
# KTU NOTES

The learning companion.

**KTU STUDY MATERIALS | SYLLABUS | LIVE NOTIFICATIONS | SOLVED QUESTION PAPERS**

Data Link layer Design Issues – Flow Control and ARQ techniques. Data link Protocols – HDLC. DLL in Internet. MAC Sub layer – IEEE 802 FOR LANs & MANs, IEEE 802.3, 802.4, 802.5. Bridges - Switches – High Speed LANs - Gigabit Ethernet. Wireless LANs - 802.11 a/b/g/n, 802.15.PPP

## DATA LINK LAYER DESIGN ISSUES

The data link layer uses the services of the physical layer to send and receive bits over communication channels.
It has a number of functions, including:

1. Providing a well-defined service interface to the network layer.
2. Dealing with transmission errors.
3. Regulating the flow of data so that slow receivers are not swamped by fast senders.

To accomplish these goals, the data link layer takes the packets it gets from the network layer and encapsulates them into **frames** for transmission.
**Each frame** contains:

a frame header,
a payload field for holding the packet,
a frame trailer

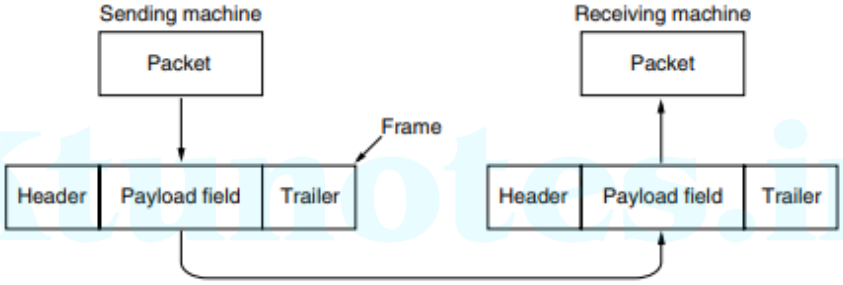



**Figure 3-1.** Relationship between packets and frames.

**Frame management** forms the heart of what the data link layer does.

### Function1 : service provided to network layer

The principal service is transferring data from the network layer on the source machine to the network layer on the destination machine.
various services offered are:

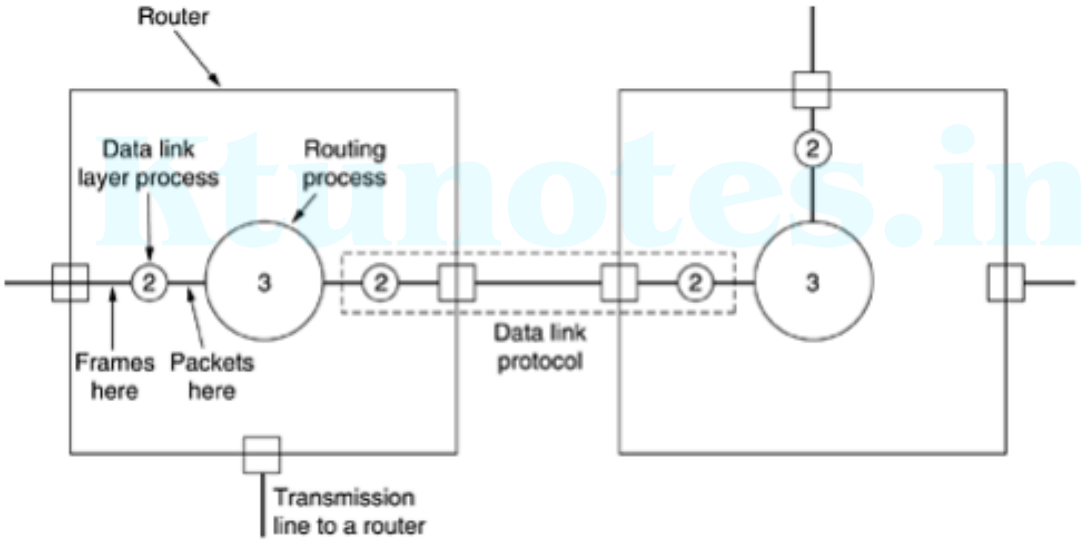
1. Unacknowledged connectionless service.
2. Acknowledged connectionless service.
3. Acknowledged connection-oriented service

1. **Unacknowledged connectionless service** consists of having the source machine send independent frames to the destination machine without having the destination machine acknowledge them. Eg : Ethernet. No logical connection is established beforehand or released afterward. If a frame is lost due to noise on the line, attempt is made to detect the loss or recover from it in the data link layer. This class of service is appropriate when the error rate is very low, so recovery is left to higher layers. It is also appropriate for real-time traffic, such as voice, in which late data are worse than bad data.

2. **Acknowledged connectionless service.** there are still no logical connections used, but each frame sent is individually acknowledged. The sender knows whether a frame has arrived correctly or been lost. If it has not arrived within a specified time interval, it can be sent again. This service is useful over unreliable channels, such as wireless systems. 802.11 (WiFi).

3.**Connection-oriented service**. The source and destination machines establish a connection before any data are transferred. Each frame sent over the connection is numbered, and the data link layer guarantees that each frame sent is indeed received. It guarantees that each frame is received exactly once and that all frames are received in the right order. Connection-oriented service thus provides the network layer processes with the equivalent of a reliable bit stream. In this transfers go through three distinct phases.
In **the first phase**, the connection is established by having both sides initialize variables and counters needed to keep track of which frames have been received and which ones have not. In the **second phase**, one or more frames are actually transmitted.
In the third and **final phase**, the connection is released, freeing up the variables, buffers, and other resources used to maintain the connection.

***Figure 3-3. Placement of the data link protocol.***




**Function2 : Framing:**
The data link layer have to detect and, if necessary, correct errors in the bit stream received from the physical layer.
The data link layer to break up the bit stream into discrete **frames,** compute a short token called a **checksum** for each frame, and include the checksum in the frame when it is transmitted. When a frame arrives at the **destination**, the checksum is **recomputed**. If the newly computed checksum is different from the one contained in the frame, the data link layer knows that an error has occurred and takes steps to deal with it.
Four methods of breaking up the bit stream into frames:

1. Byte count.
2. Flag bytes with byte stuffing.
3. Flag bits with bit stuffing.
4. Physical layer coding violations.

**1. Byte count** :

A **field in the header** is used to specify the **number of bytes** in the frame. When the data link layer at the destination sees the byte count, it knows how many bytes follow and hence where the end of the frame is. Example frames of sizes 5, 5, 8, and 8 bytes, respectively.

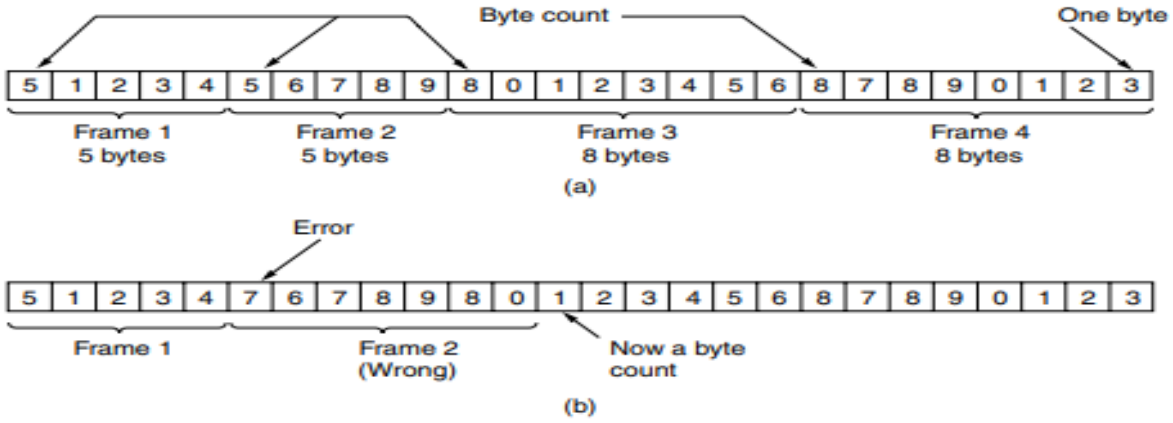


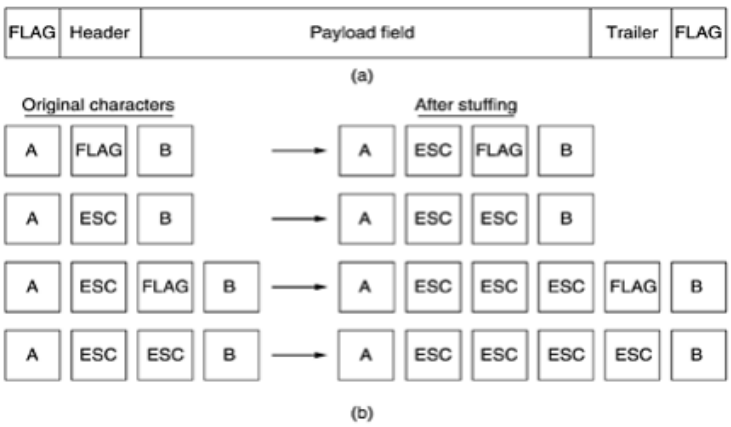

**Figure 3-3.** A byte stream. (a) Without errors. (b) With one error.

Drawback: The count can be garbled by a transmission error. For example, if the byte count of 5 in the second frame of Fig. 3-3(b) becomes a 7 due to a single bit flip, the destination will get out of synchronization. It will then be unable to locate the correct start of the next frame. Even if the checksum is incorrect so the destination knows that the frame is bad, it still has no way of telling where the next frame starts. Sending a frame back to the source asking for a retransmission does not help either, since the destination does not know how many bytes to skip over to get to the start of the retransmission.

**2. Flag bytes with byte stuffing:**

This method gets around the problem of resynchronization after an error by having each frame start and end with special bytes. In the past, the starting and ending bytes were different, but in recent years most protocols have used the same byte, called a flag byte, as both the starting and ending delimiter, as shown in Fig. 3-5(a) as FLAG.

***Figure 3-5. (a) A frame delimited by flag bytes. (b) Four examples of byte sequences before and after byte stuffing.***



If the receiver ever loses synchronization, it can just search for the flag byte to find the end of the current frame. Two consecutive flag bytes indicate the end of one frame and start of the next one.
Problem: when binary data, are being transmitted. The flag byte's bit pattern occurs in the data. The sender's data link layer insert a special escape byte (ESC) just before each "accidental" flag byte in the data. The data link layer on the receiving end removes the escape byte before the data are given to the network layer. This technique is called **byte stuffing or character stuffing**. Thus, a framing flag byte can be distinguished from one in the data by the absence or presence of an escape byte before it.
Any single escape byte is part of an escape sequence, whereas a doubled one indicates that a single escape occurred naturally in the data.
Disadvantage: It is closely tied to the use of 8-bit characters.
New technique had to be developed to allow arbitrary sized characters.

**3. Flag bits with bit stuffing:**
This technique allows data frames to contain an arbitrary number of bits and allows character codes with an arbitrary number of bits per character.
Each frame begins and ends with a special bit pattern, 01111110 (in fact, a flag byte).
Whenever the sender's data link layer encounters five consecutive 1s in the data, it automatically stuffs a 0 bit into the outgoing bit stream. This bit stuffing is analogous to byte stuffing, in which an escape byte is stuffed into the outgoing character stream before a flag byte in the data.
When the receiver sees five consecutive incoming 1 bits, followed by a 0 bit, it automatically deletes the 0 bit. Just as byte stuffing is completely transparent to the network layer in both computers, so is bit stuffing. If the user data contain the flag pattern, 01111110, this flag is transmitted as 011111010 but stored in the receiver's memory as 01111110.

***Figure 3-6. Bit stuffing. (a) The original data. (b) The data as they appear on the line. (c) The data as they are stored in the receiver's memory after destuffing.***


The boundary between two frames can be unambiguously recognized by the flag pattern. Thus, if the receiver loses track of where it is, all it has to do is scan the input for flag sequences, since they can only occur at frame boundaries and never within the data.

**4. Physical layer coding violations.**
It is only applicable to networks in which the encoding on the physical medium contains some redundancy. For example, some LANs encode 1 bit of data by using 2 physical bits. Normally, a 1 bit is a high-low pair and a 0 bit is a low-high pair. The scheme means that every data bit has a transition in the middle, making it easy for the receiver to locate the bit boundaries. The

combinations high-high and low-low are not used for data but are used for delimiting frames in some protocols.

Many data link protocols use a combination of a character count with one of the other methods for extra safety. When a frame arrives, the count field is used to locate the end of the frame. Only if the appropriate delimiter is present at that position and the checksum is correct is the frame accepted as valid. Otherwise, the input stream is scanned for the next delimiter.

**Function3 : Error Control:**

Making sure all frames are eventually delivered to the network layer at the destination and in the proper order.

To ensure reliable delivery is to provide the sender with some feedback : Typically, the protocol calls for the receiver to send back special control frames bearing positive or negative acknowledgements about the incoming frames. If the sender receives a **positive acknowledgement** about a frame, it knows the frame has arrived safely. A **negative acknowledgement** means that something has gone wrong, and the frame must be transmitted again.

