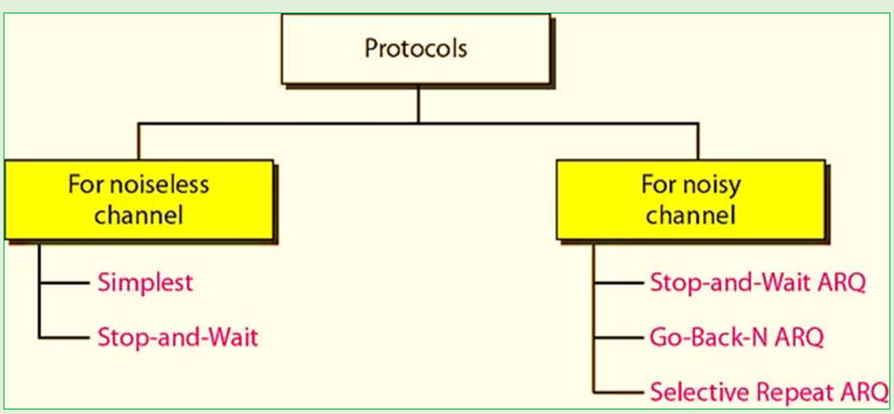
C. Bit Stuffing

- Most protocols use a special 8-bit pattern flag 01111110 as the delimiter to define the beginning and the end of the frame
- That is, if the flag pattern appears in the data, we need to somehow inform the receiver that this is not the end of the frame.
- We do this by stuffing 1 single bit (instead of 1 byte) to prevent the pattern from looking like a flag. The strategy is called bit stuffing.
- In bit stuffing, if a 0 and five consecutive 1 bits are encountered, an extra 0 is added.
- This extra stuffed bit is eventually removed from the data by the receiver.



2. ERROR CONTROL

- Error control is both error detection and error correction.
- Any time an error is detected in an exchange, specified frames are retransmitted. This process is called automatic repeat request (ARQ).

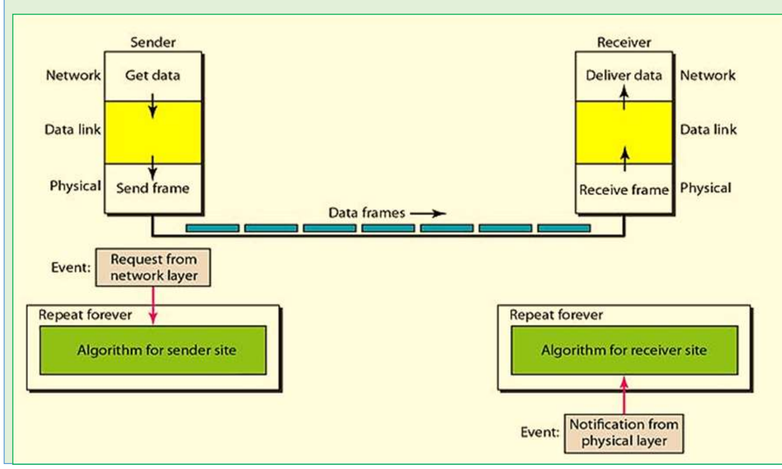


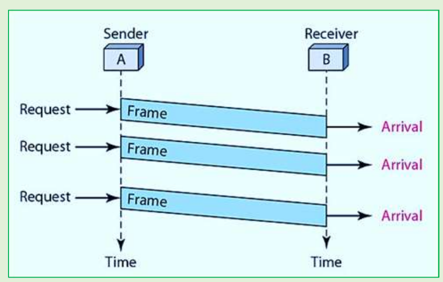
❖NOISELESS CHANNELS

1. Simplest Protocol

- No flow control and error control
- It is a unidirectional protocol in which data frames are traveling in only one direction-from the sender to receiver.
- There is no need for flow control in this scheme. The data link layer at the sender site gets data from its network layer, makes a frame out of the data, and sends it.
- The data link layer at the receiver site receives a frame from its physical layer, extracts data from the frame, and delivers the data to its network layer.

- The data link layers of the sender and receiver provide transmission services for their network layers. The data link layers use the services provided by their physical layers (such as signaling, multiplexing, and so on) for the physical transmission of bits.



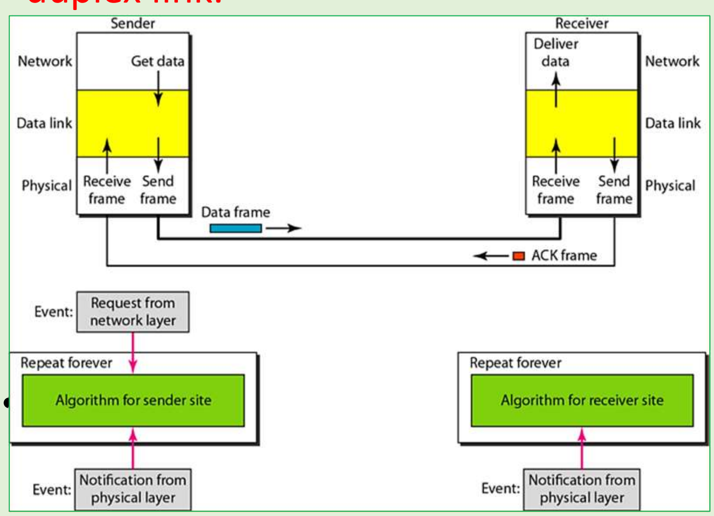


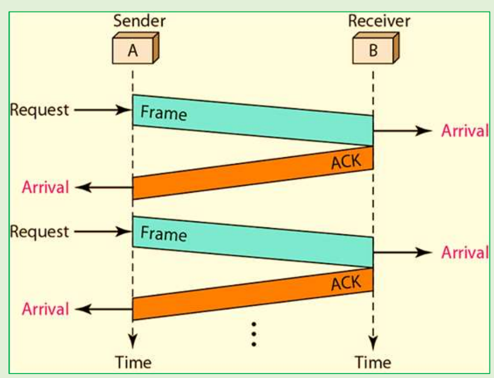
Flow Diagram

2. Stop-and-Wait Protocol

- To prevent the receiver from becoming overwhelmed with frames, we somehow need to tell the sender to slow down. There must be feedback from the receiver to the sender.
- In Stop-and-Wait Protocol , the sender sends one frame, stops until it receives confirmation from the receiver (okay to go ahead), and then sends the next frame.
- We still have unidirectional communication for data frames, but auxiliary ACK frames (simple tokens of acknowledgment) travel from the other direction.
- We add flow control to our previous protocol. After a frame is sent, the algorithm must ignore another network layer request until that frame is acknowledged.

