Data Communication

BCE 6th Sem

Er. Anuj Sherchan
Assistant Professor

Unit 5: Data Link Control and Protocols

- Outline:
- Flow Control
- Data Link Protocols
- A simplex stop and wait protocol
- Sliding window protocols
- One Bit Sliding Window Protocol
- A Protocol Using Go Back N
- A Protocol Using Selective Repeat
- Asynchronous and Synchronous Protocols and their types

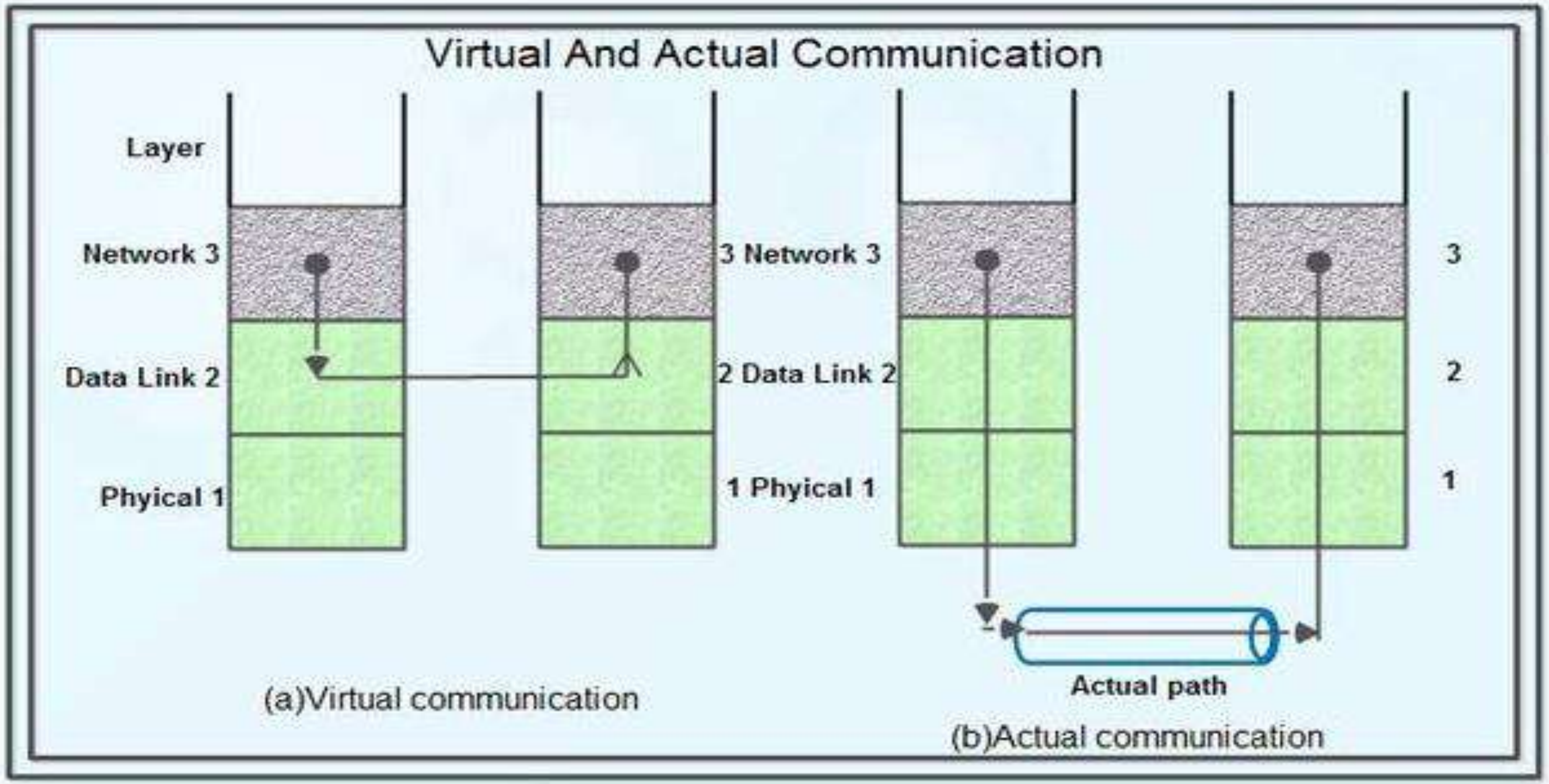
Data Link Layer

- The data link layer is responsible for node to node communication of data frames within the same network.
- This layer divides the stream of data received from the network layer into the manageable data units called frames.
- Specific responsibilities of the data link layer include
- framing,
- physical addressing,
- flow control,
- error control and media access control.

Services Provided to Network Layer

- **Services Provided to Network Layer:**
- Data link layer provides several services to the network layer.
- The one of the major services provided is the transferring the data from network layer on the source machine to the network layer on destination machine.
- On source machine data link layer receives the data from network layer and on destination machine pass on this data to the network layer as shown in Figure.
- The path shown in fig (a) is the virtual path. But the actual path is Network layer -> Data link layer -> Physical layer on source machine, then to physical media and thereafter physical layer -> Data link layer -> Network layer on destination machine.

Services Provided to Network Layer



Data Link Protocols

- **Data Link Protocols**
- Data link layer protocols are mainly responsible for flow control.
- When a data frame is sent from one host to another over a single medium, it is required that the sender and receiver should work at the same speed.
- That is, sender sends at a speed on which the receiver can process and accept the data.
- If sender is sending too fast, the receiver may be overloaded and data may be lost.
- Two types of mechanisms can be deployed to control the flow.
    - A simple stop and wait Protocol
    - Sliding Window Protocol

Data Link Protocols

- **Simplex Stop and Wait**
- This flow control mechanism forces the sender after transmitting a data frame to stop and wait until the acknowledgement of the data-frame sent is received.
- The sender sends the next frame only when it has received a positive acknowledgement from the receiver that it is available for further data processing.
- Data transmission is one directional, but must have bidirectional line.