The hardware troubles may cause a frame to vanish completely. Then introducing **timers** into the data link layer. When the sender transmits a frame, it generally also starts a timer. The timer is set to expire after an interval long enough for the frame to reach the destination, be processed there, and have the acknowledgement propagate back to the sender. Normally, the frame will be correctly received and the acknowledgement will get back before the timer runs out, in which case the timer will be canceled.

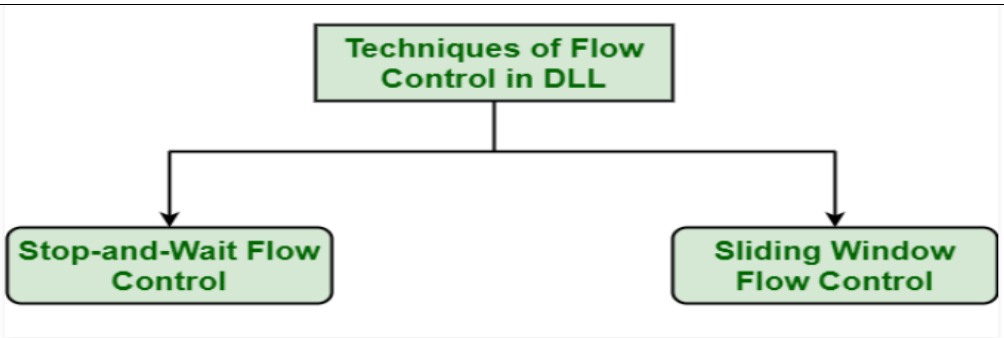
Either the frame or the acknowledgement is lost, the timer will go off, alerting the sender. The solution is to just transmit the frame again. However, when frames may be transmitted multiple times there is a danger that the receiver will **accept the same frame two or more times** and pass it to the network layer more than once. To prevent this from happening, it is generally necessary to assign **sequence numbers** to outgoing frames, so that the receiver can distinguish retransmissions from originals.

**Function4: Flow Control:**

Design issue associated when sender is fast and receiver is slow. The sender keeps pumping the frames out at a high rate until the receiver is completely swamped. At a certain point the receiver will simply be unable to handle the frames as they arrive and will start to lose some. Two approaches are commonly used.

1. **feedback-based flow control**, the receiver sends back information to the sender giving it permission to send more data or at least telling the sender how the receiver is doing.
2. **rate-based flow control,** the protocol has a built-in mechanism that limits the rate at which senders may transmit data, without using feedback from the receiver.

## FLOW CONTROL

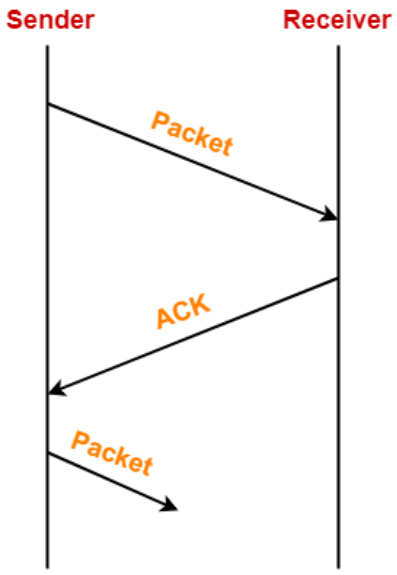



Flow control is a set of procedures which tells the sender how much data can be transmitted before it must wait for an acknowledgment from the receiver.
Flow of data must not overload the receiver
Receiver must also be able to inform the sender before it reaches the limit.
Flow control techniques are of two types :

1. stop and wait / request-reply FC
2. sliding window FC

**1. Stop-and-Wait Flow Control:**

This method is the easiest and simplest form of flow control. A message or data is broken down into various **multiple frames**, and then receiver indicates its readiness to receive frame of data. When acknowledgment is received, then only sender will send or transfer the next frame. This process is continued until sender transmits **EOT (End of Transmission)** frame. In this method, only one of frames can be in transmission at a time. It leads to inefficiency i.e. less productivity if propagation delay is very much longer than the transmission delay.
Sender sends one data packet and then waits for its acknowledgement.
Sender sends the next packet only after it receives the acknowledgement for the previous packet.




Stop and Wait Protocol

**Advantages –**
This method is very easiest and simple and each of the frames is checked and acknowledged well.
It can also be used for noisy channels.
This method is also very accurate.
**Disadvantages –**
This method is fairly slow.
In this, only one packet or frame can be sent at a time.
It is very inefficient and makes the transmission process very slow.

**2. Sliding Window Flow Control:**
This method is required where **reliable in-order delivery** of packets or frames is very much needed like in data link layer. It is point to point protocol that assumes that none of the other entity tries to communicate until current data or frame transfer gets completed. The sender transmits or sends various frames or packets before receiving any acknowledgment. Both the sender and receiver agree upon total number of data frames after which acknowledgment is needed to be transmitted. Data Link Layer requires and uses this method that simply allows sender to have more than one unacknowledged packet "in-flight" at a time. This increases and improves network throughput.

For any sliding window protocol to work without any problem, the following condition must be satisfied-

Available Sequence Numbers <= Sender Window Size + Receiver Window Size



**Advantages –**
It performs much better than stop-and-wait flow control.
This method increases efficiency.
Multiples frames can be sent one after another.
**Disadvantages –**
The main issue is complexity at the sender and receiver due to the transferring of multiple frames.
The receiver might receive data frames or packets out the sequence.

Maximum number of frames that sender can send without acknowledgement = Sender window size

Sliding window protocol again is of three types:

1. stop and wait ARQ
2. go back n ARQ
3. Selective repeat ARQ

**Stop and Wait ARQ**

Stop and Wait ARQ
= Stop and Wait Protocol + Time Out Timer + Sequence Numbers for Data Packets and Acknowledgement

Stop and wait ARQ is a one bit sliding window protocol where-
Sender window size = 1
Receiver window size = 1

Thus, in stop and wait ARQ,
Minimum number of sequence numbers required = Sender Window Size + Receiver Window Size
= 1+1 =2.

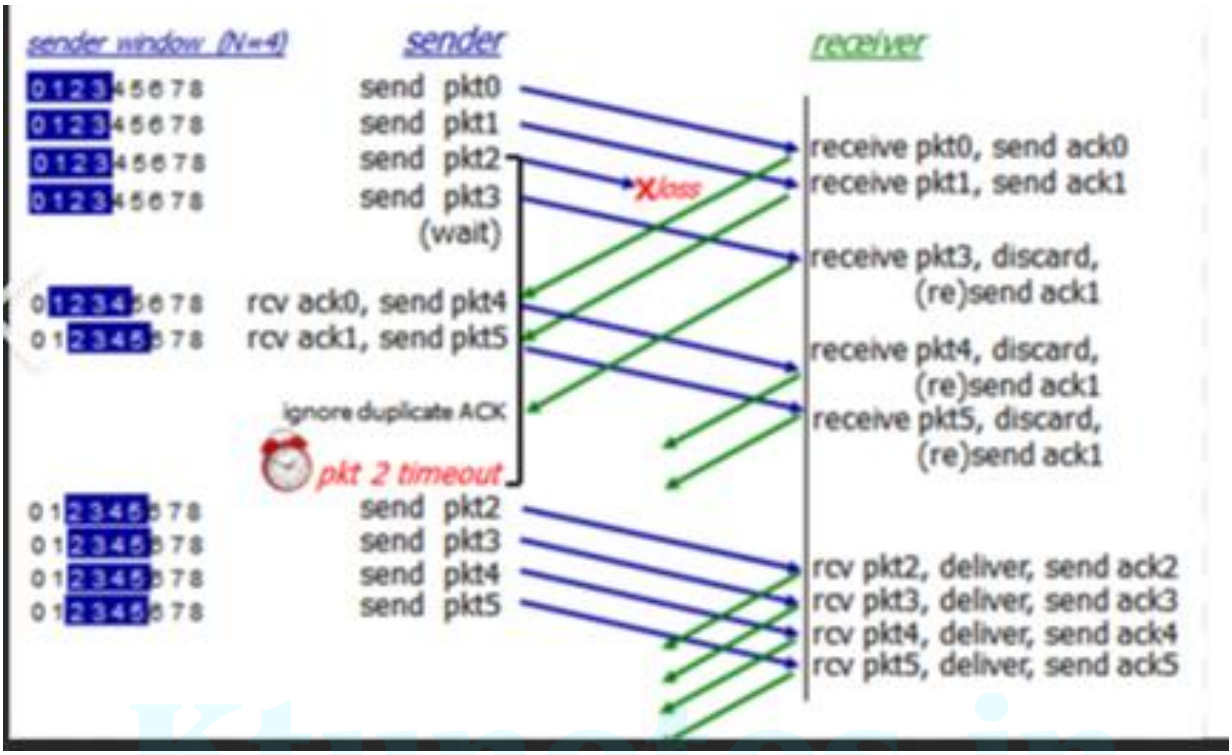
Minimum number of sequence numbers required in Stop and Wait ARQ = 2.
Two sequence number used are 0 and 1

| Sr. No. | Key | Stop and Wait protocol | Sliding Window protocol |
|---|---|---|---|
| 1 | Mechanism | In Stop and Wait protocol, sender sends single frame and waits for acknowledgment from the receiver. | In Sliding window protocol, sender sends multiple frames at a time and retransmits the damamged frames. |
| 2 | Efficiency | Stop and Wait protocol is less efficient. | Sliding Window protocol is more efficient than Stop and Wait protocol. |
| 3 | Window Size | Sender's window size in Stop and Wait protocol is 1. | Sender's window size in Sliding Window protocol varies from 1 to n. |
| 4 | Sorting | Sorting of frames is not needed. | Sorting of frames helps increasing the efficiency of the protocol. |
| 5 | Efficiency | Stop and Wait protocol efficiency is formulated as 1/(1+2a) where a is ratio of propagation delay vs transmission delay. | Sliding Window protocol efficiency is formulated as N/(1+2a) where N is no. of window frames and a is ratio of propagation delay vs transmission delay. |
| 6 | Duplex | Stop and Wait protocol is half duplex in nature. | Sliding Window protocol is full duplex in nature. |

**Go back N ARQ / continuous arq**

In Go back N, sender window size is N and receiver window size is always 1.
Go back N uses cumulative acknowledgements – sender keep on sending without waiting for the acknowledgement. Sender keeps a copy of the message till ack received. Receiver sends ack if message delivered correct or else negative acknowledgement nack.
Receiver maintains an acknowledgement timer.
Each time the receiver receives a new frame, it starts a new acknowledgement timer.
After the timer expires, receiver sends the cumulative acknowledgement for all the frames that are unacknowledged at that moment.
If receiver receives a frame that is corrupted, then it silently discards that frame.
The correct frame is retransmitted by the sender after the time out timer expires.
Go back N does not accept out of order frames and silently discards them.

Go back N leads to retransmission of entire window if for any frame, no ACK is received by the sender.




Over all go back n

**Selective Repeat Protocol-**

In SR protocol, sender window size is always same as receiver window size. SR protocol uses independent acknowledgements only.

In SR protocol,

- If receiver receives a frame that is corrupted, then it does not silently discard that frame.
- Receiver handles the situation efficiently by sending a negative acknowledgement (NACK).
- Negative acknowledgement allows early retransmission of the corrupted frame.
- It also avoids waiting for the time out timer to expire at the sender side to retransmit the frame.

Consider receiver receives a frame whose sequence number is not what the receiver expects. Then, it does not discard that frame rather accepts it and keeps it in its window.