- At any time, there is either one data frame on the forward channel or one ACK frame on the reverse channel. We therefore need a half-duplex link.





Flow Diagram

❖NOISY CHANNELS

1. Stop-and-Wait Automatic Repeat Request

- It adds a simple error control mechanism to the Stop-and-Wait Protocol
- To detect and correct corrupted frames, we need to add redundancy bits to our data frame .When the frame arrives at the receiver site, it is checked and if it is corrupted, it is silently discarded. The detection of errors in this protocol is manifested by the silence of the receiver.
- Lost frames are more difficult to handle than corrupted ones. The received frame could be the correct one, or a duplicate, or a frame out of order. The solution is to number the frames. When the receiver receives a data frame that is out of order, this means that frames were either lost or duplicated.

- The lost frames need to be resent in this protocol. the sender keeps a copy of the sent frame. At the same time, it starts a timer.
- If the timer expires and there is no ACK for the sent frame, the frame is resent, the copy is held, and the timer is restarted.
- Since the protocol uses the stop-and-wait mechanism, there is only one specific frame that needs an ACK even though several copies of the same frame can be in the network.
- Since an ACK frame can also be corrupted and lost, it too needs redundancy bits and a sequence number. The ACK frame for this protocol has a sequence number field. In this protocol, the sender simply discards a corrupted ACK frame or ignores an out-of-order one

➢Sequence Numbers

- The frames need to be numbered. This is done by using sequence numbers. A field is added to the data frame to hold the sequence number of that frame.
- One important consideration is the range of the sequence numbers. Since we want to minimize the frame size, we look for the smallest range.
- Assume we have used **x** as a sequence number; we only need to use **x + 1** after that. There is no need for **x + 2**.
- To show this, assume that the sender has sent the frame numbered x. Three things can happen.

1. The frame arrives safe and sound at the receiver site; the receiver sends an acknowledgment. The acknowledgment arrives at the sender site, causing the sender to send the next frame numbered x + 1.

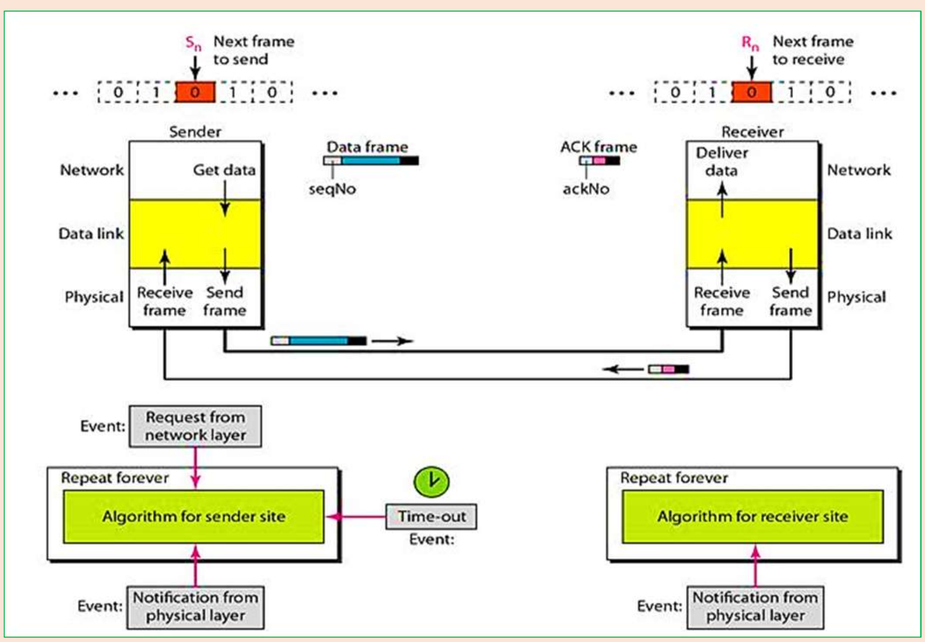
2. The frame arrives safe and sound at the receiver site; the receiver sends an acknowledgment, but the acknowledgment is corrupted or lost. The sender resends the frame (numbered x) after the time-out. Note that the frame here is a duplicate. The receiver can recognize this fact because it expects frame x + I but frame x was received.

3. The frame is corrupted or never arrives at the receiver site; the sender resends the frame (numbered x) after the time-out.

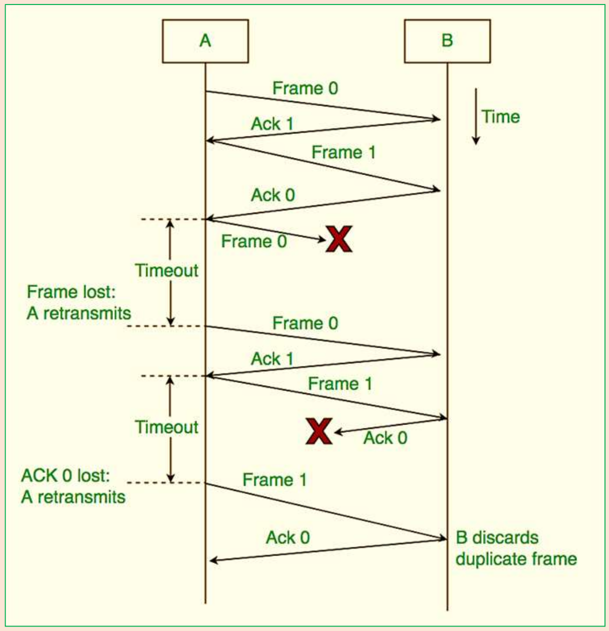
➢We can see that there is a need for sequence numbers x and x + I.

➢Acknowledgment Numbers

- The acknowledgment numbers always announce the sequence number of the next frame expected by the receiver.
- For example, if frame 0 has arrived safe and sound, the receiver sends an ACK frame with acknowledgment 1 (meaning frame 1 is expected next).



