Unit 4
Data Link Layer

Error Correction

Once error has been detected, it must be corrected. The simplest, most effective, least expensive, and most commonly used method for error correction are retransmission. With retransmission, a receiver that detects an error simply asks the sender to retransmit the message until it is received without error. It includes:

1. Automatic Repeat reQuest (ARQ)

Once error has been detected, it must be corrected. The simplest, most effective, least expensive, and most commonly used method for error correction is retransmission. With retransmission, a receiver that detects an error simply asks the sender to retransmit the message until it is received without error. This is often called **Automatic Repeat reQuest (ARQ).** There are three versions of ARQ. They are:

1) **Stop and wait ARQ**
2) **Go back 'N'ARQ**
3) **Selective-Repeat ARQ (retransmission)**

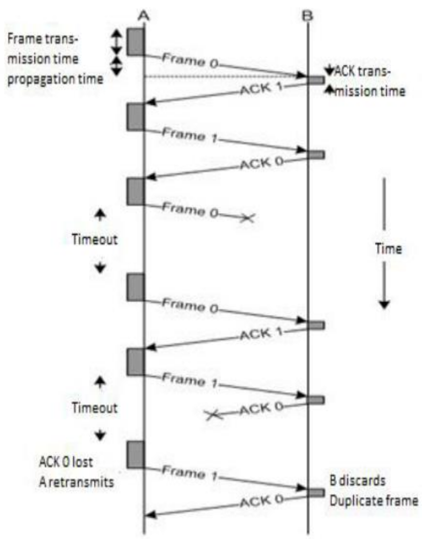
Stop and wait ARQ

It is a technique used to retransmit the data in case of damaged or lost frames. The source station transmits a single frame and then most awaits an acknowledgement from the destination station.

There are two possible errors:

First, the frame that arrive at the destination would be damaged which is detected by receiver by using error detection techniques and discard the frame. After a frame is transmitted a source station waits for an ACK. If no ACK is received by the time that the timer expires and the same frame is send again. The transmitter always maintains a copy of transmitted frame until the time out or ACK is received.

The second type of error is the damage ACK. For the successful reception of frame, if the acknowledgment is damaged in transit which is therefore time out and resend the same frame. To avoid the duplication of frame they are alternatively labeled with '0' and '1' and positive ACK are of the form ACK0 and ACK1. As in sliding window convention, an ACK0 acknowledgement receipt of a frame numbered '1' and indicates that the receiver is ready for a frame number '0'. This system is simple but inefficient.



Go Back N ARQ

In this protocol multiple numbers of frames can be transmitted without waiting of ACK. A copy of each transmitted is maintained until the receptive ACK is received.

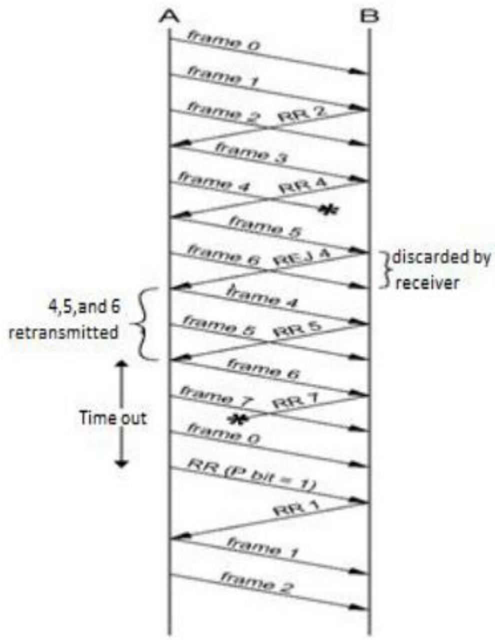
This is based on the sliding window protocol. If the receiver detects any error in the frame, it sends negative ACK (REJ) for that frame. The receiver will discard all the future incoming frames until the frame error is correctly received. Thus source transmitter must go back and retransmit that frame and all subsequent frames.

Advantages

1. Sender can send multiple frame at a time
2. More efficient than stop and wait ARQ

Disadvantages

1. Buffer requirement
2. If negative ACK is lost, there will be unnecessary transmission of frames until time out which requires retransmission of such frames.