In SR protocol,

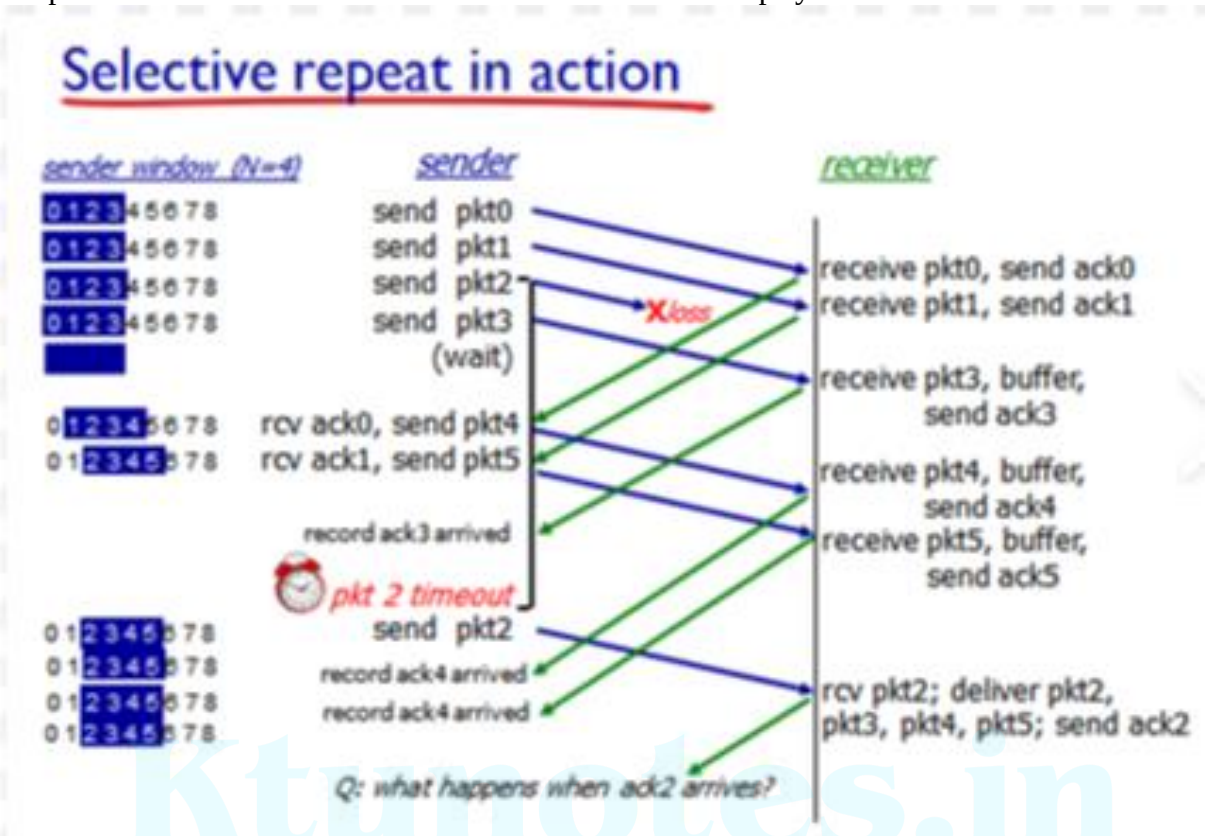
- Receiver window is implemented as a linked list.
- When receiver receives a new frame, it places the new frame at the end of the linked list.
- When the received frames are out of order, receiver performs the sorting.
- Sorting sorts the frames in the correct order.

Only the missing frame has to be sent by the sender.

- For sending the missing frame, sender performs searching and finds the missing frame.
- Then, sender selectively repeats that frame.

- Thus, only the selected frame is repeated and not the entire window.

SR protocol leads to retransmission of lost frames after expiry of time out timer.






**Piggy Backing**

To improve the efficiency of bidirectional protocols. Full duplex transmission can be achieved using two separate communicating channels that is wastage of bandwidth.

When a data frame arrives a receiver, it will not send the acknowledgment immediately instead it waits until its network layer passes data to data link layer. The acknowledgement is attached to data frame. The technique in which out going acknowledgement is delayed is called as **piggybacking.**

**DATA LINK PROTOCOLS – HDLC,PPP**

**PPP –** Point to Point Protocol

**HDLC -** High level Data Link Control

Is a bit oriented protocol.

Three types of station are being defined in HDLC:

1) **Primary station** – responsible for connecting and disconnecting data link. The frames sent by them are called **commands.**
2) **Secondary station –** operates under the control of primary station. The frames sent by them are called **response.**
3) **Combined station** – can act as a primary and secondary station. The frames can be command or response.

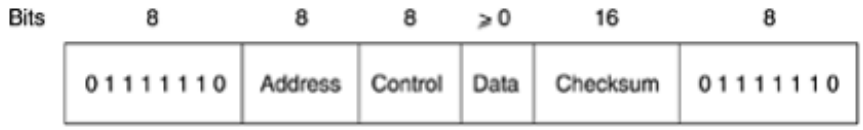
**Frame format:**

HDLC There are three kinds of frames:

1) **Information frame / I frame –** used to transport user data and control information related to user data.
2) **Supervisory frame / S frame -** used to transport control information.
3) **Unnumbered frame / U frame -** is reserved for system management and information carried for managing the link.

Each frame acts as an envelope for the transmission of different type of message. Each frame may carry up to six fields

***Figure 3-24. Frame format for bit-oriented protocols.***




frame structure.

**01111110 – flag field – 8 bit** sequence that identifies beginning an end of frames. This helps as synchronization pattern for the receiver. The frame is **delimited** with another flag sequence (01111110).

The **Address** field is used to identify one of the terminals. This can be 1 byte or more. 1 byte can identify 128 stations.

The **Control** field is 1 or 2 bytes used for flow control or error control. This bits depends on the frame type.



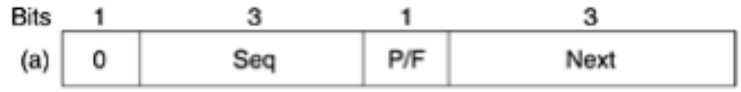
The **Data** field may contain any information. It may be arbitrarily long.

The **Checksum** field – error detection field

The minimum frame contains three fields and totals 32 bits, excluding the flags on either end.

The contents of the Control field vary for these three kinds:

**Control field of I frame :**




I frame is used to carry data from network layer. Iframe include flow control and error control information.

- 1st bit is 0 – denotes that frame is of type I
- Next 3 bits -The **Seq field** is the frame sequence number. (0 -7)
- **The P/F bit** stands for **Poll/Final**.
  **P/F = 1 poll :** frame sent by primary to secondary.
  **P/F = 0 final :** frame sent by secondary to primary.
- The Next field is a piggybacked **acknowledgement number**. Instead of piggybacking the number of the last frame received correctly, they use the number of the first frame not yet received (i.e., the next frame expected). The choice of using the last frame received or the next frame expected is arbitrary.

**Control field of S frame :**


Supervisory frame are used for flow and error control when piggybacking is inappropriate. Eg: when station has no data to send of its own or have to send some thing othe than acknowledgement.
S frame do not have information field.

- First two bits **1 0** means its S frame.
- 3 and 4 bit – **Type** of S frame :

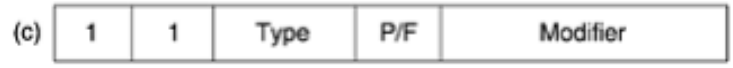
The various kinds of Supervisory frames are distinguished by the Type field.

- **Type 00 / Receive Ready** is an acknowledgement frame used to indicate the next frame expected.
- **Type 01/ Reject S frame** is a negative acknowledgement frame. It is used to indicate that a transmission error has been detected. The **Next field** indicates the first frame in sequence not received correctly (i.e., the frame to be retransmitted). The sender is required to retransmit all outstanding frames starting at Next.
- **Type 10/ Receive not ready** - mentions receiver is busy and cannot receive any frame.
- **Type 11 / Selective reject**. It calls for retransmission of only the frame specified.

- **The P/F bit** stands for **Poll/Final**.
  **P/F = 1 poll :** frame sent by primary to secondary.
  **P/F = 0 final :** frame sent by secondary to primary.
- The Next field is a piggybacked **acknowledgement number**

**Control field of U frame :**

The third class of frame is the **Un-numbered frame**. It is sometimes used for control purposes but can also carry data when unreliable connectionless service is called for.




2 bits befor P/F and 3 bits after together they contribute upto 32 different types of u frames.

HDLC provides two communication **transfer modes**:

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

**Normal Response Mode (NRM)**

Is having unbalanced station configuration. One primary station and multiple secondary stations. NRM is used for both point to point and multi point links.

**Asynchronous Balanced Mode (ABM)**

Configuration is balanced.
Link is point to point.
Each station can function as primary and secondary.
Frame transmitted as full duplex mode.

## DLL IN INTERNET

The Internet consists of individual machines (hosts and routers) and the communication infrastructure.
Data link protocol used in point to point line in internet.
Point to point communication is used in two situations:

1. Multiple Number of hosts along with a router.
2. Single host along with a modem and telephone dialup connection.

### PPP—The Point-to-Point Protocol

The Internet needs a point-to-point protocol for a variety of purposes, including router-to-router traffic and home user-to-ISP traffic. PPP handles error detection, supports multiple protocols, allows IP addresses to be negotiated at connection time, permits authentication, and has many other features. PPP is a multiprotocol framing mechanism suitable for use over modems, HDLC bit-serial lines, SONET, and other physical layers. It supports error detection, option negotiation, header compression, and, optionally, reliable transmission using an HDLC-type frame format.

**PPP provides three features/services:**

1. A **framing** method – that shows the end of one frame and the start of the next one without confusion. The frame format also handles **error detection**.

2. A link control protocol for bringing lines up, testing them, negotiating options, and bringing them down again gracefully when they are no longer needed. This protocol is called **LCP (Link Control Protocol)**. It supports synchronous and asynchronous circuits and byte-oriented and bit-oriented encodings.

3. A way to negotiate network-layer options in a way that is independent of the network layer protocol to be used. The method chosen is to have a different **NCP (Network Control Protocol**) for each network layer supported.

**Missing services of PPP:**

1. Flow control
2. Very simple error control mechanism
3. Does not provide a addressing mechanism to handle frame in multipoint configuration.

**Working scenario:**

A home user calling up an Internet service provider to make a home PC a temporary Internet host.

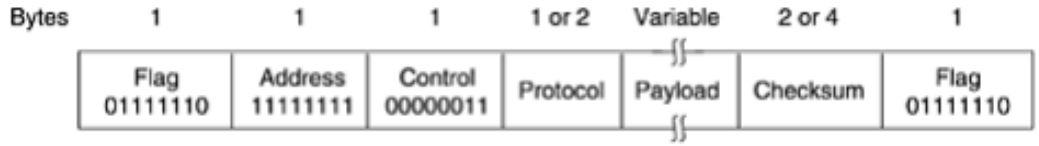
1. The PC first calls the provider's router via a modem.
2. The router's modem has answered the phone and established a physical connection
3. The PC sends the router a series of LCP packets in the payload field of one or more PPP frames.
4. These packets and their responses select the PPP parameters to be used.
5. Once the parameters have been agreed upon, a series of NCP packets are sent to configure the network layer.
6. Typically, the PC wants to run a TCP/IP protocol stack, so it needs an IP address. There are not enough IP addresses to go around, so normally each Internet provider gets a block of them and then dynamically assigns one to each newly attached PC for the duration of its login session.

7. The NCP for IP assigns the IP address.
8. At this point, the PC is now an Internet host and can send and receive IP packets, just as hardwired hosts can.
9. When the user is finished, NCP tears down the network layer connection and frees up the IP address.
10. Then LCP shuts down the data link layer connection.
11. Finally, the computer tells the modem to hang up the phone, releasing the physical layer connection.

**PPP frame format**

PPP is character oriented. PPP uses byte stuffing on dial-up modem lines, so all frames are an integral number of bytes.

***Figure 3-27. The PPP full frame format for unnumbered mode operation.***



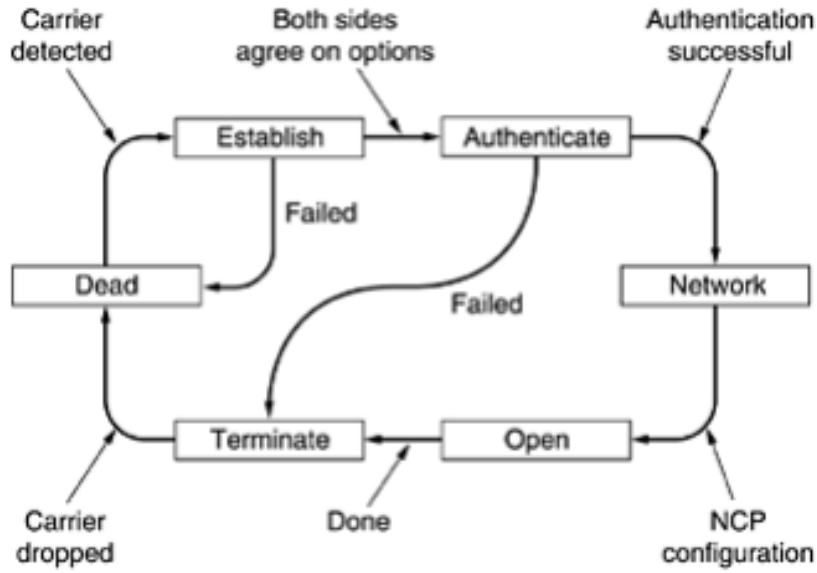

1. All PPP frames begin with the standard **HDLC flag byte (01111110),** which is byte stuffed if it occurs within the payload field.
2. **Address field**, which is always set to the binary value **11111111** to indicate that **all stations** are to accept the frame. Using this value **avoids** the issue of having **to assign data link addresses.**
3. **Control field**, the default value of which is **00000011**. This value indicates an **unnumbered frame**. PPP does not provide reliable transmission using sequence numbers and acknowledgements as the default. In noisy environments, such as wireless networks, reliable transmission using numbered mode can be used.

**Address and Control fields are always constant** in the default configuration, LCP provides the necessary mechanism for the two parties to negotiate an option to just omit them altogether and save 2 bytes per frame.

4. **Protocol field** - Its job is to tell what kind of packet is in the Payload field. Protocols **starting with a 0 bit** are **network layer protocols.** Those **starting with a 1** bit are used **to negotiate other protocols**. These include LCP and a different NCP for each network layer protocol supported. The default size of the Protocol field is 2 bytes, but it can be negotiated down to 1 byte using LCP.
5. The **Payload field** is variable length, up to some negotiated maximum. If the length is not negotiated using LCP during line setup, a default length of 1500 bytes is used. **Padding** may follow the payload if need be.
6. **Checksum field,** which is normally 2 bytes, but a 4-byte checksum can be negotiated.