Flow diagram



2. Go-Back-N Automatic Repeat Request

- In this protocol we can send several frames before receiving acknowledgments.
- We keep a copy of these frames until the acknowledgments arrive.

➢ **Sequence Numbers**

- Frames from a sending station are numbered sequentially. Because we need to include the sequence number of each frame in the header, we need to set a limit.
- If the header of the frame allows m bits for the sequence number, the sequence numbers range from **0 to 2^m - 1**. For example, if m is 4, the only sequence numbers are 0 – 15. However, we can repeat the sequence. So the sequence numbers are

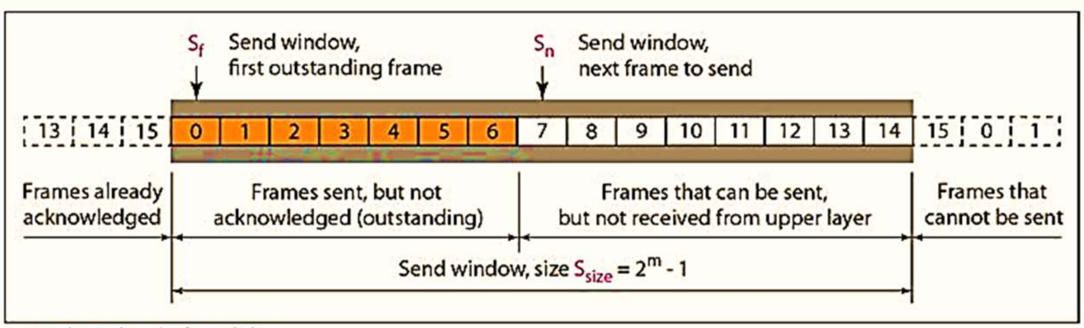
0, 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,0,1,2,3,4,5,6,7,8,9,10, 11, ...

- In the Go-Back-N Protocol, the sequence numbers are modulo2^m , where m is the size of the sequence number field in bits.

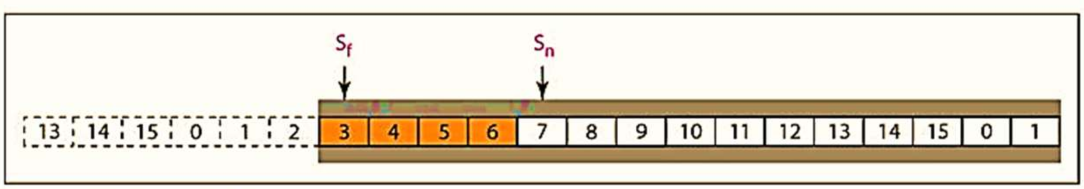
➢Sliding Window

- The sliding window is an abstract concept that defines the range of sequence numbers .
- The sender and receiver need to deal with only part of the possible sequence numbers.
- The window at any time divides the possible sequence numbers into four regions

Send Window



a. Send window before sliding

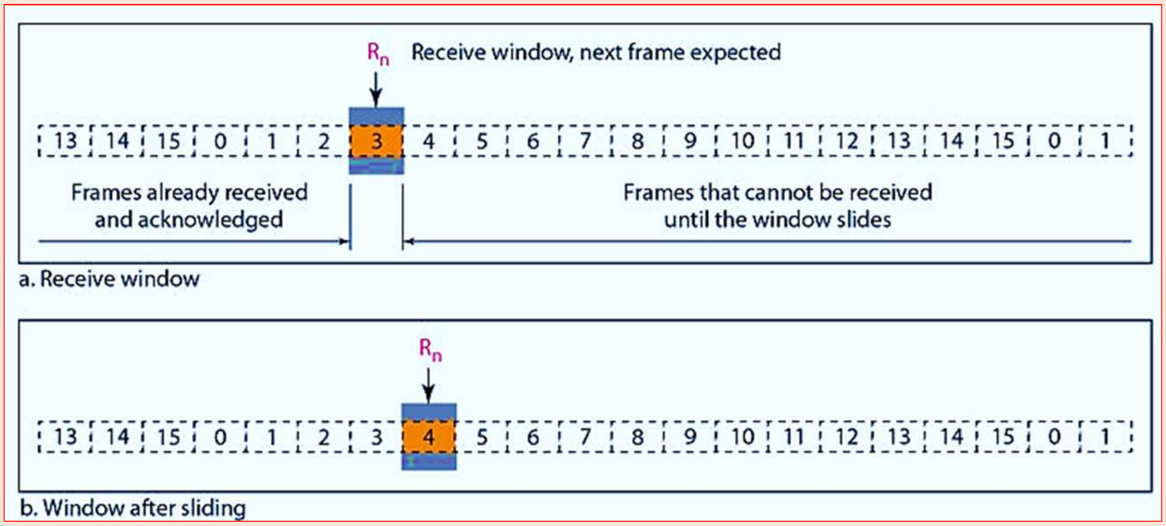


b. Send window after sliding

- The window itself is an abstraction; three variables define its size and location at any time.
- **Sf**(send window, the first outstanding frame)
- **Sn** (send window, the next frame to be sent)
- **Ssize** (send window, size).
- The variable Sf defines the sequence number of the first (oldest) outstanding frame.
- The variable Sn holds the sequence number that will be assigned to the next frame to be sent.
- Finally, the variable Ssize defines the size of the window, which is fixed in our protocol.

- The acknowledgments in this protocol are cumulative, meaning that more than one frame can be acknowledged by an ACK frame.
- In Figure b, frames 0, I, and 2 are acknowledged, so the window has slide to the right three slots.
- The receive window makes sure that the correct data frames are received and that the correct acknowledgments are sent.
- The size of the receive window is always 1.
- The receiver is always looking for the arrival of a specific frame.
- Any frame arriving out of order is discarded and needs to be resent.