Simplex Stop and Wait

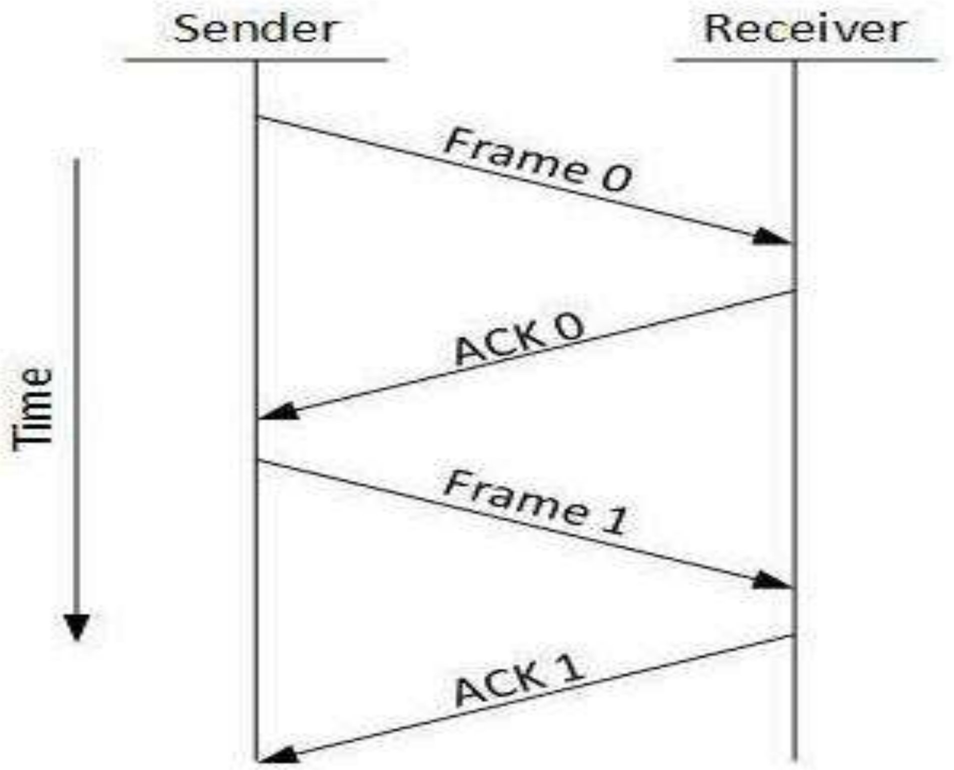


Fig : Simplex Stop and Wait

Stop and Wait ARQ

- **Stop and Wait ARQ**
- The following transition may occur in Stop-and-Wait ARQ:
- The sender maintains a timeout counter.
- When a frame is sent, the sender starts the timeout counter.
- If acknowledgement of frame comes in time, the sender transmits the next frame in queue.
- If acknowledgement does not come in time, the sender assumes that either the frame or its acknowledgement is lost in transit.
- Sender retransmits the frame and starts the timeout counter.
- If a negative acknowledgement is received, the sender retransmits the frame.

Stop and Wait ARQ

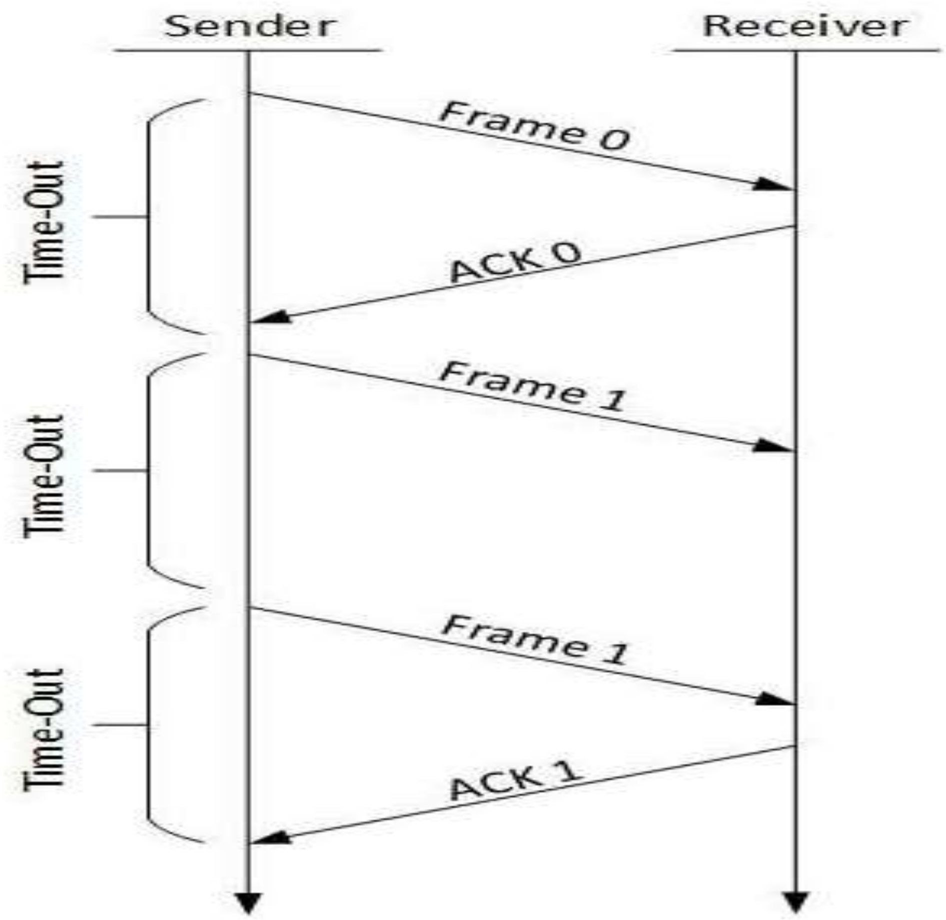
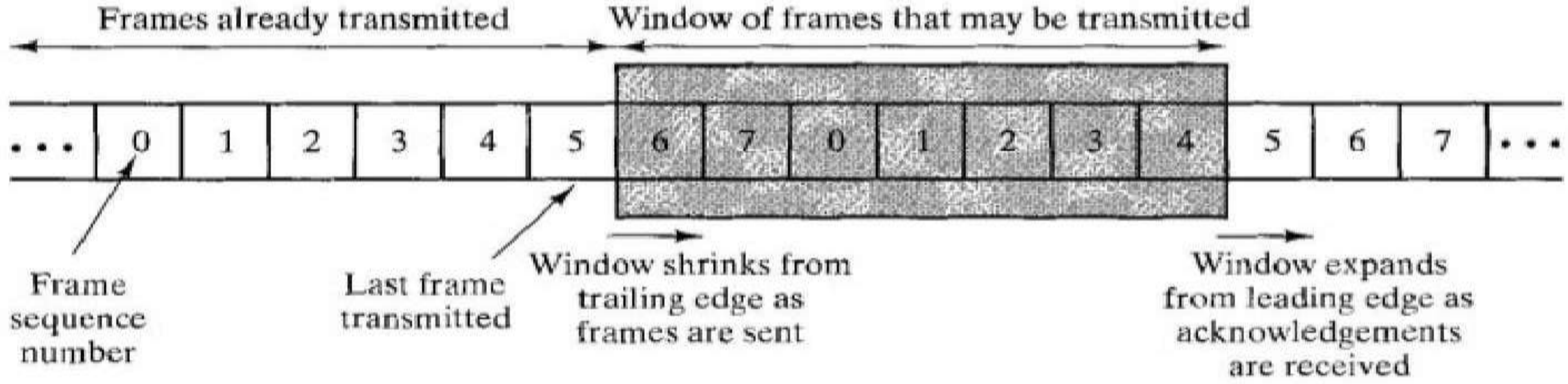