Selective Repeat ARQ (retransmission)

In this protocol only the rejected frames are transmitted. Suppose F4 is rejected then the subsequent frames are accepted by the receiver and store in the buffer until the valid F4 is received.

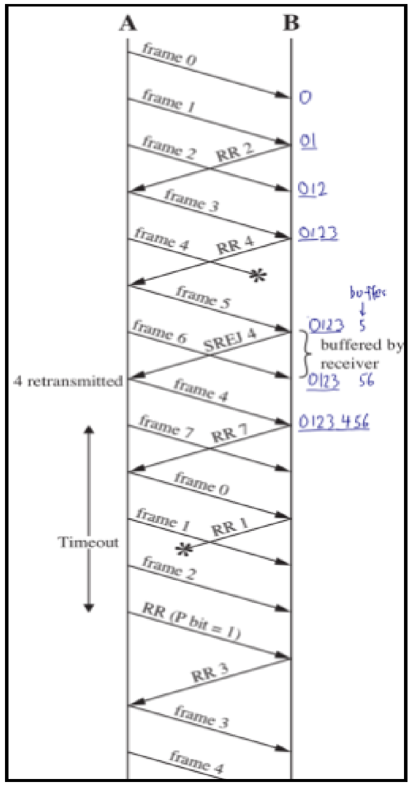
This is more efficient than go back N ARQ because it minimizes the amount of retransmission. But the receiver must maintain the buffering capacity large enough to save the frame until the frame in error is retransmitted and must contain logic for re-inserting that frame in the proper sequence.

Advantage

More efficient than go back NARQ (minimization of retransmission)

Disadvantage

Receiver requires larger buffer size.