Receive Window



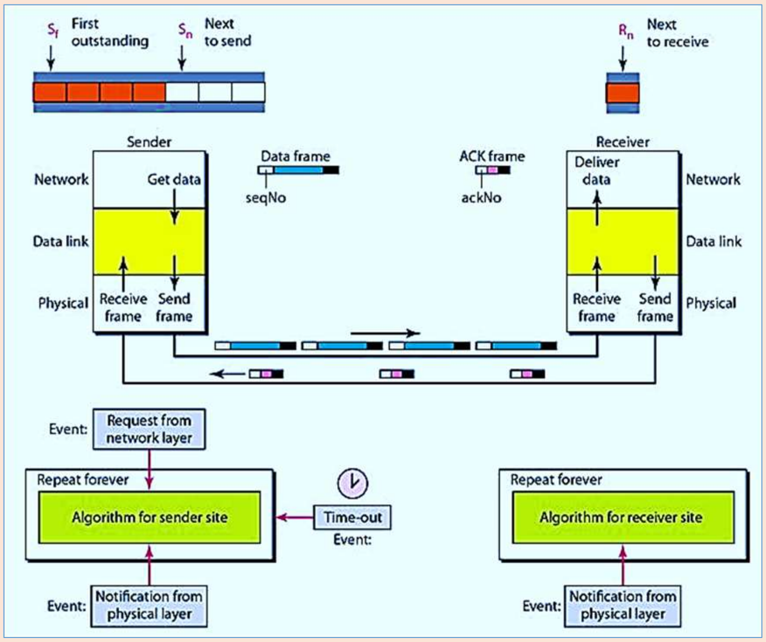
- we need only one variable Rn (receive window, next frame expected) to define this abstraction.
- The sequence numbers to the left of the window belong to the frames already received and acknowledged
- The sequence numbers to the right of this window define the frames that cannot be received. Any received frame with a sequence number in these two regions is discarded.
- Only a frame with a sequence number matching the value of Rn is accepted and acknowledged.
- The receive window also slides, but only one slot at a time. When a correct frame is received (and a frame is received only one at a time), the window slides.

➢Acknowledgment

- The receiver sends a positive acknowledgment if a frame has arrived safe and sound and in order.
- If a frame is damaged or is received out of order, the receiver is silent and will discard all subsequent frames until it receives the one it is expecting.
- The silence of the receiver causes the timer of the unacknowledged frame at the sender site to expire. This, in turn, causes the sender to go back and resend all frames, beginning with the one with the expired timer.
- The receiver does not have to acknowledge each frame received. It can send one cumulative acknowledgment for several frames

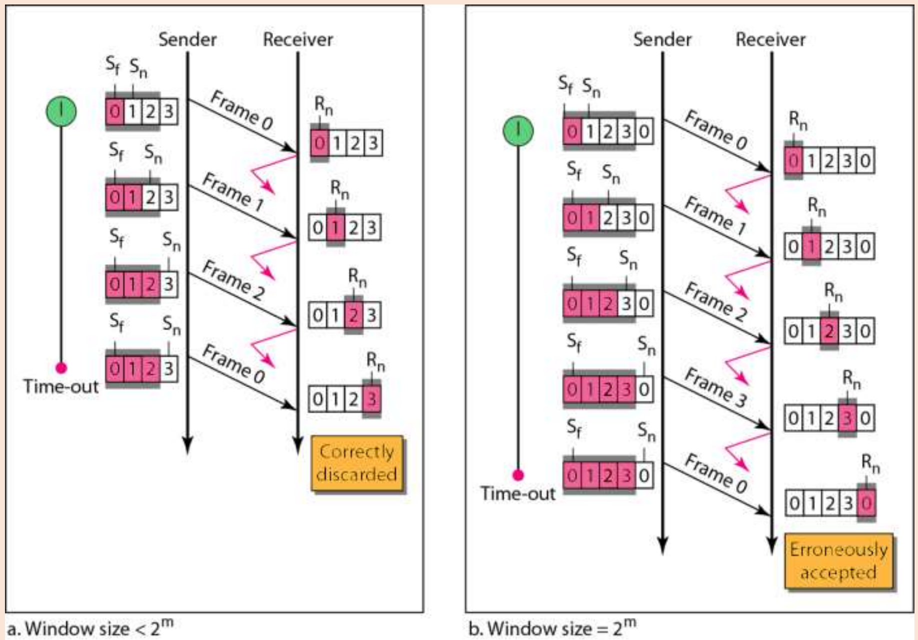
➢Resending a Frame

- When the timer expires, the sender resends all outstanding frames.
- For example, suppose the sender has already sent frame 6, but the timer for frame 3 expires. This means that frame 3 has not been acknowledged; the sender goes back and sends frames 3, 4,5, and 6 again.
- That is why the protocol is called Go-Back-N ARQ.
- In Go-Back-N ARQ, the size of the send window must be **less than 2^m** and the size of the receiver window is always **1**



➢Send Window Size

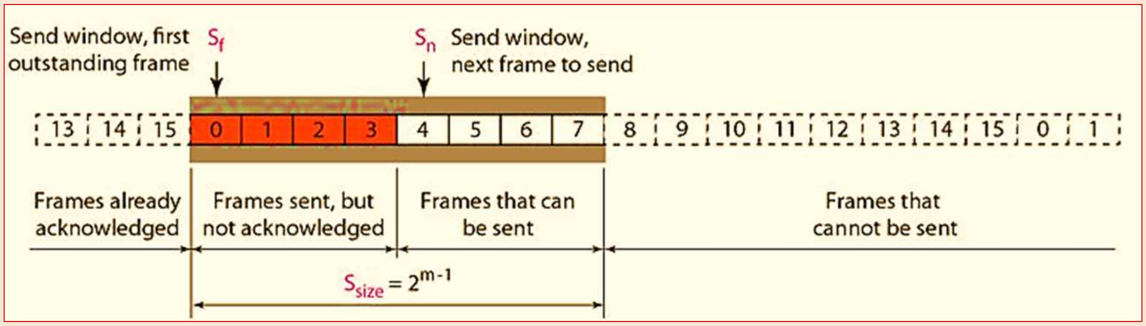
- Suppose m =2, which means the size of the window can be 2^m - 1, or 3.
- If the size of the window is 3 (less than 2^2) and all three acknowledgments are lost, the frame timer expires and all three frames are resent.
- The receiver is now expecting frame 3, not frame 0, so the duplicate frame is correctly discarded.
- On the other hand, if the size of the window is 4 (equal to 2^2) and all acknowledgments are lost, the sender will send a duplicate of frame 0.
- However, this time the window of the receiver expects to receive frame 0, so it accepts frame 0, not as a duplicate, but as the first frame in the next cycle. This is an error.