Fig: Stop and Wait ARQ

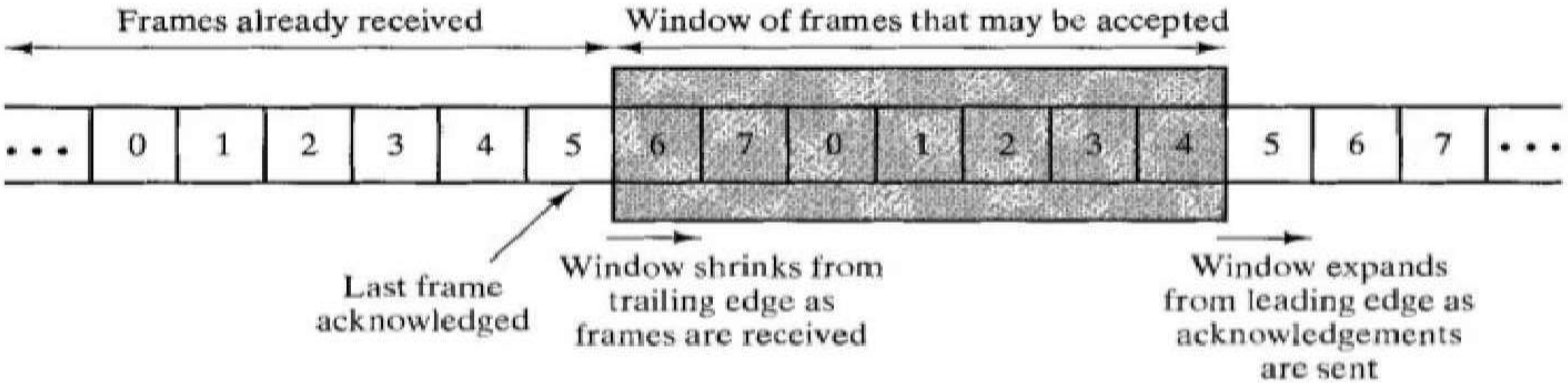
Sliding Window

- **Sliding Window**
- In this flow control mechanism, both sender and receiver agree on the number of data-frames after which the acknowledgement should be sent.
- As we learnt, stop and wait flow control mechanism wastes resources, this protocol tries to make use of underlying resources as much as possible.
- 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.
- 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.

Sliding Window



(a) Transmitter's perspective



Sliding Window

- It determines the maximum number of frames that the sender can send before receiving acknowledgment.
- The types of sliding window protocol include:
  - A One Bit Sliding Window Protocol
  - A Protocol Using Go Back N
  - A Protocol Using Selective Repeat

- Note: ARQ (Automatic Repeat Request) is designed for noisy channels.

Sliding Window

- **One Bit sliding window protocol**
- In one – bit sliding window protocol, the size of the window is 1.
- So, the sender transmits a frame, waits for its acknowledgment, then transmits the next frame.
- Thus, it uses the concept of stop and wait protocol.
- This protocol provides for full – duplex communications.
- Hence, the acknowledgment is attached along with the next data frame to be sent called piggybacking.
- So, it is better compared to stop and wait due to full duplex communications.

Sliding Window

- **Stop and Wait ARQ**
- The source station transmits a single frame and then must await an acknowledgement (ACK).
- No other data frames can be sent until the destination stations reply arrives at the source station.

• The sending device keeps a copy of the last frame until it receives an acknowledgement for that frame.

- Keeping a copy allows the sender to retransmits lost or damaged frames until they are received correctly.