**Bringing line up and down:**

*Figure 3-28. A simplified phase diagram for bringing a line up and down.*




This sequence applies both to modem connections and to router-router connections.

1. The protocol starts with the line in the DEAD state, - no physical layer carrier is present and no physical layer connection exists.
2. After physical connection is established, the line moves to ESTABLISH. At that point LCP option negotiation begins, which, if successful,
3. leads to AUTHENTICATE. Now the two parties can check on each other's identities if desired.
4. When the NETWORK phase is entered, the appropriate NCP protocol is invoked to configure the network layer.
5. If the configuration is successful, OPEN is reached and data transport can take place.
6. When data transport is finished, the line moves into the TERMINATE phase, and from there, back to DEAD when the carrier is dropped.

LCP negotiates data link protocol options during the ESTABLISH phase.

## MAC SUB LAYER – IEEE 802 FOR LANS & MANS

Networks can be divided into two categories:

1. pointto-point connections
2. broadcast channel/s multiaccess channels or random access channels.

In any broadcast network, the key issue is how to determine who gets to use the channel when there is competition for it. The protocols used to determine who goes next on a multiaccess channel belong to a sublayer of the data link layer called the MAC (Medium Access Control) sublayer. The MAC sublayer is especially important in LANs, many of which use a multiaccess channel as the basis for communication. WANs, in contrast, use point-to-point links, except for satellite networks.

The IEEE has subdivided the data link layer into two sublayers:

1. logical link control (LLC) and
2. media access control (MAC).

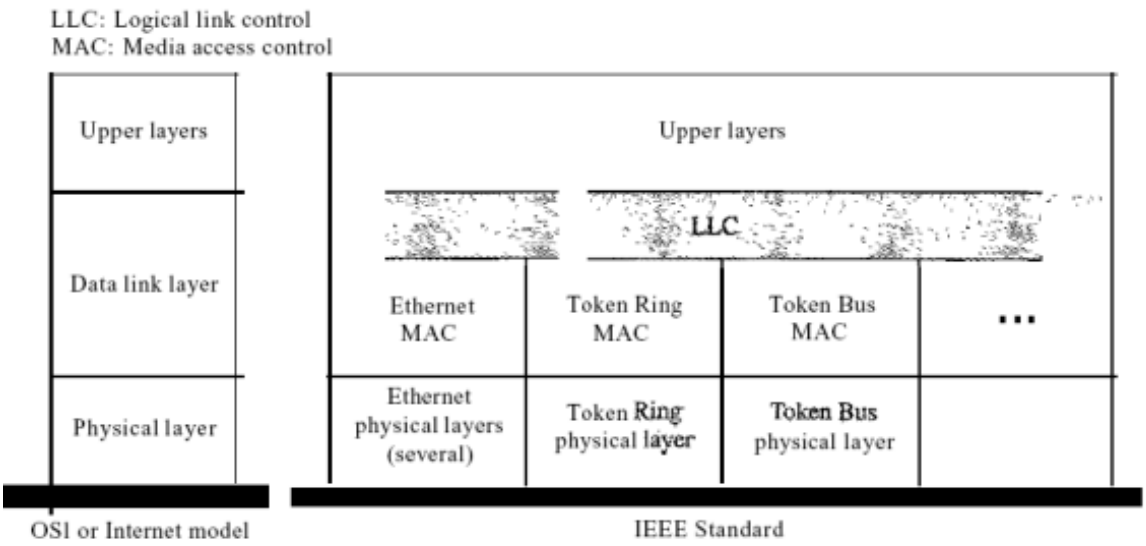



Figure 1 *IEEE standardfor LANs*

**Logical Link Control (LLC)**

Data link control handles framing, flow control, and error control.

In IEEE Project 802, **flow control, error control,and part of the framing** duties are collected into one sublayer called the logical link control.

Framing is handled in both the LLC sublayer and the MAC sublayer. The LLC provides one single data link control protocol for all IEEE LANs. In this way, the LLC is different from the media access control sublayer, which provides different protocols for different LANs.

A single LLC protocol can provide interconnectivity between different LANs because it makes the MAC sublayer transparent.

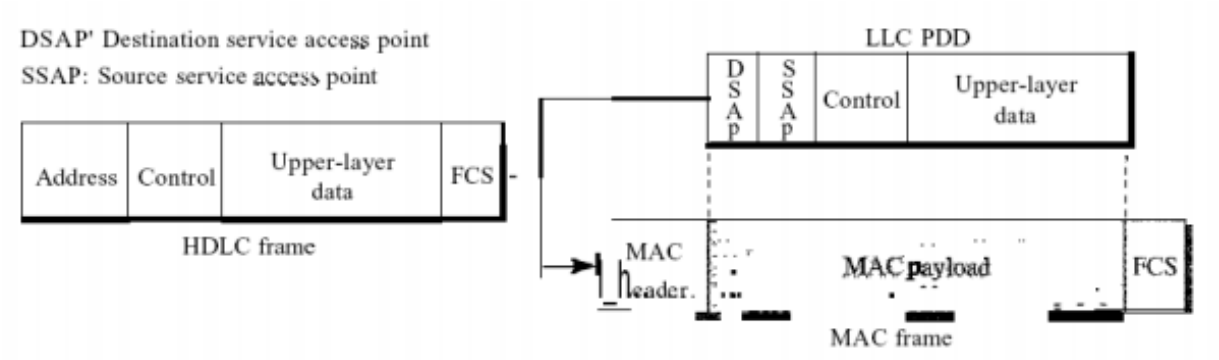



Figure 13.2 *HDLCframe compared with LLC and MAC frames*

Framing LLC defines **a protocol data unit (PDU**).

- The header contains a **control field** for flow and error control.

- The two other header fields define the upper-layer protocol at the source and destination that uses LLC. These fields are called the **destination service access point (DSAP)** and the **source service access point (SSAP).**
- The other fields defined in a typical data link control protocol are moved to the MAC sublayer.
- In other words, a frame defined in HDLC is divided into a PDU at the LLC sublayer and a frame at the MAC sublayer, as shown in Fig 2.

**Media Access Control (MAC)**

IEEE Project 802 has created a sublayer called media access control that defines the specific access method for each LAN.

For example, it defines CSMA/CD as the media access method for Ethernet LANs and the token passing method for Token Ring and Token Bus LANs. A part of the framing function is also handled by the MAC layer. The MAC sublayer contains a number of distinct modules; each defines the access method and the framing format specific to the corresponding LAN protocol.

**Functions of MAC:**

1. Perform control of access to media
2. Perform unique addressing to station directly connected to LANS
3. Error detection

**Channel allocation problem:**

1. Static channel allocation - eg: FDM, TDM
2. Dynamic channel allocation – eg: no fixed time and frequency

**Multiple access protocols:**

| | |
|---|---|
| **1.** ALOHA<br>**a.** Pure aloha<br>**b.** Slotted aloha<br>**2.** Carrier sense multiple access (CSMA)<br>**a.** Persistent and non-persistent CSMA<br>**b.** CSMA with collision detection<br>**3.** Wavelength division multiplexing | **1.** Random access<br>**2.** Controlled access<br>**3.** Channelization |

**MAC – MEDIA ACCESS CONTROL**

**ALOHA system,** used groundbased radio broadcasting, the basic idea is applicable to any system in which uncoordinated users are competing for the use of a single shared channel. There are two versions of ALOHA: **pure and slotted**. They differ with respect to whether time is continuous, as in the pure version, or divided into discrete slots into which all frames must fit.

**Pure ALOHA**

The basic idea of an ALOHA system is simple: users transmit whenever they have data to be sent. The colliding frames will be damaged. Senders need some way to find out the collision.

After each station has sent its frame to the central computer, this computer rebroadcasts the frame to all of the stations. A sending station thus listen for the broadcast from the hub to see if its frame has gotten through.
In other systems, such as wired LANs, the sender might be able to listen for collisions while transmitting. If the frame was destroyed, the sender just waits a random amount of time and sends it again. The waiting time must be random or the same frames will collide over and over, in lockstep. Systems in which multiple users share a common channel in a way that can lead to conflicts are known as **contention systems.**

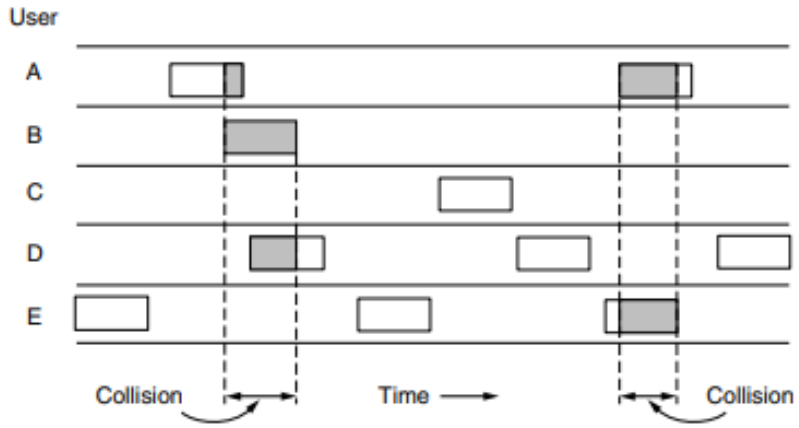



**Figure 4-1.** In pure ALOHA, frames are transmitted at completely arbitrary times.

All the frames are with the same length because the throughput of ALOHA systems is maximized by having a uniform frame size rather than by allowing variable-length frames. Whenever two frames try to occupy the channel at the same time, there will be a collision (as seen in Fig. 4-1) and both will be garbled. If the first bit of a new frame overlaps with just the last bit of a frame that has almost finished, both frames will be totally destroyed (i.e., have incorrect checksums) and both will have to be retransmitted later. The checksum does not (and should not) distinguish between a total loss and a near miss.

If N > 1, the user community is generating frames at a higher rate than the channel can handle, and nearly every frame will suffer a collision. For reasonable throughput, we would expect 0 <N< 1.

A frame will not suffer a collision if no other frames are sent within one frame time of its start, as shown in Fig. 4-2.

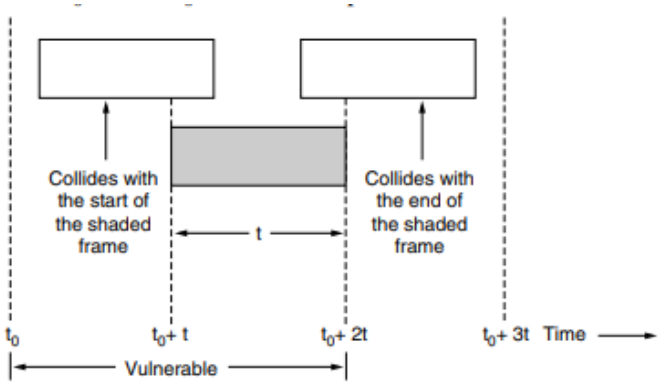



**Figure 4-2.** Vulnerable period for the shaded frame.

Let t be the time required to send one frame. If any other user has generated a frame between time t 0 and t 0 + t, the end of that frame will collide with the beginning of the shaded one. In fact, the shaded frame's fate was already sealed even before the first bit was sent, but since in pure ALOHA a station does not listen to the channel before transmitting, it has no way of knowing that another frame was already underway. Similarly, any other frame started between t 0 + t and t 0 + 2t will bump into the end of the shaded frame.