3. Selective Repeat Automatic Repeat Request

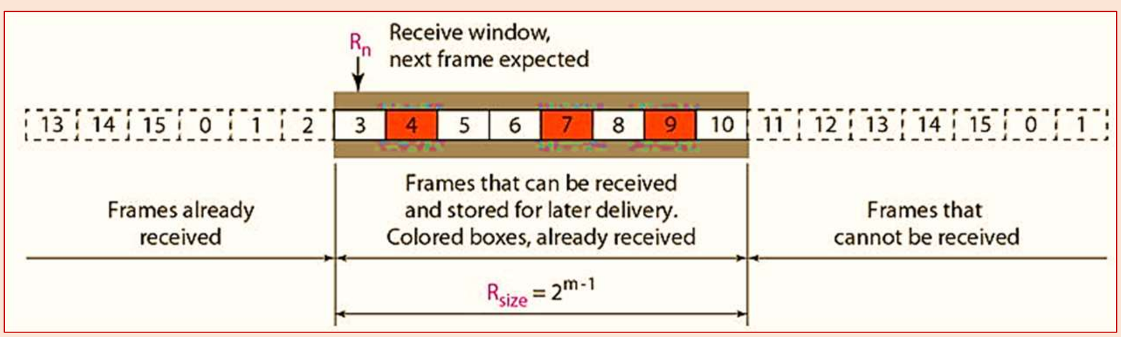
- Go-Back-N ARQ resending of multiple frames.
- For noisy links, there is another mechanism that does not resend N frames when just one frame is damaged; only the damaged frame is resent.
- This mechanism is called Selective Repeat ARQ. It is more efficient for noisy links.
- The Selective Repeat Protocol also uses two windows: a send window and a receive window
- The size of the send window is much smaller; it is 2^{m-1}
- The receive window is the same size as the send window

- If m = 4, the sequence numbers go from 0 to 15, but the size of the window is just 8 (it is 15 in the Go-Back-N Protocol).

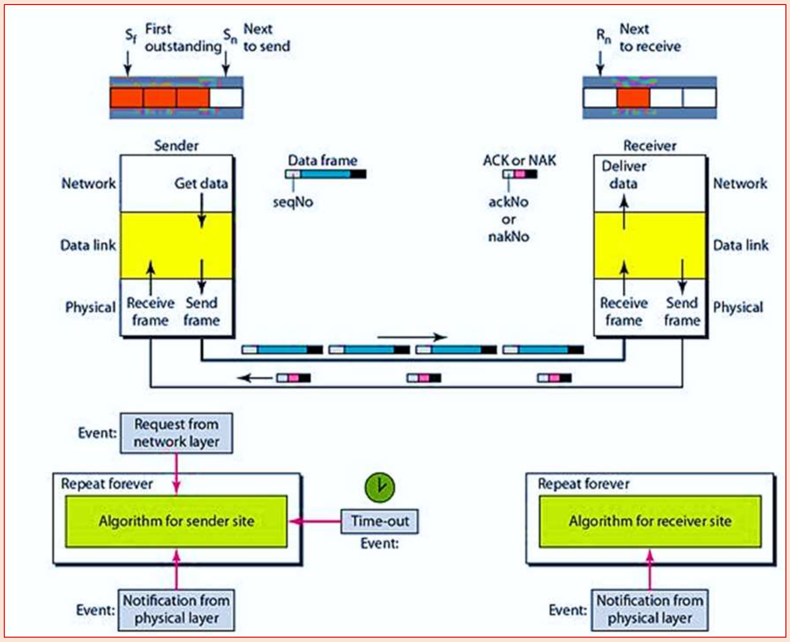


- The Selective Repeat Protocol allows as many frames as the size of the receive window to arrive out of order and be kept until there is a set of in-order frames to be delivered to the network layer.

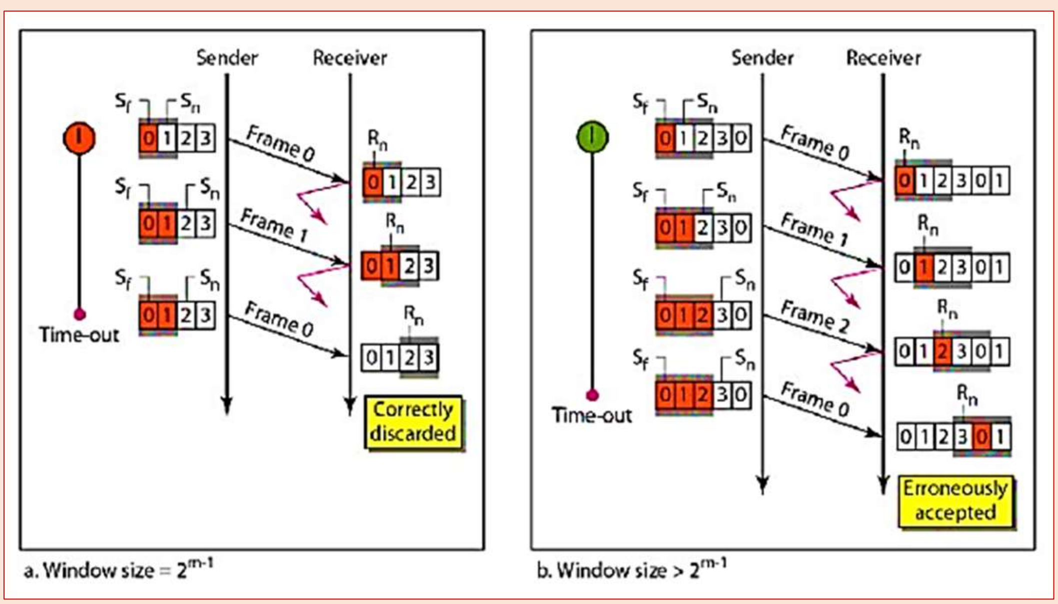
- Because the sizes of the send window and receive window are the same, all the frames in the send frame can arrive out of order and be stored until they can be delivered



- Those slots inside the window that are colored define frames that have arrived out of order and are waiting for their neighbors to arrive before delivery to the network layer.