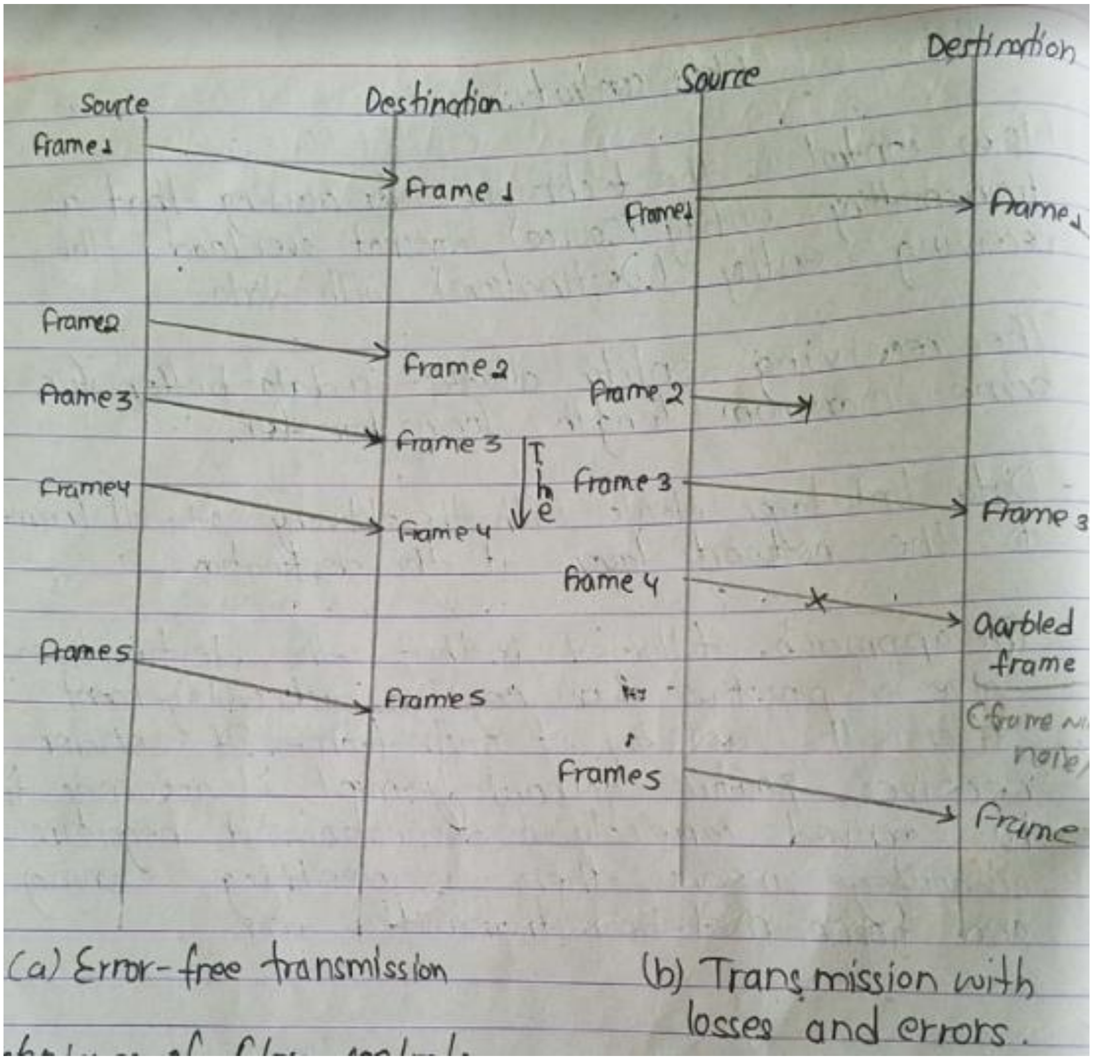


2. **Forward Error Correction**
   Consult your book for this topic

Flow Control

Flow control is the technique for assuring that a transmitting entity (source) does not overload the receiving entity (destination) with data. The receiving entity allocates a data buffer for some maximum length for a transfer. Data link layer deals with the delivery of all frames to the network layer at the destination.

The approach followed is that the destination gives a positive or negative acknowledgement after the delivery of each frame. If sender receives positive ACK, it assumes frame has arrived correctly at destination. A negative ACK means there is something wrong and frame must be re-transmitted.



Types of flow control

A. Stop and Wait Flow Control

B. Sliding Window Flow Control

Stop and Wait Flow Control

Stop and wait flow control works under the following assumption:

- Data transmission in one direction only.
- Channel is error free.

The simple form of flow control is known as stop and wait flow control.