The probability that k frames are generated during a given frame time, in which G frames are expected, is given by the Poisson distribution.

$$\Pr[k] = \frac{G^k e^{-G}}{k!} \tag{4-2}$$

so the probability of zero frames is just $e^{-G}$. In an interval two frame times long, the mean number of frames generated is $2G$. The probability of no frames being initiated during the entire vulnerable period is thus given by $P_0 = e^{-2G}$. Using $S = GP_0$, we get

$$S = Ge^{-2G}$$

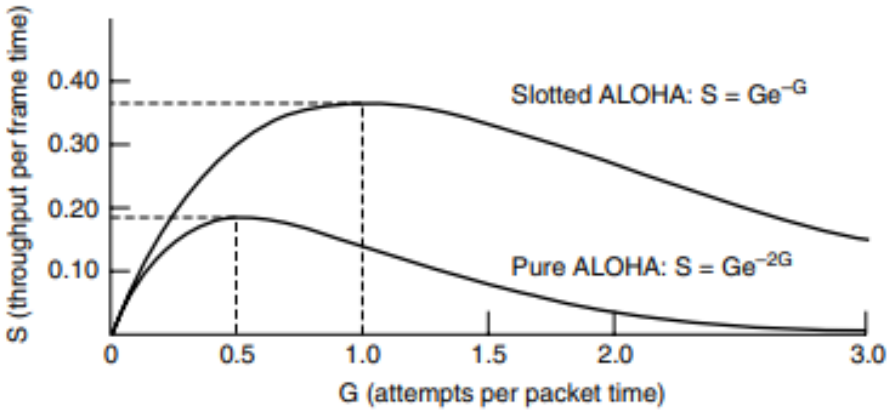



**Figure 4-3.** Throughput versus offered traffic for ALOHA systems.

**Slotted ALOHA**

A method for doubling the capacity of an ALOHA system. His proposal was to divide time into discrete intervals called **slots**, each interval corresponding to one frame. This approach requires the users to agree on slot boundaries. One way to achieve synchronization would be to have one special station emit a pip at the start of each interval, like a clock the continuous time ALOHA is turned into a discrete time one. This halves the vulnerable period. To see this, look at Fig. 4-3 and imagine the collisions that are now possible. The probability of no other traffic during the same slot as our test frame is then e −G, which leads to

$$S = Ge^{-G}$$

The slotted ALOHA peaks at G = 1, with a throughput of S = 1/e or about 0.368, twice that of pure ALOHA. The probability of a collision is then just 1 − e −G. The probability of a transmission requiring exactly k attempts (i.e., k − 1 collisions followed by one success) is

$$P_k = e^{-G}(1 - e^{-G})^{k-1}$$

The expected number of transmissions, $E$, per line typed at a terminal is then

$$E = \sum_{k=1}^{\infty} kP_k = \sum_{k=1}^{\infty} ke^{-G}(1 - e^{-G})^{k-1} = e^{G}$$

With slotted ALOHA, the best channel utilization that can be achieved is 1/e

**Carrier sense multiple access (CSMA)**

Protocols in which stations listen for a carrier (i.e., a transmission) and act accordingly are called **carrier sense protocols.**

**a.** Persistent and non-persistent CSMA
**b.** CSMA with collision detection

**Persistent and Nonpersistent CSMA:**

**1-persistent CSMA -** When a station has data to send, it first listens to the channel to see if anyone else is transmitting at that moment. If the channel is idle, the stations sends its data. Otherwise, if the channel is busy, the station just waits until it becomes idle. Then the station transmits a frame. If a collision occurs, the station waits a random amount of time and starts all over again. The protocol is called 1-persistent because the station transmits with a probability of 1 when it finds the channel idle. If two stations become ready in the middle of a third station's transmission, both will wait politely until the transmission ends, and then both will begin transmitting exactly simultaneously, resulting in a collision. This chance depends on the number of frames that fit on the channel, or **the bandwidth-delay** product of the channel.

**nonpersistent CSMA-** As before, a station senses the channel when it wants to send a frame, and if no one else is sending, the station begins doing so itself. However, if the channel is already in use, the station does not continually sense it for the purpose of seizing it immediately upon detecting the end of the previous transmission. Instead, it waits a random period of time and then repeats the algorithm. Consequently, this algorithm leads to better channel utilization but longer delays than 1-persistent CSMA.

**p-persistent CSMA.** It applies to slotted channels.. When a station becomes ready to send, it senses the channel. If it is idle, it transmits with a probability p. With a probability q = 1 − p, it

defers until the next slot. If that slot is also idle, it either transmits or defers again, with probabilities p and q. This process is repeated until either the frame has been transmitted or another station has begun transmitting. In the latter case, the unlucky station acts as if there had been a collision (i.e., it waits a random time and starts again). If the station initially senses that the channel is busy, it waits until the next slot and applies the above algorithm. IEEE 802.11 uses a refinement of p-persistent CSMA

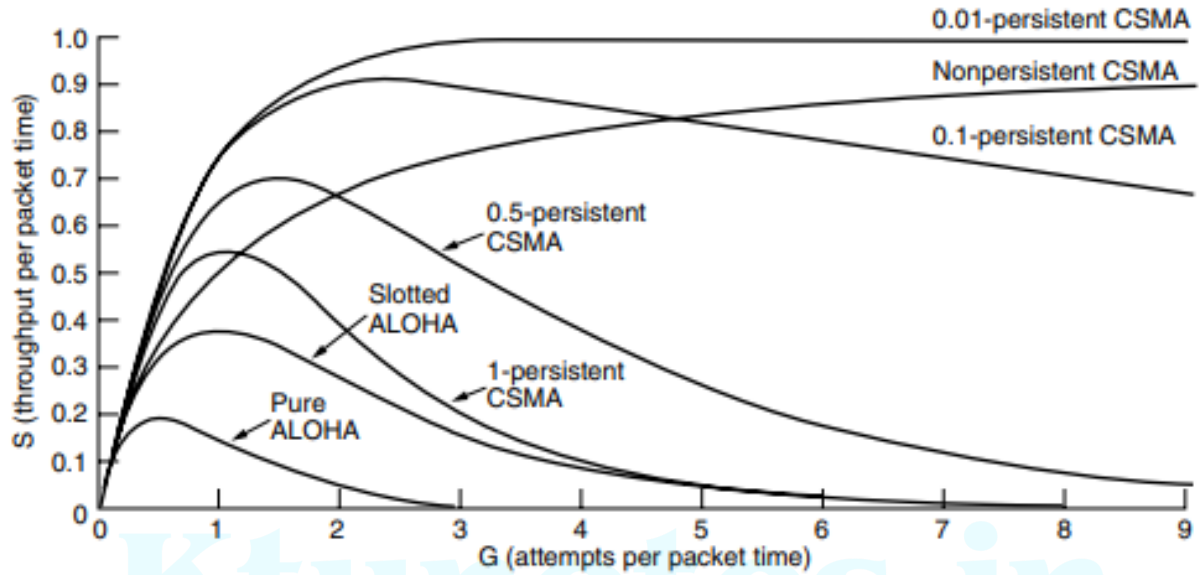



**Figure 4-4.** Comparison of the channel utilization versus load for various random access protocols.

**CSMA with Collision Detection:**

CSMA/CD (CSMA with Collision Detection), is the basis of the classic Ethernet LAN. Collision detection is an analog process. The station's hardware must listen to the channel while it is transmitting. If the signal it reads back is different from the signal it is putting out, it knows that a collision is occurring. The implications are that a received signal must not be tiny compared to the transmitted and that the modulation must be chosen to allow collisions to be detection.

At the point marked t 0, a station has finished transmitting its frame. Any other station having a frame to send may now attempt to do so. If two or more stations decide to transmit simultaneously, there will be a collision. If a station detects a collision.

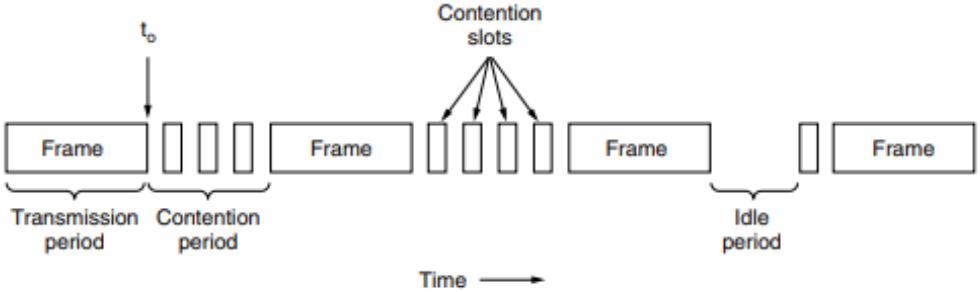



**Figure 4-5.** CSMA/CD can be in contention, transmission, or idle state.

Our model for CSMA/CD will consist of alternating contention and transmission periods, with idle periods occurring when all stations are quiet.

**IEEE 802 STANDARDS:**

IEEE has developed the layered architecture and other standards of LAN. In LAN all station share a common cable. So IEEE developed 3 mechanisms of media access control.

- Carrier Sense Multiple Access / Collision Detection (CSMA/CD)- IEEE 802.3- ETHERNET
- Token bus – IEEE 802.4
- Token Ring – IEEE 802.5

**IEEE 802.3**

The IEEE has standardized a number of local area networks and metropolitan area networks under the name of IEEE 802(Ethernet).
Two kinds of Ethernet exist:

- **Classic Ethernet**, which solves the multiple access problem using the techniques. ran at rates from 3 to 10 Mbps
- **Switched Ethernet,** in which devices called switches are used to connect different computers. runs at 100, 1000, and 10,000 Mbps, in forms called fast Ethernet, gigabit Ethernet, and 10 gigabit Ethernet.



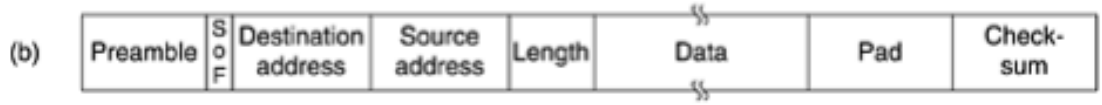
**ETHERNET cabling:**

The first number is the speed in Mbps. Then comes the word "Base" (or sometimes "BASE") to indicate baseband transmission.

- **10Base5** cabling, popularly called **thick Ethernet –**
  It has markings every 2.5 meters to show where the taps go. Connections to it are generally made using **vampire taps.** It operates at 10 Mbps, uses baseband signaling, and can support segments of up to 500 meters.
- **10Base2,** or **thin Ethernet** - covers 200m
  Connections to it are made using industry**standard BNC** connectors to form T junctions, rather than using vampire taps. BNC connectors are easier to use and more reliable. Thin Ethernet is much cheaper and easier to install, but it can run for only 185 meters per segment. It can handle up to 30 machines.
- **10BaseT – Twisted pair cable –** covers 100m and can connect up to 1024 machines.
- **10BaseF – Fiber Optics –** covers 2000m

10base5 and 10base2 uses bus topology, 10baseT and 10baseF uses star topology.

**The Ethernet MAC Sublayer Protocol**




- Each frame starts with a **Preamble of 8 bytes (**preamble 7 bytes + SFD 1 byte), each containing the bit pattern 10101010.

- The frame contains **two addresses**, one for the destination and one for the source. The standard allows 2-byte and 6-byte addresses. When a frame is sent to a group address, all the stations in the group receive it. Sending to a group of stations is called **multicast.** The address consisting of all 1 bits is reserved for **broadcast.** A frame containing all 1s in the destination field is accepted by all stations on the network.
  The difference between multicast and broadcast is: A multicast frame is sent to a selected group of stations on the Ethernet; a broadcast frame is sent to all stations on the Ethernet. Multicast is more selective, but involves group management. Broadcasting is coarser but does not require any group management.
  - **the destination address –** physical address of destination station.
  - **Source address –** physical address of the node which send the packets.
- **The Type field**, which tells the receiver what to do with the frame.
- Next come **the data**, up to 1500 bytes.
- If the data portion of a frame is less than 46 bytes, **the Pad** field is used to fill out the frame to the minimum size.
- The final Ethernet field is **the Checksum**. It is effectively a 32-bit hash code of the data for error detection. There is a restriction imposed on minimum(512 bits) and maximum(12144 bits) length of frame. The restriction of minimum length is to ensure correct option of CSMA/CD.

**Addressing**

Each station on an Ethernet network (such as a PC, workstation, or printer) has its own network interface card (NIC). The NIC fits inside the station and provides the station with a 6-byte physical address.

The Ethernet address is 6 bytes (48 bits), normally written in hexadecimal notation, with a colon between the bytes.

06:01 :02:01:2C:4B

6 bytes =12 hex digits =48 bits

**Type of address –** last significant bit of first byte decides the type of addressing. If its 0 – unicast else multicast. Broadcasting is a special type of multicasting in which all bits are 1's.

1) Unicast – source address is always unicast addressing
2) Multicast -
3) Broadcast

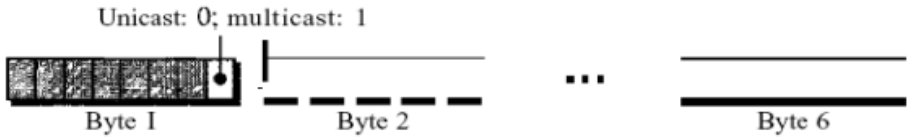



Figure 7 *Unicast and multicast addresses*

A unicast destination address defines only one recipient; the relationship between the sender and the receiver is one-to-one. A multicast destination address defines a group of addresses; the relationship between the sender and the receivers is one-to-many. The broadcast address is a special case of the multicast address; the recipients are all the stations on the LAN. A broadcast destination address is forty-eight.

To differentiate between an idle sender and o 0 bit ‘1’voltid uses for 1 bit and -1 for 0 bit.