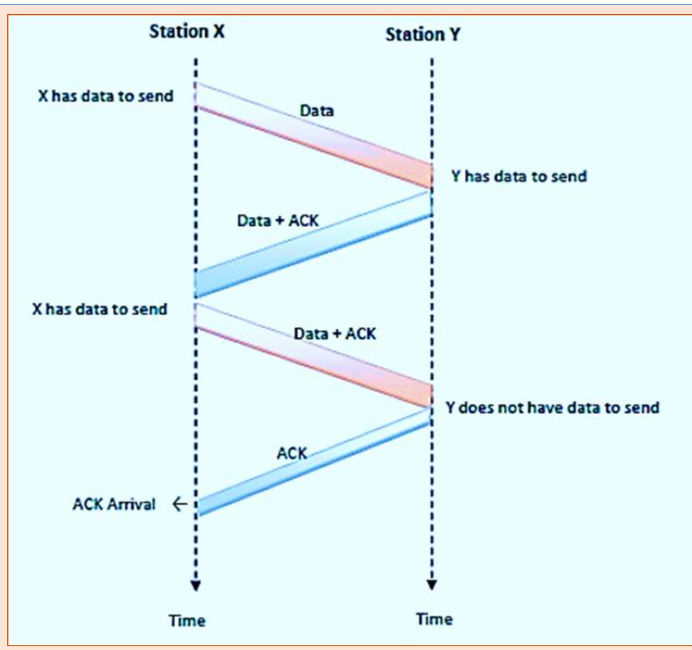
Sf First outstanding
Sn Next to send
Rn Next to receive
Sender
Receiver
Data frame
seqNo
ACK or NAK
ackNo
or
nakNo
Network
Data link
Physical
Get data
Receive frame
Send frame
Deliver data
Event:
Request from network layer
Repeat forever
Algorithm for sender site
Time-out
Event:
Notification from physical layer
Algorithm for receiver site



Sender
Receiver
Sf
Sn
Rn
Frame 0
Frame 1
Frame 2
Time-out
Correctly discarded
Erroneously accepted
a. Window size = 2m-1
b. Window size > 2m-1

❖Piggybacking

- Communications are mostly full – duplex in nature, i.e. data transmission occurs in both directions
- In reliable full - duplex data transmission, the technique of hooking up acknowledgments onto outgoing data frames is called piggybacking.
- Piggybacking is used to improve the efficiency of the bidirectional protocols
- When a frame is carrying data from A to B, it can also carry control information about arrived (or lost) frames from B; when a frame is carrying data from B to A, it can also carry control information about the arrived (or lost) frames from A.



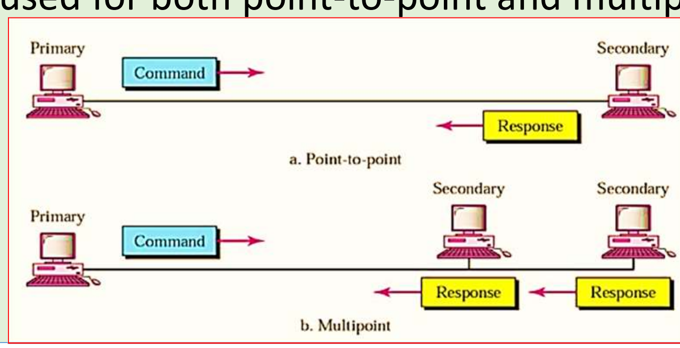
HDLC

- High-level Data Link Control (HDLC) is a bit-oriented protocol for communication over point-to-point and multipoint links.
- It implements the ARQ mechanisms
- HDLC provides two common transfer modes that can be used in different configurations:

1. Normal Response Mode (NRM)
2. Asynchronous Balanced Mode (ABM).

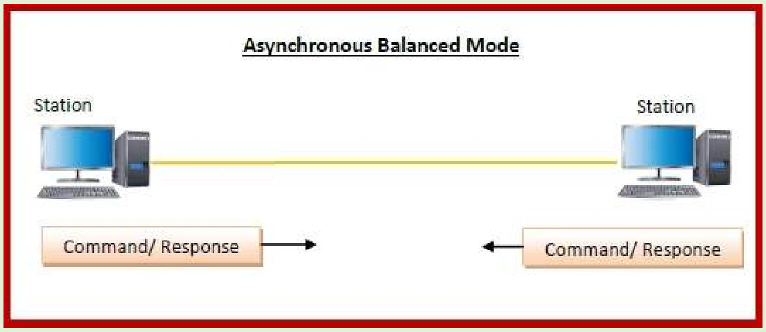
1. Normal Response Mode

- The station configuration is unbalanced.
- We have one primary station and multiple secondary stations. A primary station can send commands; a secondary station can only respond.
- The NRM is used for both point-to-point and multiple-point links



2. Asynchronous Balanced Mode

- The configuration is balanced.
- The link is point-to-point, and each station can function as a primary and a secondary (acting as peers)
- This is the common mode today.



❖Frames

- HDLC defines three types of frames:

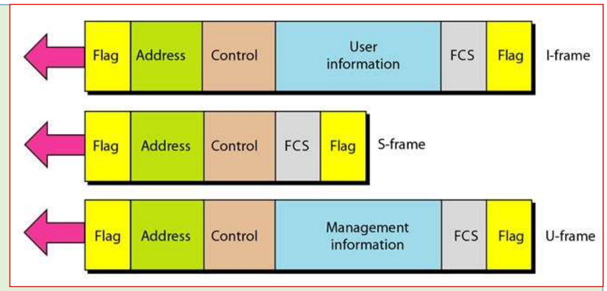
➢ **Information frames (I-frames)**

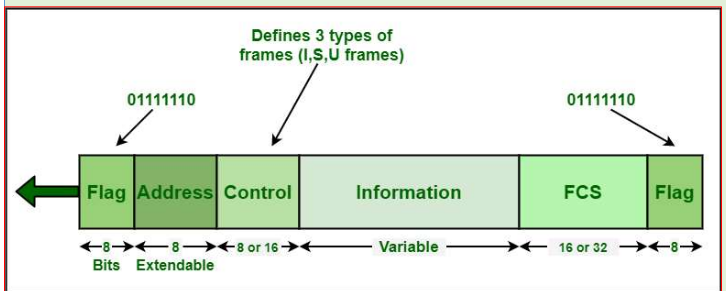
➢ **Supervisory frames (S-frames)**

➢ **Unnumbered frames (V-frames)**

- I-frames are used to transport user data and control information relating to user data (piggybacking).
- S-frames are used only to transport control information.
- V-frames are reserved for system management. Information carried by V-frames is intended for managing the link itself.