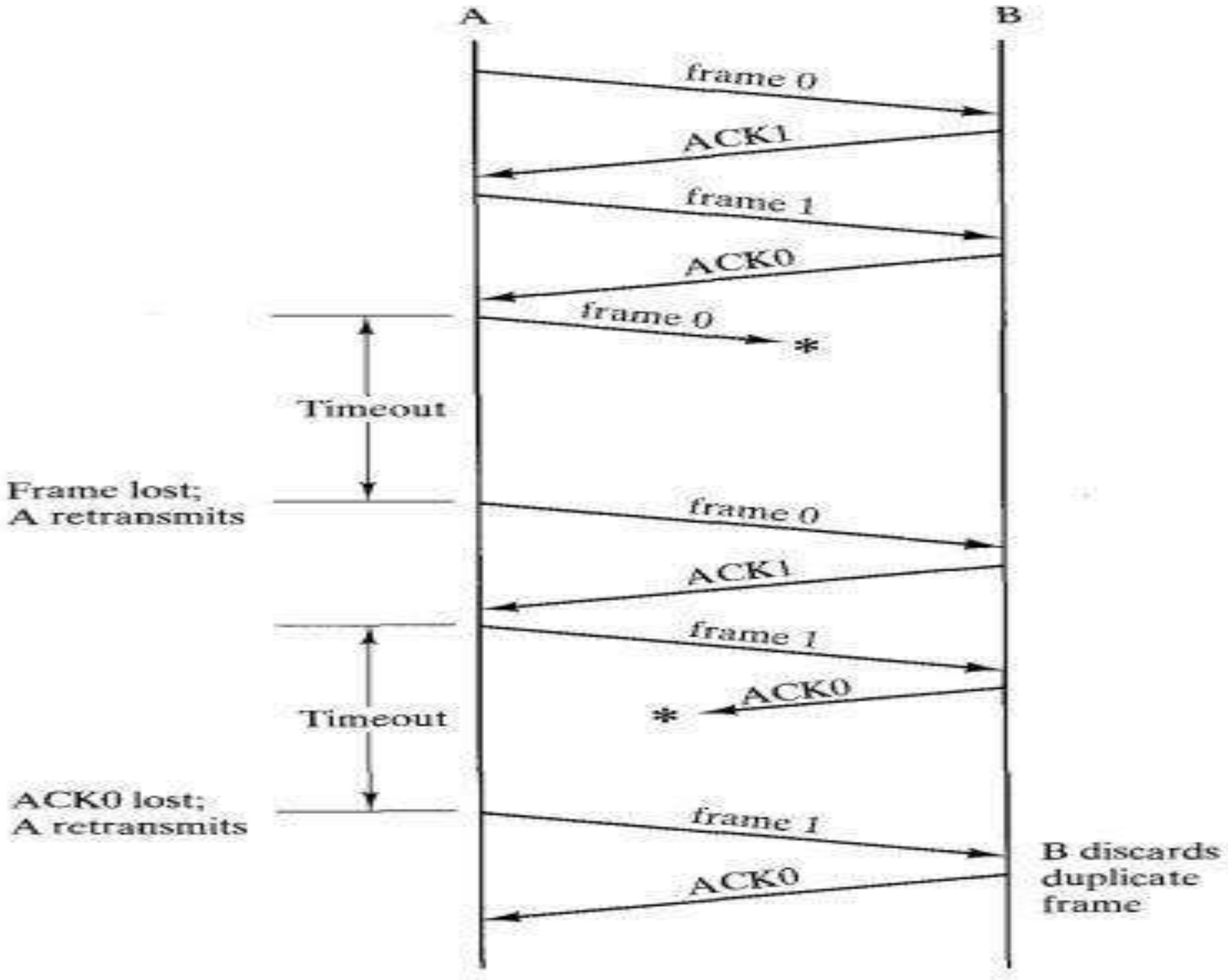
• For identification purposes, both data frames and acknowledgement frames (ACK) are numbered 0 & 1.

- A data 0 frame is acknowledged by and ACK1 frame, indicating that the receiver has received data frame 0 and is now expecting data frame 1.

Stop and Wait ARQ

- The sender starts a timer when it sends a frame.
- If an acknowledgement is not received within an allotted time period, the sender assumes that the frame was lost or damage and resends it.
- The receiver sends only +ve ACK for frame received safe and sound.
- It is silent about the frames damaged or lost.
- The acknowledgement number always define the number of next expected frame.
- If frame 0 is received, ACK1 is sent; if frame 1 is received ACK 0 is sent.

Stop and Wait ARQ



Sliding Window

- **Go-Back-N ARQ**
- In this protocol, we can send several frames before receiving acknowledgements; we keep a copy of these frames until the acknowledgements arrive.
- Stop and wait mechanism does not utilize the resources at their best.
- When the acknowledgement is received, the sender sits idle and does nothing.
- In Go-Back-N method, both sender and receiver maintain a window.
- The sending-window size enables the sender to send multiple frames without receiving the acknowledgement of the previous ones.
- The receiving-window enables the receiver to receive multiple frames and acknowledge them.
- The receiver keeps track of incoming frame’s sequence number.

Go-Back-N ARQ

- When the sender sends all the frames in window, it checks up to what sequence number it has received positive acknowledgement.
- If all frames are positively acknowledged, the sender sends next set of frames.
- If sender finds that it has received NACK or has not receive any ACK for a particular frame, it retransmits all the frames after which it does not receive any positive ACK.