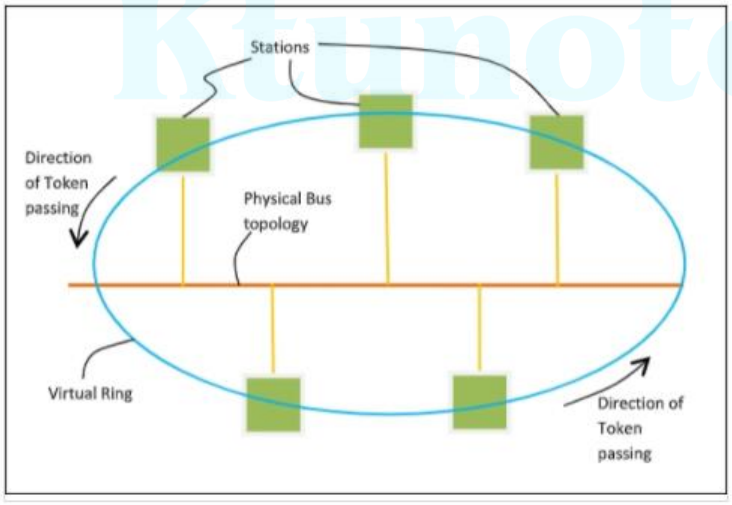
If receiver and sender have different speed, receiver need synchronization of where the boundaries are. So we need to know the start, end and middle of bits without an external clock. Two such approaches are :

1. Manchester encoding
2. Differential Manchester encoding.

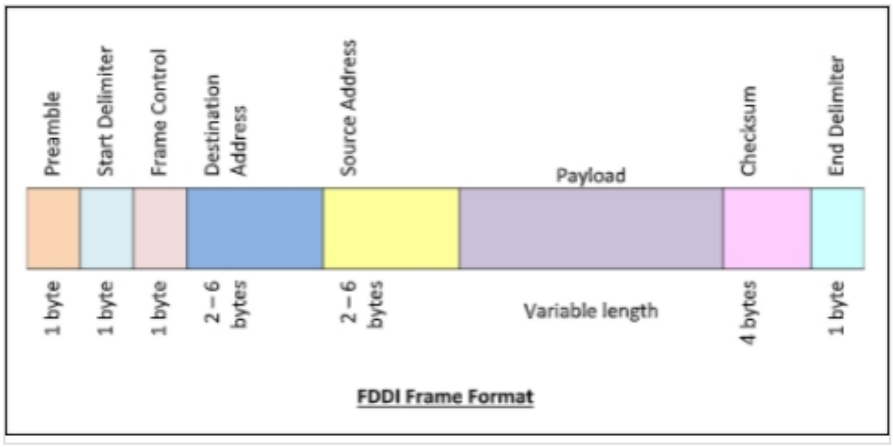
## IEEE 802.4

IEEE 802.4 describes a **token bus** **LAN standards**. In token passing methods, stations connected on a bus are arranged in a logical ring. In this method only the station having token(token holder)is being permitted to transmit frames.
Steps followed:

1. A virtual ring is created with the nodes/stations.
2. Station bearing the highest seq number and holding the token passes the frame.
3. the token is passed from one node to the next in a sequence along this virtual ring. Each node knows the address of its preceding station and its succeeding station. A station can only transmit data when it has the token. The working principle of the token bus is similar to Token Ring.




**Frame format:**




The fields of a token bus frame are −

- **Preamble:** 1 byte for synchronization.
- **Start Delimiter:** 1 byte that marks the beginning of the frame.
- **Frame Control:** 1 byte that specifies whether this is a data frame or control frame.
- **Destination Address:** 2-6 bytes that specifies address of destination station.
- **Source Address:** 2-6 bytes that specifies address of source station.
- **Payload:** A variable length field that carries the data from the network layer.
- **Checksum:** 4 bytes frame check sequence for error detection.
- **End Delimiter:** 1 byte that marks the end of the frame.



two types of transmission system are use:

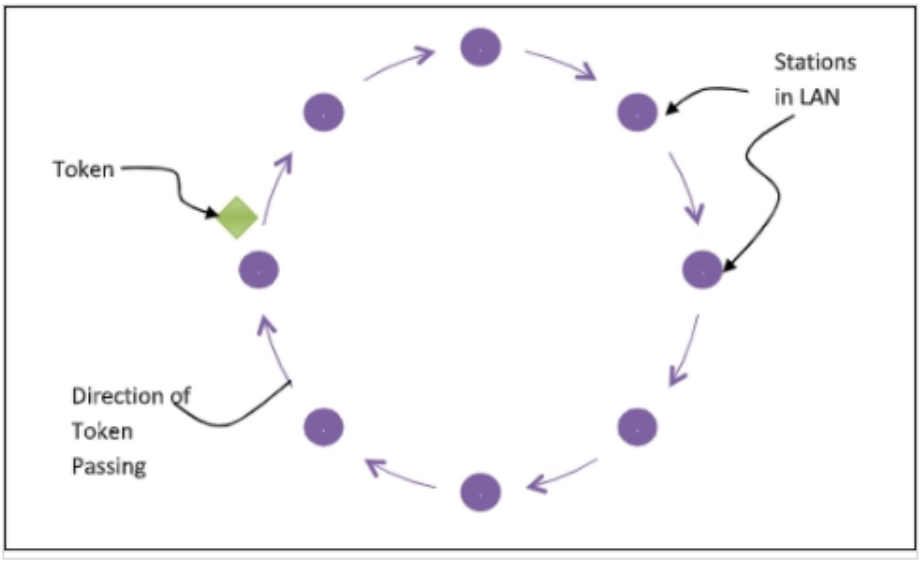
1. Career band
2. Broad band

**IEEE 802.5**

Token ring (IEEE 802.5) is a communication protocol in a local area network (LAN) where all stations are connected in a ring topology and pass one or more tokens for channel acquisition. A token is a special frame of 3 bytes that circulates along the ring of stations. A station can send data frames only if it holds a token. The tokens are released on successful receipt of the data frame.

**Token Passing Mechanism in Token Ring**

If a station has a frame to transmit when it receives a token, it sends the frame and then passes the token to the next station; otherwise it simply passes the token to the next station. Passing the token means receiving the token from the preceding station and transmitting to the successor station. The data flow is unidirectional in the direction of the token passing. In order that tokens are not circulated infinitely, they are removed from the network once their purpose is completed. This is shown in the following diagram −




**Frame format:**

**Data Frame**

| SFD | AC | FC | DA | SA | Data | CRC | ED | FS |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 6 | 6 | >= 0 | 1 | 1 | 1 |

- **Start frame delimiter (SFD) –** Alerts each station for the arrival of token(or data frame) or start of the frame. It is used to synchronize clocks.

- **Access control (AC)-**

**Priority bits** and **reservation bits** help in implementing priority. Priority bits = reservation bits = 3.
**Token bit** is used to indicate presence of token frame. If token bit = 1 –> token frame and if token bit = 0 –> not a token frame.
**Monitor bit** helps in solving orphan packet problem. It is covered by CRC as monitor are powerful machines which can recalculate CRC when modifying monitor bit. If monitor bit = 1 –> stamped by monitor and if monitor bit = 0 –> not yet stamped by monitor.

- **Frame control (FC) –** First 2 bits indicates whether the frame contains data or control information. In control frames, this byte specifies the type of control information.
- **Destination address (DA) and Source address (SA) –** consist of two 6-byte fields which is used to indicate MAC address of source and destination.

- **Data –** Data length can vary from 0 to maximum token holding time (THT) according to token reservation strategy adopted. Token ring imposes no lower bound on size of data i.e. an advantage over Ethernet.

- **Cyclic redundancy check (CRC) –** 32 bit CRC which is used to check for errors in the frame, i.e., whether the frame is corrupted or not. If the frame is corrupted, then its discarded.

- **End delimiter (ED) –** It is used to mark the end of frame. In Ethernet, length field is used for this purpose. It also contains bits to indicate a damaged frame and identify the frame that is the last in a logical sequence.

- **Frame status (FS) –** It Is a 1-byte field terminating a data frame

### Differences between Token Ring and Token Bus

| Token Ring | Token Bus |
|---|---|
| The token is passed over the physical ring formed by the stations and the coaxial cable network. | The token is passed along the virtual ring of stations connected to a LAN. |
| The stations are connected by ring topology, or sometimes star topology. | The underlying topology that connects the stations is either bus or tree topology. |
| It is defined by IEEE 802.5 standard. | It is defined by IEEE 802.4 standard. |
| The maximum time for a token to reach a station can be calculated here. | It is not feasible to calculate the time for token transfer. |

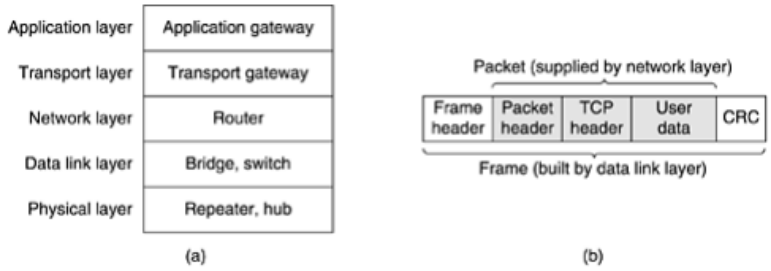


## Bridges

Network connecting devices comes in five categories based on the layer they act up on:

1. Repeater & hub – physical layer
2. Bridge / two layer switch – data link & physical layer
3. Router / three layer switch – network , datalink & physical layer
4. Gateway – on five layers

***Figure 4-46. (a) Which device is in which layer. (b) Frames, packets, and headers.***



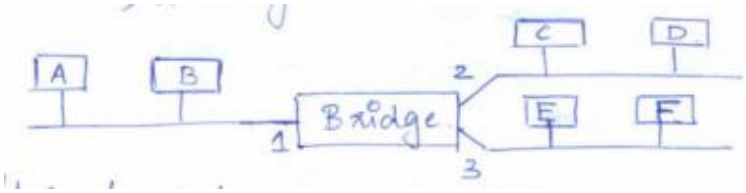

## BRIDGES:

Multiple LANs can be connected by devices called **bridges,** which operate in the data link layer. Bridges examine the data layer link addresses to do routing. Since they are not supposed to examine the payload field of the frames they route, they can transport any kinds of packets.

Why bridges?

- multiple LANs came into existence due to the autonomy of their owners. Two connect multiple LANs bridges needed. Eg: different departments of a same college
- the organization may be geographically spread over several buildings separated by considerable distances. It may be cheaper to have separate LANs in each building and connect them with bridges.
- it may be necessary to split what is logically a single LAN into separate LANs to accommodate the load.
- Using bridges, the total physical distance covered can be increased.
- there is the matter of reliability. On a single LAN, a defective node that keeps outputting a continuous stream of garbage can cripple the LAN. Bridges can be inserted at critical places, like fire doors in a building, to prevent a single node that has gone berserk from bringing down the entire system. a bridge can be programmed to exercise some discretion about what it forwards and what it does not forward.
- bridges can contribute to the organization's security. Most LAN interfaces have a promiscuous mode, in which all frames are given to the computer, not just those addressed to it. Spies and busybodies love this feature. By inserting bridges at various places and being careful not to forward sensitive traffic, a system administrator can isolate parts of the network so that its traffic cannot escape and fall into the wrong hands

Bridge performs **filtering**. Bridge can check the destination address and decide whether the packet is to be forwarded or dropped. If to be forwarded the decision must specify the port. A bridge has a table that maps ports to address.




| Address | Port |
|---|---|
| A | 1 |
| B | 1 |
| C | 2 |
| D | 2 |
| E | 3 |
| F | 3 |

Eg:- of Address
71:2B:13:45:61:41
71:2B:13:45:61:4

Bridges are of four types:

1. routing bridges / source routing bridges
2. transparent bridges
3. Spanning Tree Bridges
4. Remote bridges

**Transparent bridges**

A device that ties two network segments together. Commonly used in Ethernet networks and also called an "adaptive bridge," the transparent bridge learns which node is connected to which port by examining the packets transmitted to the port. Transparent bridge automatically maintains a routing table and update table in response to maintain changing topology. Transparent bridge mechanism consists of five mechanisms:

1. **Learning** - is the process of obtaining the MAC address of devices. When a bridge is first turned on, it has no entries in its bridge table. As traffic passes through the bridge, the sender's MAC addresses are stored in a table along with the associated port on which the traffic was received. This table is often called a bridge table, MAC table, or content addressable memory (CAM) table.
2. **Flooding** - When a bridge does not have an entry in its bridge table for a specific address, it must transparently pass the traffic through all its ports except the source port. This is known as flooding. The source port is not "flooded" because the original traffic came in on this port and already exists on that segment. Flooding allows the bridge to learn, as well as stay transparent to the rest of the network, because no traffic is lost while the bridge is learning.
3. **Filtering** - After the bridge learns the MAC address and associated port of the devices to which it is connected, the benefits of transparent bridging can be seen by way of filtering. Filtering occurs when the source and destination are on the same side (same bridge port) of the bridge
4. **Forwarding** - Forwarding is simply passing traffic from a known device located on one bridge port to another known device located on a different bridge port.
5. **Aging** - In addition to the MAC address and the associated port, a bridge also records the time that the device was learned. Aging of learned MAC addresses allows the bridge to adapt to moves, adds, and changes of devices to the network. After a device is learned, the bridge starts an aging timer. Each time the bridge forwards or filters a frame from a device, it restarts that device's timer. If the bridge doesn't hear from a device in a preset period of time, the aging timer expires and the bridge removes the device from its table. Aging ensures that the bridge tracks only active systems, and ensures that the MAC address table does not consume too much system memory.

Transparent bridge is easy to use, install the bridge and no software changes are needed in hosts. In all the cases, transparent bridge flooded the broadcast and multicast frames.

**source routing bridges**

Source-route transparent (SRT) bridging is a bridging scheme developed by IBM that combines source-route bridging (SRB) and transparent bridging in the same network. SRT is commonly used with token ring networks.