❖Frame Format





Basic Frame Structure

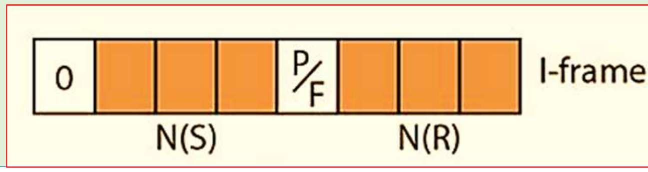
- Flag field - The bit pattern 01111110, that identifies both the beginning and the end of a frame
- Address field - Contains the address of the secondary station. If a primary station created the frame, it contains a to address. If a secondary creates the frame, it contains a from address.
- Control field - The control field is a 1 or 2 byte segment of the frame used for flow and error control.
- Information field - The information field contains the user's data from the network layer or management information
- FCS field - The frame check sequence (FCS) is the HDLC error detection field. It can contain either a 2- or 4-byte ITU-T CRC

❖Control Field

- The control field determines the type of frame

➢Control Field for I-Frames

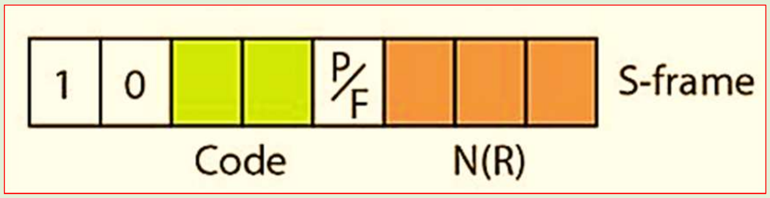
- I-frames are designed to carry user data from the network layer.
- They can include flow and error control information (piggybacking).
- The first bit defines the type. If the first bit of the control field is 0, this means the frame is an I-frame.
- The next 3 bits, called N(S),define the sequence number of the frame



- The last 3 bits, called N(R), correspond to the acknowledgment number when piggybacking is used
- The single bit between N(S) and N(R) is called the P/F bit.
- The P/F field is a single bit with a dual purpose. It has meaning only when it is set (bit = 1) and can mean Poll or Final. It means poll when the frame is sent by a primary station to a secondary (when the address field contains the address of the receiver). It means final when the frame is sent by a secondary to a primary (when the address field contains the address of the sender).

➢Control Field for S-Frames

- S-frames do not have information fields.
- If the first 2 bits of the control field is 10, this means the frame is an S-frame.
- The last 3 bits, called N(R), corresponds to the acknowledgment number (ACK) or negative acknowledgment number (NAK) depending on the type of S-frame



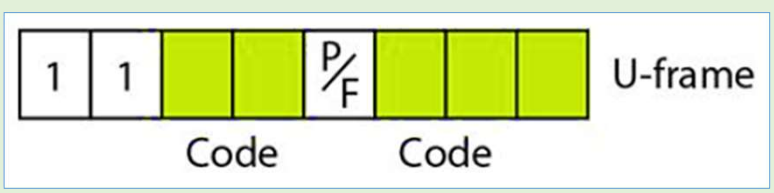
- The 2 bits called code is used to define the type of S-frame itself. With 2 bits, we can have four types of S-frames.

1. **Receive ready (RR)** - If the value of the code subfield is 00, it is an RR S-frame. This kind of frame acknowledges the receipt of a safe and sound frame or group of frames. In this case, the value N(R) field defines the acknowledgment number.
2. **Receive not ready (RNR)** - If the value of the code subfield is 10, it is an RNR S-frame. It announces that the receiver is busy and cannot receive more frames. It acts as a kind of congestion control mechanism by asking the sender to slow down. The value of N(R) is the acknowledgment number.

3. **Reject (REJ)** - If the value of the code subfield is 01, it is a REJ S-frame. This is a NAK frame. It is a NAK that can be used in Go-Back-N ARQ to improve the efficiency of the process by informing the sender, before the sender time expires, that the last frame is lost or damaged. The value of N(R) is the negative acknowledgment number.
4. **Selective reject (SREJ)** - If the value of the code subfield is 11, it is an SREJ S-frame. This is a NAK frame used in Selective Repeat ARQ. Note that the HDLC Protocol uses the term selective reject instead of selective repeat. The value of N(R) is the negative acknowledgment number

➢ Control Field for U-Frames

- Unnumbered frames are used to exchange session management and control information between connected devices.
- U-frame codes are divided into two sections: a 2-bit prefix before the P/F bit and a 3-bit suffix after the P/F bit.
- Together, these two segments (5 bits) can be used to create up to 32 different types of U-frames.



- It reduces the chance of collision and improves efficiency. In this method, after the station finds the line idle it

follows these steps:

1. With probability p, the station sends its frame.

2. With probability q = 1 − p, the station waits for the beginning of the next time slot and checks the line again.

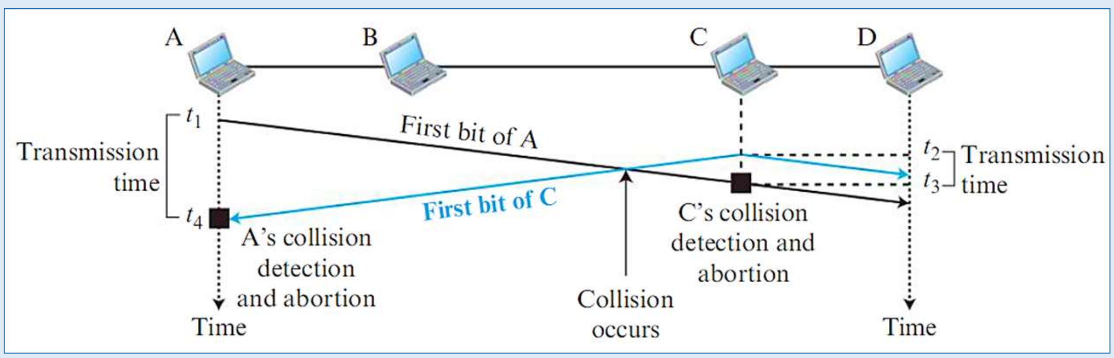
a. If the line is idle, it goes to step 1.

b. If the line is busy, it acts as though a collision has occurred and uses the backoff procedure.