Go-Back-N ARQ

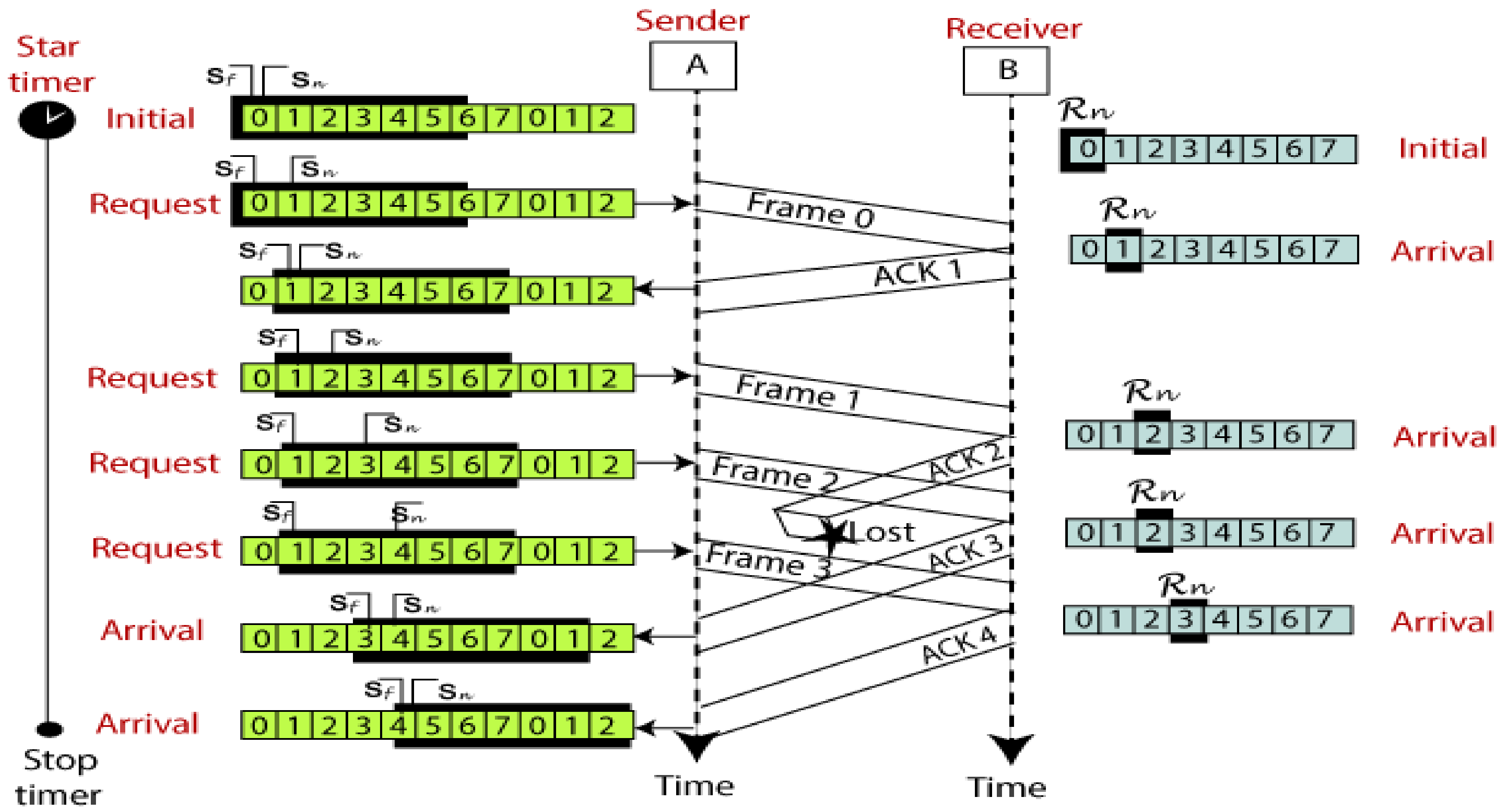


Fig : Go-Back-N ARQ

Sliding Window

- Selective Repeat ARQ
- In Go-back-N ARQ, it is assumed that the receiver does not have any buffer space for its window size and has to process each frame as it comes.
- This enforces the sender to retransmit all the frames which are not acknowledged.
- In Selective-Repeat, the receiver while keeping track of sequence numbers, buffers the frames in memory and sends NACK for only frame which is missing or damaged.
- The sender in this case, sends only packet for which NACK is received.