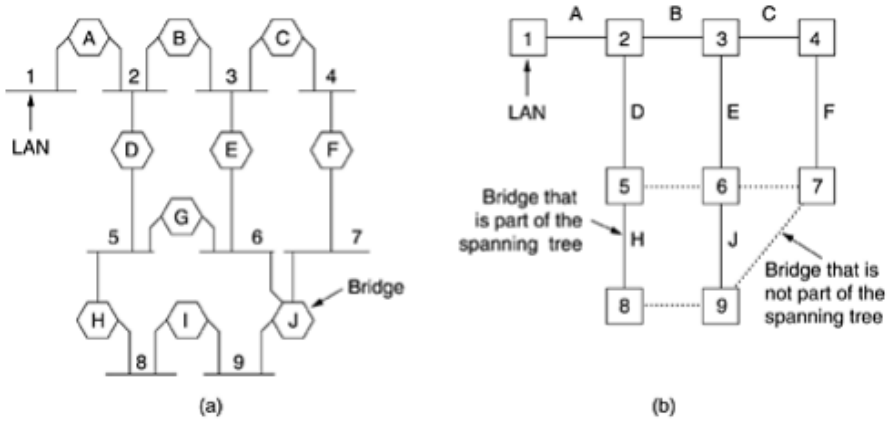
The SRB packet contains the route. So SRB algorithm checks for the RI(Route Information) frame and decides the direction.

| Transparent Bridge | Source Routing Bridge |
|---|---|
| Transparent bridge service is connectionless. | Source Routing Bridge service is connection oriented. |
| In transparent bridge mechanism bridges automatically develop a routing table. | In source routing bridge, bridges do not maintain any routing information. |
| Transparent bridge does not support multipath routing. | Source routing bridge can make use of multiple path to same destination. |
| The path used by transparent bridge between any two hosts may not be the optimal path. | Source route bridge always uses the optimal path. |
| Failures are handled by the transparent bridge on its own. | Host handle the failure of bridge on its own. |
| Transparent bridges are fully transparent to the users. | Source routing bridges are not visible to the hosts. |
| The frame processing delay is more. | The frame processing delay is less. |
| Load sharing is not possible through blocked routes. | Load sharing is possible by judicious choice of routes. |



## Spanning Tree Bridges

***Figure 4-44. (a) Interconnected LANs. (b) A spanning tree covering the LANs. The dotted lines are not part of the spanning tree.***




in Fig. 4-44(a) we see nine LANs interconnected by ten bridges. This configuration can be abstracted into a graph with the LANs as the nodes. An arc connects any two LANs that are

connected by a bridge. The graph can be reduced to a spanning tree by dropping the arcs shown as dotted lines in Fig. 4-44(b). Using this spanning tree, there is exactly one path from every LAN to every other LAN. Once the bridges have agreed on the spanning tree, all forwarding between LANs follows the spanning tree. Since there is a unique path from each source to each destination, loops are impossible.

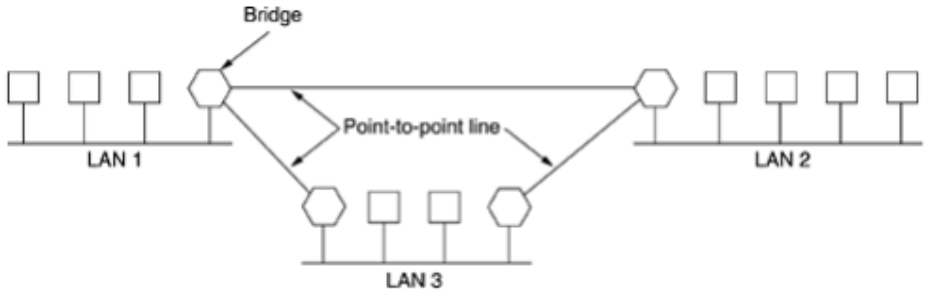
To build the spanning tree:

1. first the bridges have to choose one bridge to be the root of the tree.
2. They make this choice by having each one broadcast its serial number, installed by the manufacturer and guaranteed to be unique worldwide.
3. The bridge with the lowest serial number becomes the root.
4. Next, a tree of shortest paths from the root to every bridge and LAN is constructed. This tree is the spanning tree.
5. If a bridge or LAN fails, a new one is computed.
6. The result of this algorithm is that a unique path is established from every LAN to the root and thus to every other LAN.
7. Although the tree spans all the LANs, not all the bridges are necessarily present in the tree (to prevent loops).
8. Even after the spanning tree has been established, the algorithm continues to run during normal operation in order to automatically detect topology changes and update the tree.

**Remote bridges**

A common use of bridges is to connect two (or more) distant LANs. This goal can be achieved by putting a bridge on each LAN and connecting the bridges pairwise with point-to-point lines (e.g., lines leased from a telephone company). A simple system, with three LANs, is illustrated in Fig. 4-45. The usual routing algorithms apply here.

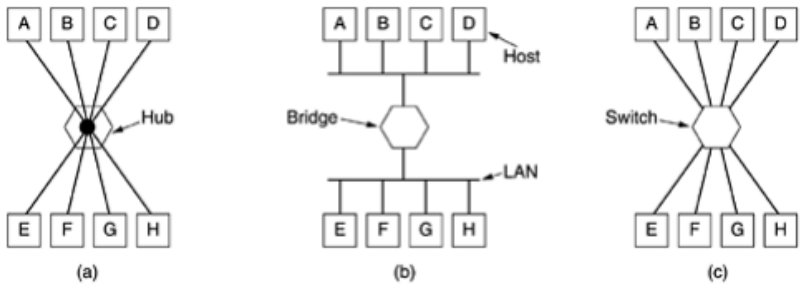
***Figure 4-45. Remote bridges can be used to interconnect distant LANs.***




Various protocols can be used on the point-to-point lines.

## SWITCHES

***Figure 4-47. (a) A hub. (b) A bridge. (c) A switch.***




A switch is most often used to connect individual computers. when host A in Fig. 4-47(b) wants to send a frame to host B, the bridge gets the frame but just discards it. In contrast, in Fig. 4-47(c), the switch must actively forward the frame from A to B because there is no other way for the frame to get there. Since each switch port usually goes to a single computer, switches must have space for many more line cards. Each line card provides buffer space for frames arriving on its ports. Since each port is its own collision domain, switches never lose frames to collisions.

Switches are of two types:

1. **Store and forward switch –** stores the frame in the input buffer until the whole packet arrived.
2. **Cut through switch –** forward the packet to the output buffer as the destination address is received.

Switch uses the information from frame and switching table to find the output port.

Switches are of 2 types:

1. **Two layer switches / Cut Through Switches**
2. **Three layer switches / Routers –** acts in network layer.

## HIGH SPEED LANS / FAST ETHERNET / IEEE 802.3u

Fast Ethernet was designed to compete with LAN protocols such as FDDI or Fiber Channel. Fast Ethernet is backward-compatible with Standard Ethernet, but it can transmit data 10 times faster at a rate of 100 Mbps.

The goals of Fast Ethernet can be summarized as:

1. Upgrade the data rate to 100 Mbps.
2. Make it compatible with Standard Ethernet.
3. Keep the same 48-bit address.
4. Keep the same frame format.
5. Keep the same minimum and maximum frame lengths.

## GIGABIT ETHERNET / IEEE 802.3ab

Goals:

- Increase performance tenfold while maintaining compatibility with all existing Ethernet standards.

- Offer unacknowledged datagram service with both unicast and broadcast, use the same 48-bit addressing scheme already in use, and maintain the same frame format, including the minimum and maximum frame sizes.
- All configurations of gigabit Ethernet use point-to-point links.

**It supports two different modes of operation:**
**full-duplex mode and half-duplex mode**.

- The ‘‘normal’’ mode is **full-duplex mode,** which allows traffic in both directions at the same time. This mode is used when there is a central switch connected to computers (or other switches) on the periphery. In this configuration, all lines are buffered so each computer and switch is free to send frames whenever it wants to. The sender does not have to sense the channel to see if anybody else is using it because contention is impossible. On the line between a computer and a switch, the computer is the only possible sender to the switch, and the transmission will succeed even if the switch is currently sending a frame to the computer. no contention is possible, the CSMA/CD protocol is not used, so the maximum length of the cable is determined by signal strength.
- **half-duplex,** is used when the computers are connected to a hub rather than a switch. A hub does not buffer incoming frames. Instead, it electrically connects all the lines internally, simulating the multidrop-cable used in classic Ethernet. In this mode, collisions are possible, so the standard CSMA/CD protocol is required. Cable length must be 25 meters.
- Two features were added to the standard to increase the maximum cable length to 200 meters. The first feature, called **carrier extension**, tells the hardware to add its own padding after the normal frame to extend the frame to 512 bytes. Since this padding is added by the sending hardware and removed by the receiving hardware, no changes are needed to existing software.
- The second feature, called **frame bursting**, allows a sender to transmit a concatenated sequence of multiple frames in a single transmission. If the total burst is less than 512 bytes, the hardware pads it again. If enough frames are waiting for transmission, this scheme is very efficient and preferred over carrier extension.
- To send bits over various versions of gigabit Ethernet, the 8B/10B encoding was borrowed from another networking technology called **Fibre Channel.** That scheme encodes 8 bits of data into 10-bit codewords that are sent over the wire or fiber, hence the name 8B/10B.
    - The following two rules were used in making the choices: 1. No codeword may have more than four identical bits in a row. 2. No codeword may have more than six 0s or six 1s.
- There is one more extension that was introduced along with gigabit Ethernet. **Jumbo frames** allow for frames to be longer than 1500 bytes, usually up to 9 KB.

**WIRELESS LANS - 802.11 A/B/G/N,**

WLANs are being outfitted with them to connect computers, PDAs, and smart phones to the Internet. Wireless LANs can also be used to let two or more nearby computers communicate without using the Internet. The main wireless LAN standard is **802.11.**

## 802.11 Architecture

There are two modes:

1) **Infrastructure mode** - The most popular mode is to connect clients, such as laptops and smart phones, to another network, such as a company intranet or the Internet. This mode is shown in Fig. 4-23(a). In infrastructure mode, each client is associated with an **AP (Access Point)** that is in turn connected to the other network. The client sends and receives its packets via the AP. Several access points may be connected together, typically by a wired network called a **distribution system,** to form an extended 802.11 network. In this case, clients can send frames to other clients via their APs.
2) **Adhoc mode** - This mode is a collection of computers that are associated so that they can directly send frames to each other. There is no access point.

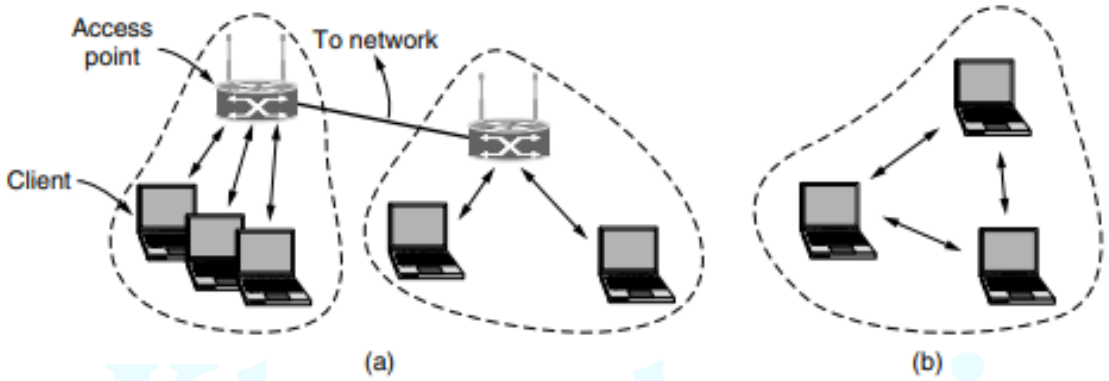
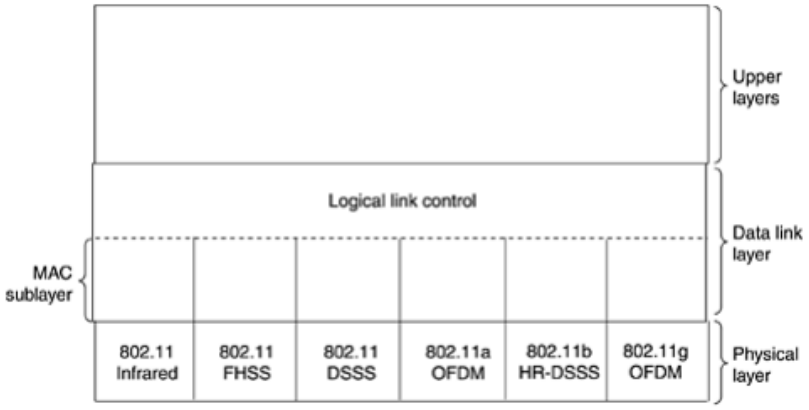



**Figure 4-23.** 802.11 architecture. (a) Infrastructure mode. (b) Ad-hoc mode.



## The 802.11 Protocol Stack

***Figure 4-25. Part of the 802.11 protocol stack.***




The **physical layer** corresponds to the OSI physical layer.