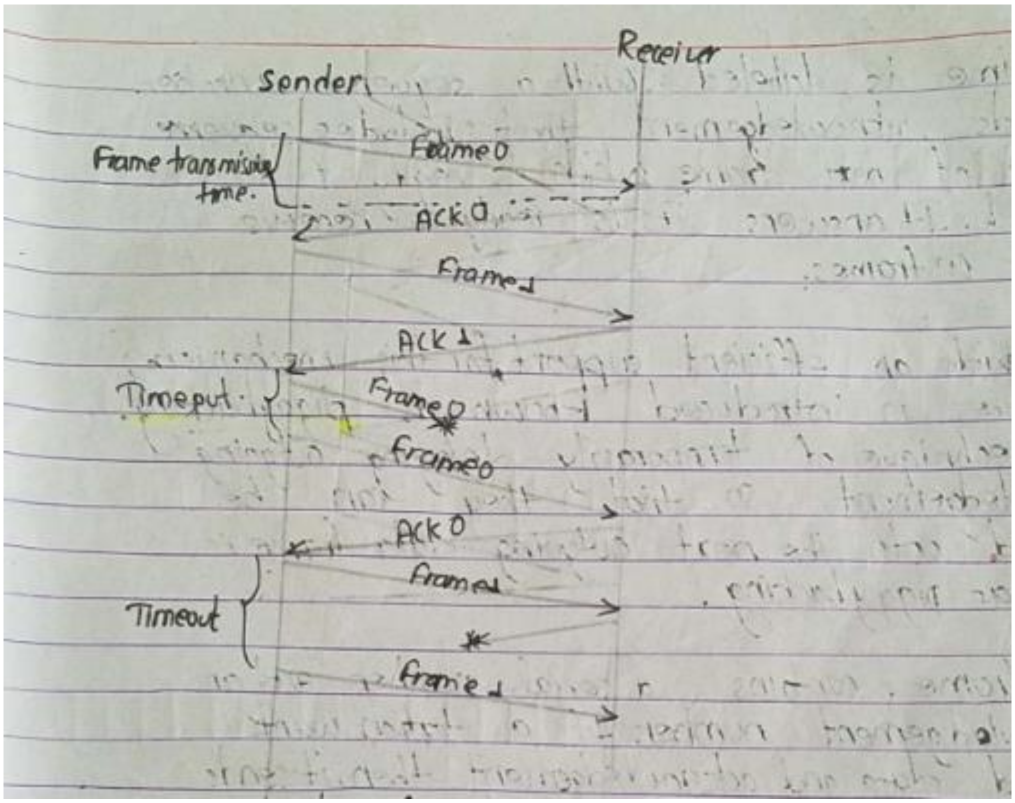
Process

- ➢ A source transmit a frame at first, after the destination receives the frame, it indicates its willingness to accept another frame of data by sending back an acknowledgement to the frame just arrived. The source must wait until it receives the acknowledgement before sending the next frame of data. The destination can stop the flow of data simply by holding ACK.

- Receiver sends a positive ACK to sender to transmit the next data frame. Error free channel is assumed. Data frames are transmitted in one direction where each frame is individually acknowledged by the receiver by a separate ACK frame.
- The sender transmits one frame, starts a timer and waits for an ACK from receiver before sending further frames. A time out period is used where frames are not acknowledged by the receiver and are retransmitted automatically by the sender.
- Frames received which are damaged are not acknowledged and are retransmitted by the sender when time out.
- A one bit sequence 0 or 1 is used to distinguish between original data frames and duplicate retransmitted frames which are to be discarded.

Disadvantages

- At a time only one frame can be transmitted.
- No full utilization of the particular channel.
- Single message is sent in different frames so there is the higher chance of error.
- If only one frame collides then again whole frame has to be transmitted.

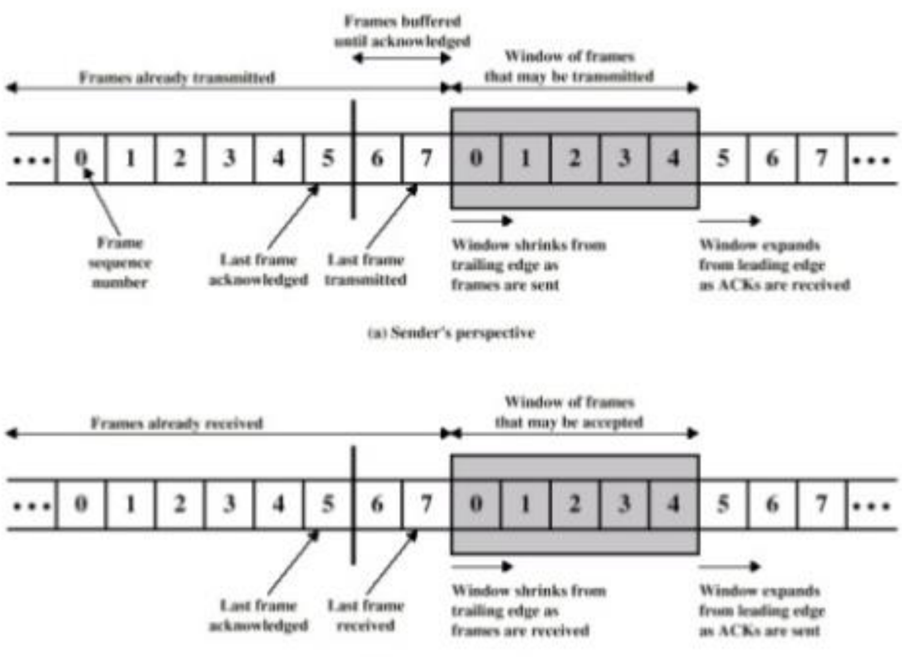


Sliding Window Flow Control

In this mechanism of flow control, the system is full duplex and can transmit multiple frames at a same time.