Selective Repeat ARQ

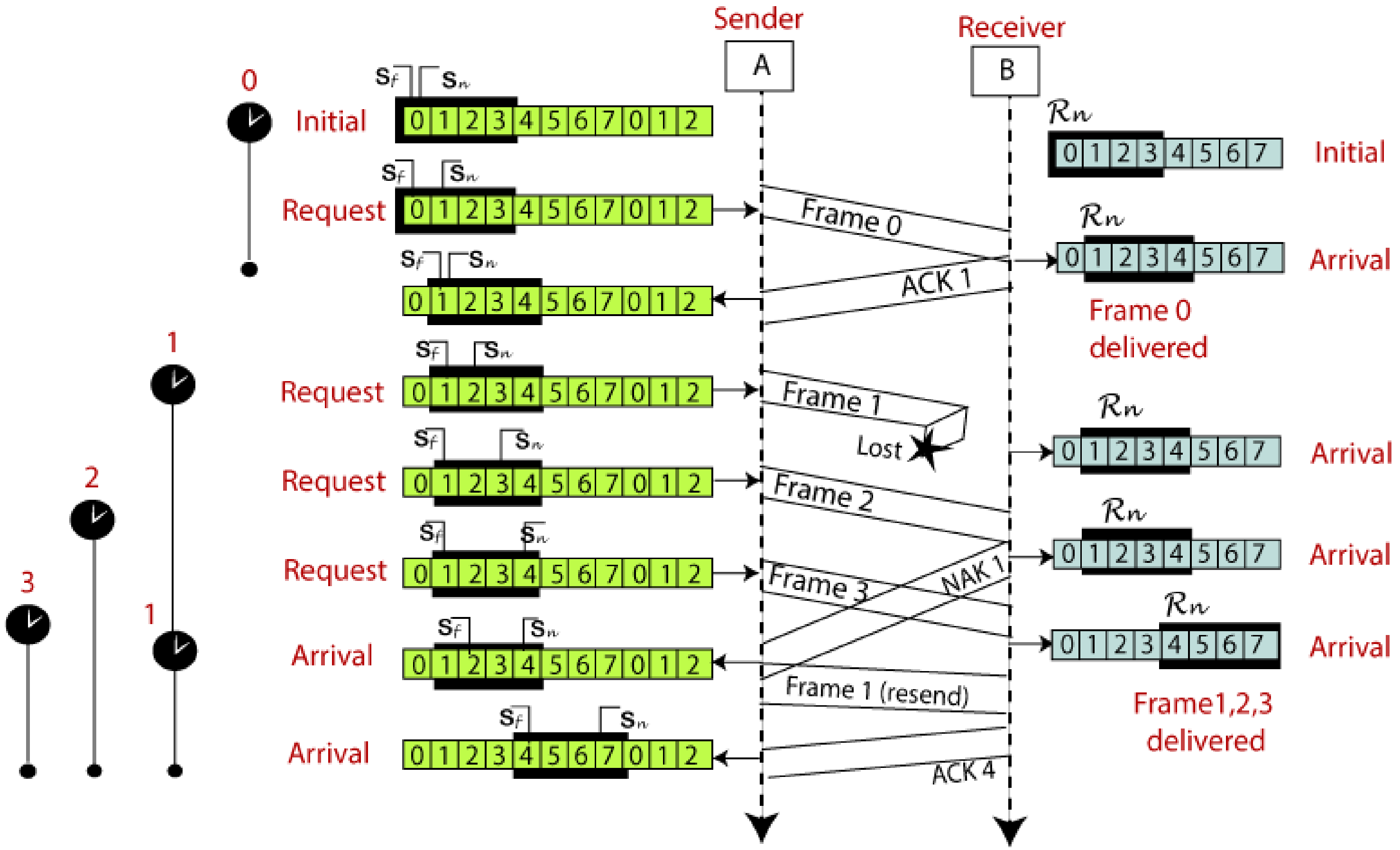


Fig : Selective Repeat ARQ

Asynchronous and Synchronous Protocols

- **Asynchronous Protocols**
- Asynchronous protocols treat each character in a bit stream independently.
- These protocols are used in modems.
- They use start and stop bits, and variable gaps between characters.
- They are slower than synchronous protocols in transmitting data.
- The different asynchronous protocols are:
- XMODEM
- YMODEM
- ZMODEM
- Block Asynchronous Transmission (BLAST)
- Kermit

Asynchronous and Synchronous Protocols

- **XMODEM**
- It is a half duplex stop & wait protocol.
- It is used for telephone line communication between PCs.
- The sender sends a frame to receiver & waits for ACK frame.
- The receiver can send one cancel signal (CAN) to abort the transmission.
- The frame format of XMODEM is:

SOH	Header	Data	CRC
1 Byte	2 Bytes	128 Bytes	

Asynchronous and Synchronous Protocols

- The various fields of the frame are:
- SOH: It is the start of the header. It is a 1-byte field.
- Header: It contains the sequence number. It is 2 bytes in length.
- Data: This field holds 128 bytes of data.
- CRC: It is a Cyclic Redundancy Check. This field checks the errors in the data field.

Asynchronous and Synchronous Protocols

- **YMODEM**
- This protocol is similar to XMODEM with the following major differences:
- Two cancel signals (CAN) are used to abort the transmission.
- The data field is 1024 bytes long.
- ITU-T CRC-16 is used for error checking.
- **ZMODEM**
- It is a combination of XMODEM and YMODEM.

Asynchronous and Synchronous Protocols

- **BLAST**
- BLAST is more powerful than XMODEM.
- It is a full-duplex protocol.
- It uses sliding window flow control.
- **Kermit**
- It is a terminal program as well as a file transfer protocol.
- It is similar in operation to XMODEM, except that sender has to wait for a negative acknowledgment (NAK) before it starts transmission.

Asynchronous and Synchronous Protocols

- **Synchronous Protocols**
- Synchronous Protocols take the whole bit stream and divide it into characters of equal size.
- These protocols have high speed and are used for LAN, WAN and MAN.
- Synchronous protocols are categorized into two groups:
- Character-Oriented Protocol
- Bit-Oriented Protocol

Asynchronous and Synchronous Protocols

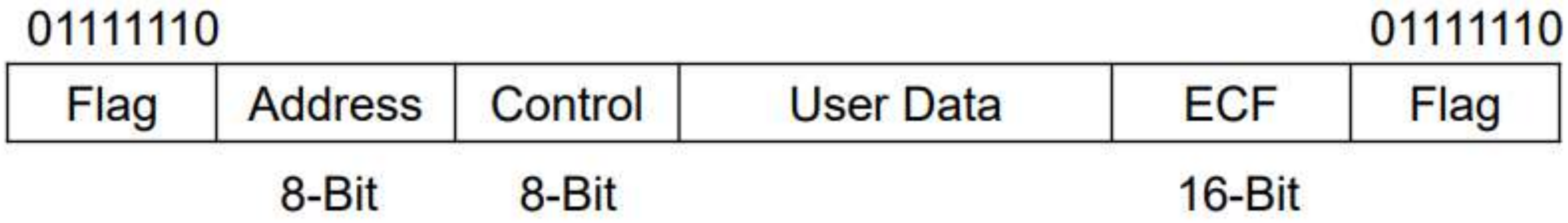
- **Character-Oriented Protocol**
- It interprets frame as a series of characters.
- These are also known as Byte-Oriented Protocols.
- Control information is inserted as separate control frames or as an addition to existing data frames.
- The example of a character-oriented protocol is Binary Synchronous Communication (BSC) developed by IBM.
- **Bit-Oriented Protocol**
- It interprets a frame as a series of bits.
- Control information can be inserted as bits depending on the information to be contained in the frame

Asynchronous and Synchronous Protocols

- bit-oriented protocol can pack more information into shorter frames.
- Examples of the bit-oriented protocol are
- Synchronous Data Link Control (SDLC)
- High-Level Data Link Control (HDLC)
- **Synchronous Data Link Control (SDLC)**
- SDLC protocol was developed by IBM in 1975.
- After developing SDLC, IBM submitted it to American National Standard Institute (ANSI) and to International Standard Organization (ISO) for acceptance.
- ANSI modified it to ADCCP (Advanced Data Communication Control Procedure).

Asynchronous and Synchronous Protocols

- ISO modified it to HDLC (High-Level Data Link Control).
- Synchronous Data Link Control (SDLC) Protocol
- The frame format of SDLC is:



Asynchronous and Synchronous Protocols

- The flag sequence of 8-bits 01111110 marks the beginning and end of the frame.
- Address field contains the address of the receiver.
- Control field carries the sequence number, acknowledgment, requests, and responses.
- The user data field carries the data and is of variable length.
- ECF stands for Error Checking Field and is of 16- bits.
- It is used for error control.