❖CSMA/CD

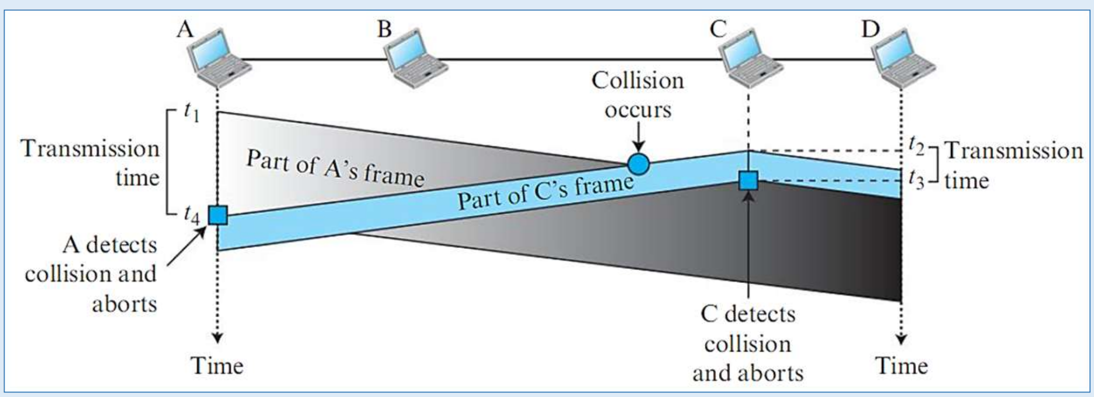
- In this method, a station monitors the medium after it sends a frame to see if the transmission was successful. If so, the station is finished. If, however, there is a collision, the frame is sent again.

Collision of the first bits in CSMA/CD



- At time t1, station A has executed its persistence procedure and starts sending the bits of its frame.
- At time t2, station C has not yet sensed the first bit sent by A.
- Station C executes its persistence procedure and starts sending the bits in its frame, which propagate both to the left and to the right.
- The collision occurs sometime after time t2. Station C detects a collision at time t3 when it receives the first bit of A's frame.
- Station C immediately (or after a short time, but we assume immediately) aborts transmission. Station A detects collision at time t4 when it receives the first bit of C's frame; it also immediately aborts transmission.

Collision and abortion in CSMA/CD



- A transmits for the duration t4 − t1.
- C transmits for the duration t3 − t2.

➢Energy Level

- The level of energy in a channel can have three values: zero, normal, and abnormal.
- At the zero level, the channel is idle.
- At the normal level, a station has successfully captured the channel and is sending its frame.
- At the abnormal level, there is a collision and the level of the energy is twice the normal level.
- A station that has a frame to send or is sending a frame needs to monitor the energy level to determine if the channel is idle, busy, or in collision mode.

➢Throughput

- The throughput of CSMA/CD is greater than that of pure or slotted ALOHA.
- For the 1-persistent method, the maximum throughput is around 50 percent when G = 1.
- For the nonpersistent method, the maximum throughput can go up to 90 percent when G is between 3 and 8.
- Traditional Ethernet was a broadcast LAN that used 1-persistence method to control access to the common media.

❖Distributed Coordination Function

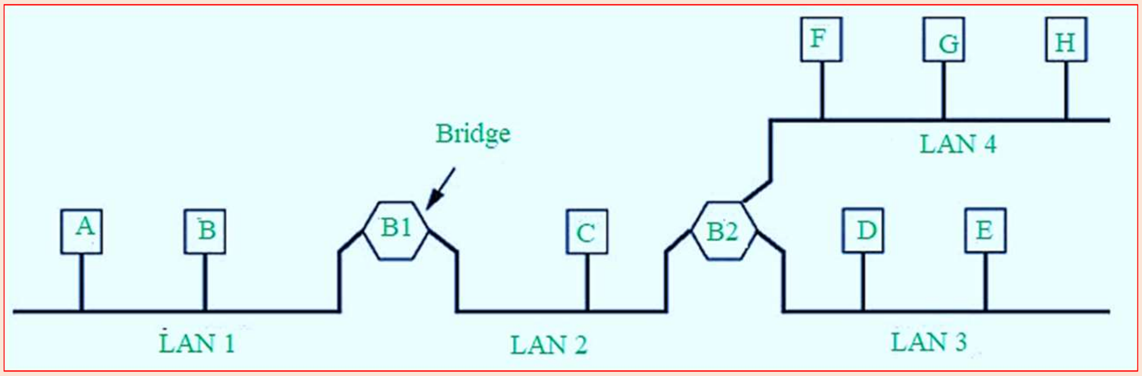
- It is a protocol defined by IEEE at the MAC sublayer
- DCF uses CSMA/CA as the access method

❖Point Coordination Function (PCF)

- The point coordination function (PCF) is an optional access method that can be implemented in an infrastructure network (not in an ad hoc network).
- It is used mostly for time-sensitive transmission.
- PCF has a centralized, contention-free polling access method.
- The AP performs polling for stations that are capable of being polled. The stations are polled one after another, sending any data they have to the AP.

❖Bridges

- Bridges are a data link layer device and can connect to different networks as well as connect different networks of different types.
- Bridges from 802.x to 802.y where x & y may both be Ethernet or one can be Ethernet and other may be a token ring, etc.
- It locally connects small LANs, whereas if LANs are big then bridges can no longer handle them.
- Bridge follows a protocol in IEEE format execute 802.1 which is a spanning tree of bridges.
- It stores and forwards Ethernet frames, i.e., it has to do with the MAC address, they handle the hardware addresses.



- It also examine the frame header and selectively forward frames based on MAC destination address, such as in the given figure if Bridge 2 receives a packet then it will selectively decide whether to send it to LAN 3 or LAN 4.

- When a frame is to be forwarded in a segment it uses CSMA/CD to access the segment.
- The hosts are unaware of the presence of bridges, it appears to them as a single whole network.
- Bridges need not be configured they are plug-and-play and self-learning devices, i.e. a bridge has a learning table, they learn which hosts can be reached through which interfaces.
- At the physical level, the bridge boosts the signal strength like a repeater or completely regenerates the signal.