Working Mechanism

- Two work station 'A' and 'B' are connected by full duplex link. If two station exchange data they need to maintain two windows; one for transmitting the frame and another for receiving the frame. 'B' allocates space for W frames and 'A' sends W frames without getting acknowledged.
- To keep the track of which frame has been acknowledged each frame is labeled with a sequence number. 'B' sends acknowledgement that includes sequence number of next frame which is ready to be received. It answers 'B' is ready to receive next W frames.
- To provide an efficient support for the mechanism, a feature is introduced known as piggybacking. The technique of temporarily delaying outgoing acknowledgement so that they can be backed onto the next outgoing data frame is known as piggybacking.
- Each frame contains a serial number and an acknowledgement number. If a station wants to send data and acknowledgment then it sends both frames so that communication capacity can be saved and utilized. If the station has acknowledgement but not the data, it sends separate acknowledgement frames as RR (Ready to Receive) and RNR (Received Not Ready). If the station has to send the data but no new acknowledgement, then it must repeat ACK sequence number.



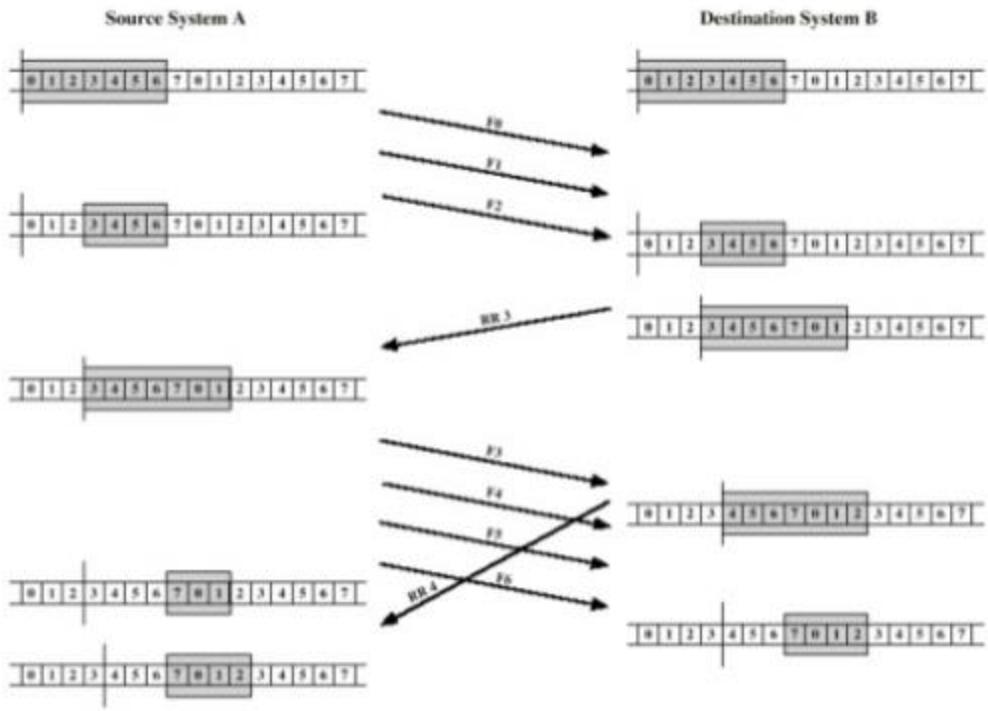
(b) Receiver's perspective

Advantages

- Multiple frames sent at a time
- Better than stop and wait flow control.

Disadvantages

- The system is very complex.



Explanation

1. Initially ‘A’ and ‘B’ have windows indicating that ‘B’ can receive 7 frames and ‘A’ may transmit 7 frames
2. After transmitting three frames F_0, F_1 and F_2 without acknowledgement, ‘A’ has shrunk (reduce) its window to four frames and maintains a copy of three transmitted frames (F_0, F_1 and F_2).
3. The window indicates that ‘A’ can transmit four frames beginning with frame F_3.
4. ‘B’ then transmits RR3 (Receive Ready Frame 3) which mean I have received all frames up to F_2 and I am ready to receive F_3 i.e. I am prepared to receive seven frames beginning with F_3
5. With this acknowledgement, ‘A’ is backup to transmit seven frames beginning with F_3 and ‘A’ discard the buffer frame which has been discarded.
6. A proceed to transmit F_3, F_4, F_5 and $F_{6.}$
7. B acknowledge F_3 by sending RR4 and allows transmission of F_4 by the time RR4 reaches A, it has already transmitted F_4, F_5 and F_6 and therefore A open its window to permit sending four frames beginning with $F_{7.}$