Asynchronous and Synchronous Protocols

- **High Level Data Link Control (HDLC) Protocol**
- HDLC came into existence after ISO modified the SDLC protocol.
- It is a bit-oriented protocol that supports both half and full duplex communication.
- Systems using HDLC are characterized by:
- Station Types
- Configuration.
- Response Modes

Asynchronous and Synchronous Protocols

- **Station Types**
- To make HDLC protocol applicable to various network configurations, three types of stations have been defined:
- Primary Station
- Secondary Station
- Combined Station

Asynchronous and Synchronous Protocols

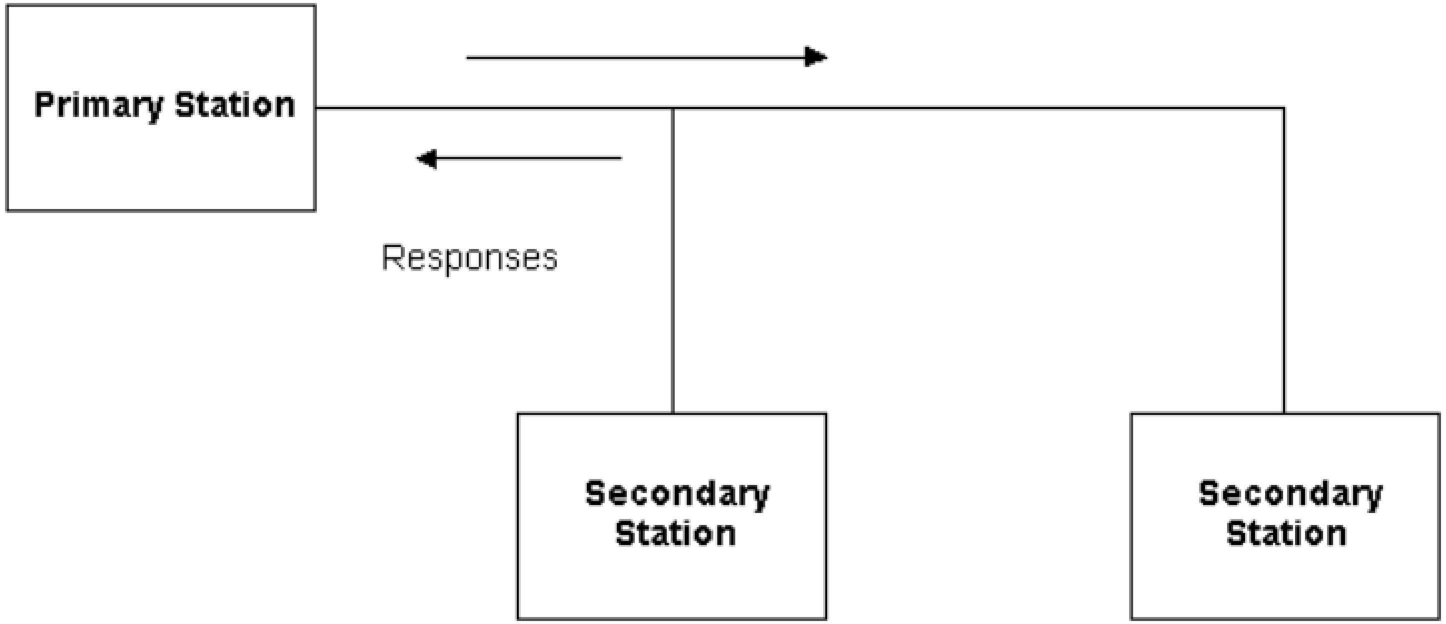
- **Primary Station**
- It has complete control over the link at any time.
- It has the responsibility of connecting & disconnecting the link.
- The frames sent by the primary station are called commands.
- **Secondary Station**
- All the secondary stations work under the control of the primary station.
- The frames sent by the secondary station are called responses.
- **Combined Station**
- A combined station can behave either as a primary or as a secondary station.
- It can send commands as well as responses.

Asynchronous and Synchronous Protocols

- **Configuration**
- Configuration defines how the various stations are connected to a link.
- There are three possible configurations:
- Unbalanced Configuration
- Symmetrical Configuration
- Balanced Configuration