Data Link Protocols

The different types of protocols used in Datalink layer are listed in the figure below. The figure includes the information about protocol, error detection technique, retransmission technique and media access:

Protocol	Size	Error Detection	Retransmission	Media Access
Asynchronous transmission	1	Parity	Continuous ARQ	Full Duplex
Synchronous protocols				
SDLC	*	16-bit CRC	Continuous ARQ	Controlled Access
HDLC	*	16-bit CRC	Continuous ARQ	Controlled Access
Ethernet	*	32-bit CRC	Stop-and-wait ARQ	Contention
PPP	*	16-bit CRC	Continuous ARQ	Full Duplex

*Varies depending on the message length.

ARQ = Automatic Repeat reQuest; CRC = cyclical redundancy check; HDLC = high-level data link control; PPP = Point-to-Point Protocol; SDLC = synchronous data link control.

Fig: Data Link Layer Protocol Summary

Asynchronous Transmission

Asynchronous transmission is often referred to as *start-stop transmission* because the transmitting computer can transmit a character whenever it is convenient, and the receiving computer will accept that character. It is typically used on point-to-point full-duplex circuits (i.e., circuits that have only two computers on them), so media access control is not a concern.

It is a mode of serial transmission in which the data is transmitted as a continuous stream of bytes separated by starts and stop bits. Data is transmitted from source to destination frame by frame (character by character). Each frames/ character begins with a **start bit** which is binary '**0**' and end with **stop bit** which is binary **'1'**.Data bits are usually followed by **parity bit** for **error checking** .

In asynchronous communication, only about 80 percent of the transmitted bits actually contain data, while the other 20 percent contain signaling information in the form of start bit and stop bit and parity bit. An 8-bit character requires 3 bits of control information (start, stop, and parity bits).The bit of the character are transmitted beginning with LSB (Least Significant Bit)

The transmission is asynchronous at frame level (character level) but is still synchronized at bit level during the reception of data.

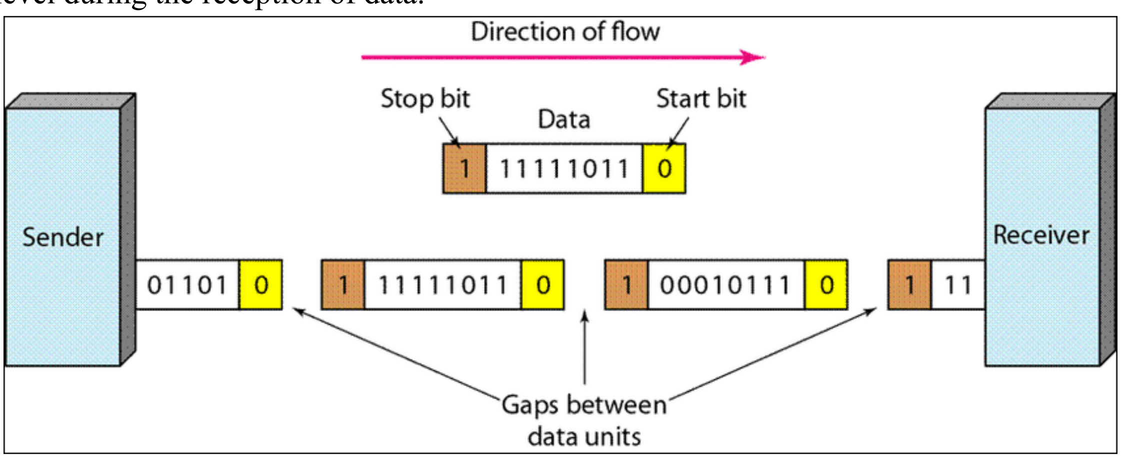
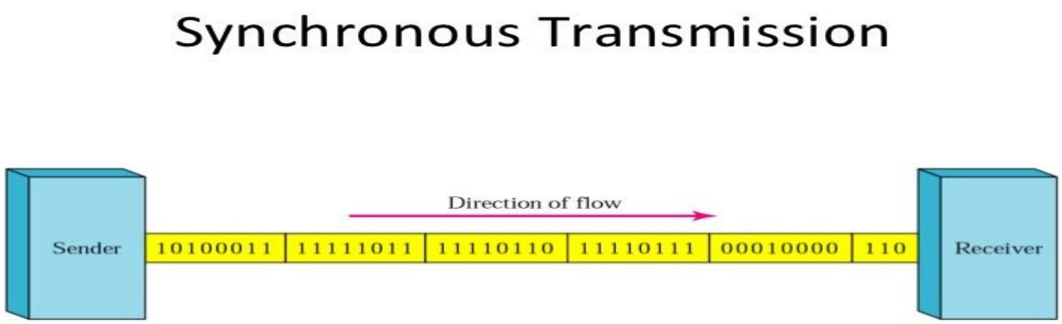


Fig: Asynchronous Transmission

Synchronous Transmission

It is a mode of serial transmission in which continuous stream of data in the form of signal is transmitted accompanied by regular timing signals which are generated by some external clocking mechanism which ensures both sender and receiver are synchronized with each other. The block of data is in the form of bit stream and is transferred without start and stop bit. The transmission is good over short distance communication. It uses preamble and postamble each of 8 bits to leave sufficient space between the blocks.It is more efficient than asynchronous transmission.

The advantage of synchronous transmission is **speed.** With no extra bits or gaps introduced at the source end and remove at the receiving end and by extension with fewer bits to move across the link, synchronous transmission is faster than asynchronous transmission. Thus it is used in high speed applications.



There are many protocols for synchronous transmission. Some of them are:

1) **Synchronous Data Link Control (SDLC)**
2) **High – Level Data Link Control (HDLC)**
3) **Ethernet**
4) **Point-to point Protocol (PPP)**

Synchronous Data Link Control (SDLC)

Synchronous data link control *(SDLC)* is a mainframe protocol developed by IBM in 1972 that is still in use today. It uses a controlled-access media access protocol. Each SDLC frame begins and ends with a special bit pattern (01111110), known as the **flag**. The **address field** identifies the destination. The length of the address field is usually 8 bits but can be set at 16 bits; all computers on the same network must use the same length. The **control field** identifies the kind of frame that is being transmitted, either information or supervisory. An **information frame** is used for the transfer and reception of messages, frame numbering of contiguous frames, and the like. A **supervisory frame** is used to transmit acknowledgments (ACKs and NAKs). The **message field** is of variable length and is the user's message. The **frame check sequence field** is a 32-bit CRC code.

Flag	Address	Control	Message	Frame check sequence	Flag
8 bits	8 bits	8 bits	Variable length	32 bits	8 bits

High – Level Data Link Control (HDLC)

High-level data link control (HDLC) is a formal standard developed by the ISO often used in WANs. The current standard for HDLC is ISO 13239. HDLC is essentially the same as SDLC, except that the address and control fields can be longer. HDLC also has several additional benefits such as a larger sliding window for continuous ARQ. It uses a controlled-access media access protocol. HDLC provides both connection-oriented and connectionless service.

Ethernet

Ethernet is a very popular LAN protocol, conceived by Bob Metcalfe in 1973 and developed jointly by Digital, Intel, and Xerox in the 1970s. Since then, Ethernet has been further refined and developed into a formal standard called **IEEE 802.3**ac. There are several versions of Ethernet in use today. Ethernet uses a **contention media access** protocol.