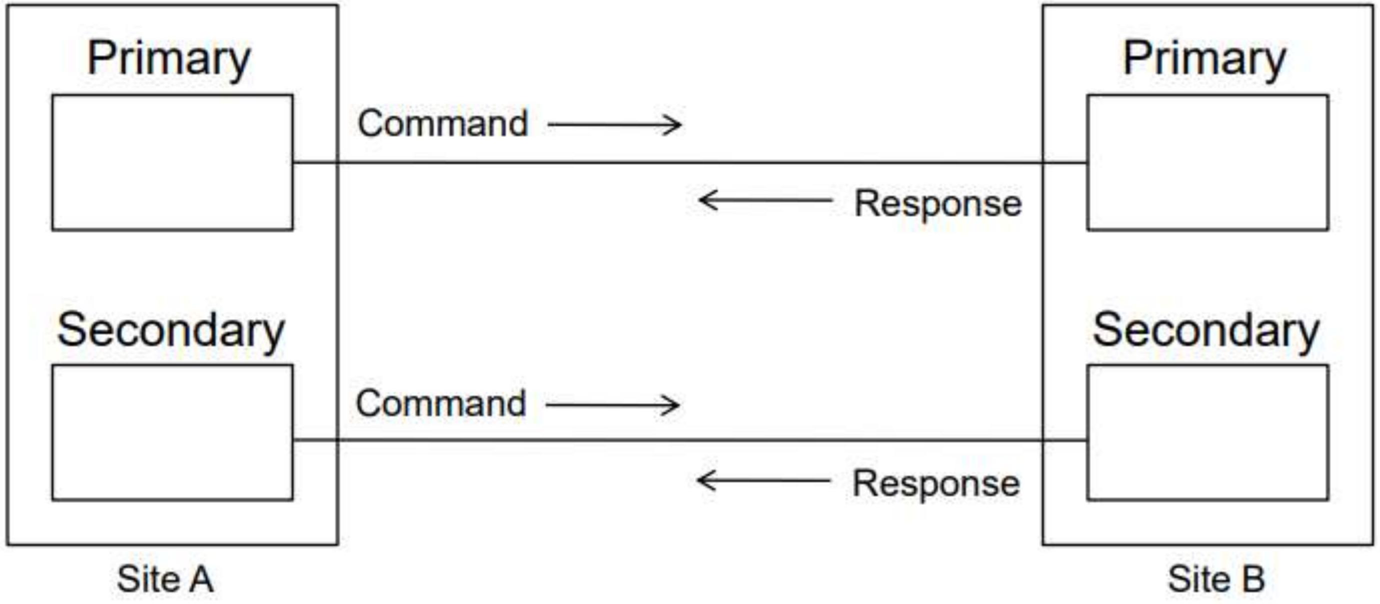
Asynchronous and Synchronous Protocols

- **Unbalanced Configuration**
- This type of configuration exists if one station is primary and the other is secondary.



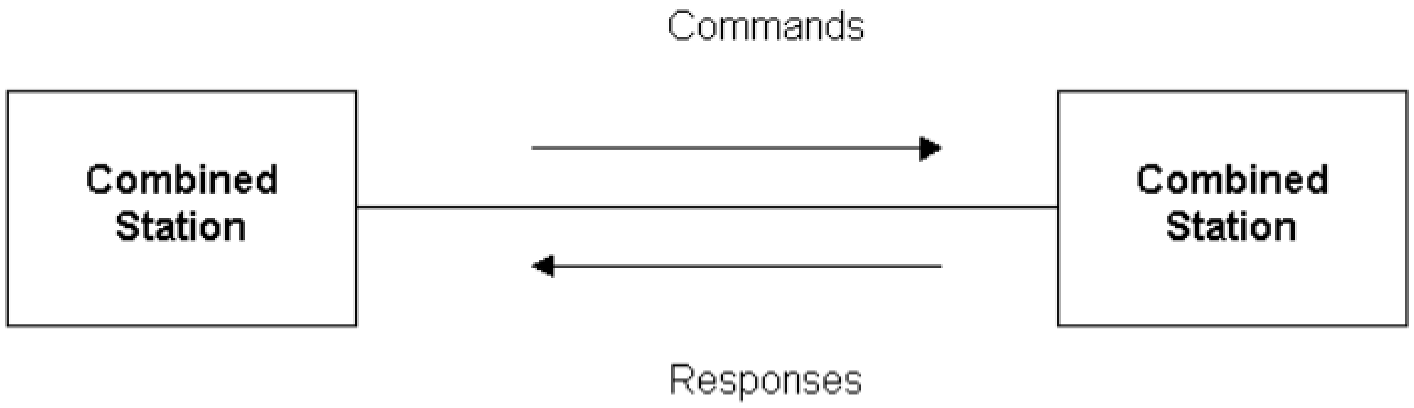
Asynchronous and Synchronous Protocols

- **Symmetrical Configuration**
- In this configuration, both sites contain two stations: one primary and one secondary.
- Primary station of one site is linked with the secondary station of the other and vice versa.



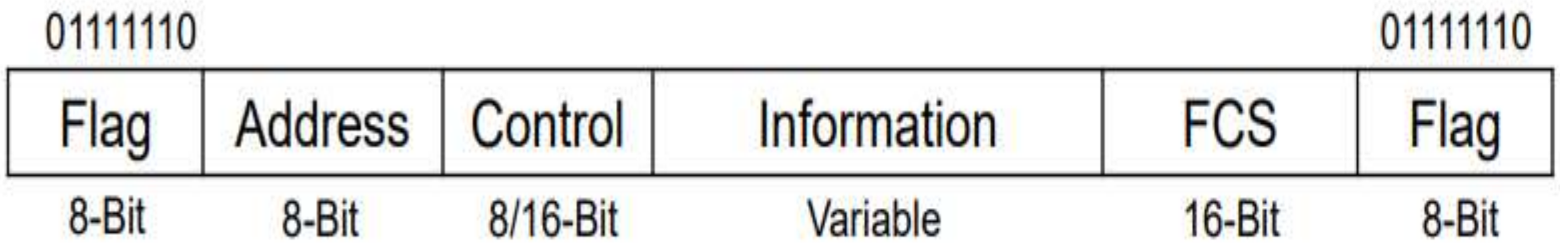
Asynchronous and Synchronous Protocols

- **Balanced Configuration**
- In this configuration, both sites have combined stations.
- These combined stations are connected with a single link.
- This single link can be controlled by either station.



Asynchronous and Synchronous Protocols

- **Frame Structure in HDLC**
- Frame in HDLC can have six fields:



Asynchronous and Synchronous Protocols

- Flag Field: It is the 8-bit field that contains 01111110.
- It marks the beginning and end of a frame.
- Address Field: This field contains the address of the receiver.
- It is 8-bit long.
- Control Field: It carries the sequence number, acknowledgments, requests, and responses.
- It can be 8-bit or 16-bit.
- Information Field: It contains user data.
- Its length is different for different networks.

Asynchronous and Synchronous Protocols

- FCS Field: FCS stands for Frame Check Sequence.
- It is the error detection field and is 16-bit long.
- It contains either 16-bit CRC or 32-bit CRC.
- **Types of Frames in HDLC**
- HDLC defines three types of frames:
- **Information Frame (I-Frame)**
- I-Frames carry user data, and control information about user’s data.
- **Supervisory Frame (S-Frame)**
- S-Frames carry flow & error control information.

Asynchronous and Synchronous Protocols

- **Unnumbered Frame (U-Frame)**
- U-Frames are reserved for system management.
- They are used to exchange session management & control information between the two connected devices.

THANK YOU