There are several standard versions of Ethernet. Figure below shows an Ethernet 803.3ac frame. The frame starts with a **7-byte preamble** which is a repeating pattern of ones and zeros (10101010). This is followed by a **start of frame delimiter**, which marks the start of the frame. The **destination address** specifies the receiver, whereas the **source address** specifies the sender. The **length** indicates the length in 8-bit bytes of the message portion of the frame. The *VLAN tag* field is an optional 4-byte address field used by virtual LANs (VLANs). The **DSAP destination service access point** and **SSAP source service access point** are used to pass control information between the sender and receiver. These are often used to indicate the type of network layer protocol the packet contains. The **control field** is used to hold the frame sequence numbers and ACKs and NAKs used for error control, as well as to enable the data link layers of communicating computers to exchange other control information. In most cases, the control field is 1-byte long. The maximum length of the message is about 1,500 bytes. The frame ends with a CRC-32 **frame check sequence** used for error detection.

Preamble	Start of Frame	Destination Address	Source Address	VLAN Tag	Length	DSAP	SSAP	Control	Data	Frame Check Sequence
7 bytes	1 byte	6 bytes	6 bytes	4 bytes	2 bytes	1 byte	1 byte	1-2 bytes	46-1,500 bytes	4 bytes

Fig: Ethernet 802.3 frame layout

Point-to point Protocol (PPP)

Point-to-Point Protocol (PPP) was developed in the early 1990s and is often used in WANs. It is designed to transfer data over a point-to-point circuit but provides an address so that it can be used on multipoint circuits. Figure below shows the basic layout of a PPP frame, which is very similar to an SDLC or HDLC frame. The frame starts with a **flag**, and has a one-byte address (which is not used on point-to-point circuits). The **control field** is typically not used. The **protocol field** indicates what type of data packet the frame contains (e.g., an IP packet). The **data field** is variable in length and may be up to 1,500 bytes. The **frame check sequence** is usually a CRC-16, but can be a CRC-32. The frame ends with a **flag**.

Flag	Address	Control	Protocol	Data	Frame Check Sequence	Flag
1 byte	1 byte	1 byte	2 bytes	Variable Length	2 or 4 bytes	1 byte

Fig: PPP frame layout

Transmission Efficiency.

The objective of a data communication network is to move the highest possible volume of accurate information through the network. The higher the volume, the greater the resulting network's efficiency and the lower the cost. Network efficiency is affected by characteristics of the circuits such as error rates and maximum transmission speed, as well as by the speed of transmitting and receiving equipment, the error-detection and control methodology, and the protocol used by the data link layer. Each protocol we discussed uses some bits or bytes to delineate the start and end of each message and to control error. These bits and bytes are necessary for the transmission to occur, but they are not part of the message. They add no value to the user, but they count against the total number of bits that can be transmitted. Each communication protocol has both information bits and overhead bits. **Information bits** are those used to convey the user's meaning. **Overhead bits** are used for purposes such as error checking and marking the start and end of characters and packets.

Transmission efficiency is defined as the total number of information bits (i.e., bits in the message sent by the user) divided by the total bits in transmission (i.e., information bits plus overhead bits).

For example, let's calculate the transmission efficiency of asynchronous transmission. Assume we are using 7-bit ASCII. We have 1 bit for parity, plus 1 start bit and 1 stop bit. Therefore, there are 7 bits of information in each letter, but the total bits per letter is 10 (7 + 3). The efficiency of the asynchronous transmission system is 7 bits of information divided by 10 total bits, or 70%. We can improve efficiency by reducing the number of overhead bits in each message or by increasing the number of information bits. For example, if we remove the stop bits from asynchronous transmission, efficiency increases to 7⁄9, or 77.8%.

The same basic formula can be used to calculate the efficiency of synchronous transmission. For example, suppose we are using SDLC. The number of information bits is calculated by determining how many information characters are in the message. If the message portion of the frame contains 100 information characters and we are using an
8-bit code, then there are 100×8=800 bits of information. The total number of bits is the 800 information bits plus the overhead bits that are inserted for delineation and error control. Figure of SDLC above shows that SDLC has a beginning flag (8 bits), an address (8 bits), a control field (8 bits), a frame check sequence (assume we use a CRC-32 with 32 bits), and an ending flag (8 bits). This is a total of 64 overhead bits; thus, efficiency is 800/(800 + 64) = 92.6%. This example shows that synchronous networks usually are more efficient than asynchronous networks and some protocols are more efficient than others.

The general rule is that the larger the message field, the more efficient the protocol.