The **data link layer** in all the 802 protocols is split into two or more sublayers. In 802.11, the **MAC (Medium Access Control)** sublayer determines how the channel is allocated, that is, who gets to transmit next. Above it is the **LLC (Logical Link Control)** sublayer, whose job it is to hide the differences between the different 802 variants and make them indistinguishable as far as the network layer is concerned.

1. **The 802.11 Physical Layer**

The 802.11 standard specifies three transmission techniques allowed in the physical layer. The **infrared method** uses much the same technology as television remote controls do. The other two use short-range radio, using techniques called **FHSS and DSSS**. Both of these use a part of the spectrum that does not require licensing. Cordless telephones and microwave ovens also use this band. All of these techniques operate at 1 or 2 Mbps and at low enough power that they do not conflict too much.

Two new techniques were introduced to achieve higher bandwidth. These are called **OFDM** and **HR-DSSS**. They operate at up to 54 Mbps and 11 Mbps, respectively.

A second **OFDM modulation** was introduced, but in a different frequency band from the first one.

**The infrared** option uses diffused (i.e., not line of sight) transmission at 0.85 or 0.95 microns. Two speeds are permitted: 1 Mbps and 2 Mbps. At 1 Mbps, an encoding scheme is used in which a group of 4 bits is encoded as a 16-bit codeword containing fifteen 0s and a single 1, using what is called **Gray code.** Infrared signals cannot penetrate walls, so cells in different rooms are well isolated from each other. Nevertheless, due to the low bandwidth (and the fact that sunlight swamps infrared signals), this is not a popular option.

**FHSS (Frequency Hopping Spread Spectrum)** uses 79 channels, each 1-MHz wide, starting at the low end of the 2.4-GHz ISM band. A **pseudorandom number generator** is used to produce the sequence of frequencies hopped to. As long as all stations use the same seed to the pseudorandom number generator and stay synchronized in time, they will hop to the same frequencies simultaneously. The amount of time spent at each frequency, the **dwell time**, is an adjustable parameter, but must be less than 400 msec. FHSS' randomization provides a fair way to allocate spectrum in the unregulated ISM band. It also provides a modicum of security since an intruder who does not know the hopping sequence or dwell time cannot eavesdrop on transmissions. Over longer distances, multipath fading can be an issue, and FHSS offers good resistance to it. It is also relatively insensitive to radio interference, which makes it popular for building-to-building links. Its main disadvantage is its low bandwidth.

**DSSS (Direct Sequence Spread Spectrum)**, is also restricted to 1 or 2 Mbps. Each bit is transmitted as 11 chips, using what is called a **Barker sequence**. It uses phase shift modulation at 1 Mbaud, transmitting 1 bit per baud when operating at 1 Mbps and 2 bits per baud when operating at 2 Mbps.

The first of the high-speed wireless LANs, 802.11a, uses **OFDM (Orthogonal Frequency Division Multiplexing)** to deliver up to 54 Mbps. Different frequencies are used—52 of them, 48 for data and 4 for synchronization.

**HR-DSSS (High Rate Direct Sequence Spread Spectrum**), another spread spectrum technique, which uses 11 million chips/sec to achieve 11 Mbps in the 2.4-GHz band. It is called **802.11b** but is not a follow-up to 802.11a. In fact, its standard was approved first and it got to market first. Data rates supported by 802.11b are 1, 2, 5.5, and 11 Mbps. The two slow rates run at 1 Mbaud, with 1 and 2 bits per baud, respectively, using phase shift modulation (for compatibility with DSSS). The two faster rates run at 1.375 Mbaud, with 4 and 8 bits per

baud, respectively, using **Walsh/Hadamard codes**. The data rate may be dynamically adapted during operation to achieve the optimum speed possible under current conditions of load and noise. In practice, the operating speed of 802.11b is nearly always 11 Mbps. Although 802.11b is slower than 802.11a, its range is about 7 times greater.

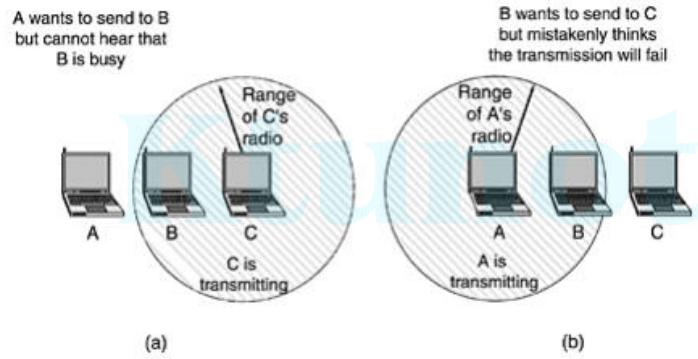
An enhanced version of 802.11b, **802.11g**, was approved by IEEE. It uses the OFDM modulation method of 802.11a but operates in the narrow 2.4-GHz ISM band along with 802.11b. It can operate at up to 54 MBps. The 802.11 committee has produced three different high speed wireless LANs: 802.11a, 802.11b, and 802.11g.

### 2. The 802.11 MAC Sublayer Protocol

two problems persist:

**The hidden station problem :** . Since not all stations are within radio range of each other, transmissions going on in one part of a cell may not be received elsewhere in the same cell. In this example, station C is transmitting to station B. If A senses the channel, it will not hear anything and falsely conclude that it may now start transmitting to B.

***Figure 4-26. (a) The hidden station problem. (b) The exposed station problem.***



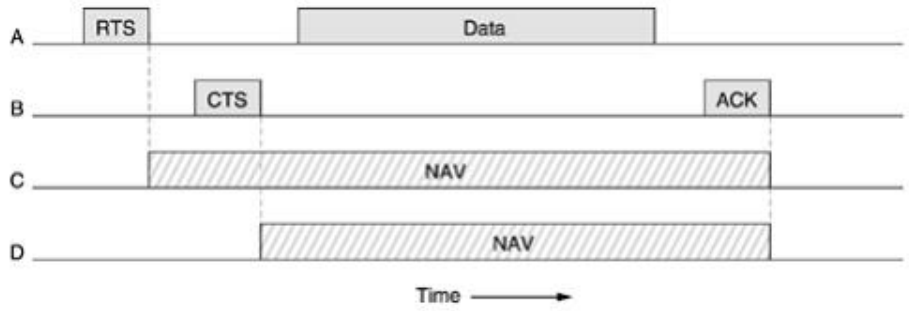

**The exposed station problem**, illustrated in Fig. 4- 26(b). Here B wants to send to C so it listens to the channel. When it hears a transmission, it falsely concludes that it may not send to C, even though A may be transmitting to D (not shown). In addition, most radios are half duplex, meaning that they cannot transmit and listen for noise bursts at the same time on a single frequency. As a result of these problems, 802.11 does not use CSMA/CD.

To deal with this problem, 802.11 supports two modes of operation. The first, called **DCF (Distributed Coordination Function)**, does not use any kind of central control (in that respect, similar to Ethernet). The other, called **PCF (Point Coordination Function),** uses the base station to control all activity in its cell. All implementations must support DCF but PCF is optional.

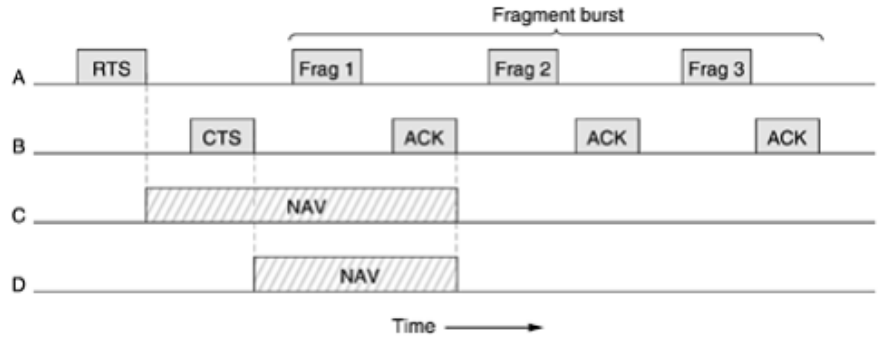
When DCF is employed, 802.11 uses a protocol **called CSMA/CA (CSMA with Collision Avoidance)**. In this protocol, both physical channel sensing and virtual channel sensing are used. Two methods of operation are supported by CSMA/CA.

- In **the first method**, when a station wants to transmit, it senses the channel. If it is idle, it just starts transmitting. It does not sense the channel while transmitting but emits its entire frame, which may well be destroyed at the receiver due to interference there. If the channel is busy, the sender defers until it goes idle and then starts transmitting. If a collision occurs, the colliding stations wait a random time, using the Ethernet binary exponential backoff algorithm, and then try again later.

***Figure 4-27. The use of virtual channel sensing using CSMA/CA.***




- The other mode of CSMA/CA operation is based on **MACAW** and uses virtual channel sensing, as illustrated in Fig. 4-27. In this example, A wants to send to B. C is a station within range of A (and possibly within range of B, but that does not matter). D is a station within range of B but not within range of A. Figure 4-27. The use of virtual channel sensing using CSMA/CA. The protocol starts when A decides it wants to send data to B. It begins by sending an RTS frame to B to request permission to send it a frame. When B receives this request, it may decide to grant permission, in which case it sends a CTS frame back. Upon receipt of the CTS, A now sends its frame and starts an ACK timer. Upon correct receipt of the data frame, B responds with an ACK frame, terminating the exchange. If A's ACK timer expires before the ACK gets back to it, the whole protocol is run again.
  Now let us consider this exchange from the viewpoints of C and D. C is within range of A, so it may receive the RTS frame. If it does, it realizes that someone is going to send data soon, so for the good of all it desists from transmitting anything until the exchange is completed. From the information provided in the RTS request, it can estimate how long the sequence will take, including the final ACK, so it asserts a kind of virtual channel busy for itself, indicated by NAV (Network Allocation Vector) in Fig. 4-27. D does not hear the RTS, but it does hear the CTS, so it also asserts the NAV signal for itself. Note that the NAV signals are not transmitted; they are just internal reminders to keep quiet for a certain period of time.
- To deal with the problem of noisy channels, 802.11 allows frames to be fragmented into smaller pieces, each with its own checksum. The fragments are individually numbered and acknowledged using a stop-and-wait protocol (i.e., the sender may not transmit fragment k + 1 until it has received the acknowledgment for fragment k). Once the channel has been acquired using RTS and CTS, multiple fragments can be sent in a row, as shown in Fig. 4-28. sequence of fragments is called a **fragment burst.**

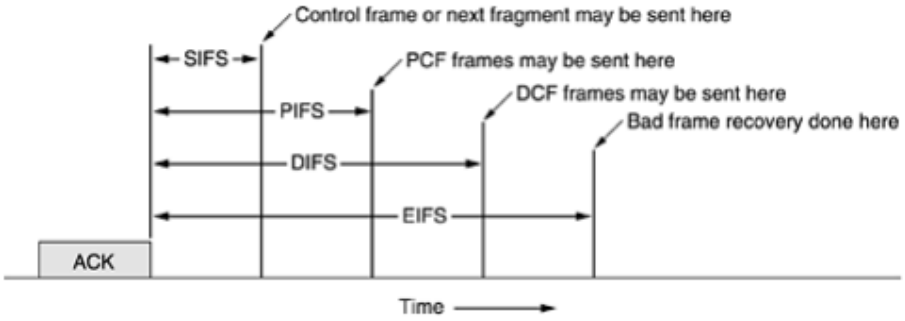



Fragmentation increases the throughput by restricting retransmissions to the bad fragments rather than the entire frame. The fragment size is not fixed by the standard but is a parameter of each cell and can be adjusted by the base station. The NAV mechanism keeps other stations quiet only until the next acknowledgement, but another mechanism (described below) is used to allow a whole fragment burst to be sent without interference.All of the above discussion applies to the 802.11 DCF mode. In this mode, there is no central control, and stations compete for air time, just as they do with Ethernet.

The other allowed mode is PCF, in which the base station polls the other stations, asking them if they have any frames to send. Since transmission order is completely controlled by the base station in PCF mode, no collisions ever occur. The standard prescribes the mechanism for polling, but not the polling frequency, polling order, or even whether all stations need to get equal service.
The basic mechanism is for the base station to broadcast **a beacon frame** periodically (10 to 100 times per second). The beacon frame contains system parameters, such as hopping sequences and dwell times (for FHSS), clock synchronization, etc. It also invites new stations to sign up for polling service. Once a station has signed up for polling service at a certain rate, it is effectively guaranteed a certain fraction of the bandwidth, thus making it possible to give quality-of-service guarantees.

PCF and DCF can coexist within one cell. At first it might seem impossible to have central control and distributed control operating at the same time, but 802.11 provides a way to achieve this goal. It works by carefully defining the interframe time interval. After a frame has been sent, a certain amount of dead time is required before any station may send a frame. Four different intervals are defined, each for a specific purpose. The four intervals are depicted in Fig. 4-29.

**Figure 4-29. Interframe spacing in 802.11**



The shortest interval is **SIFS (Short InterFrame Spacing).** It is used to allow the parties in a single dialog the chance to go first. This includes letting the receiver send a CTS to respond to an RTS, letting the receiver send an ACK for a fragment or full data frame, and letting the sender of a fragment burst transmit the next fragment without having to send an RTS again.

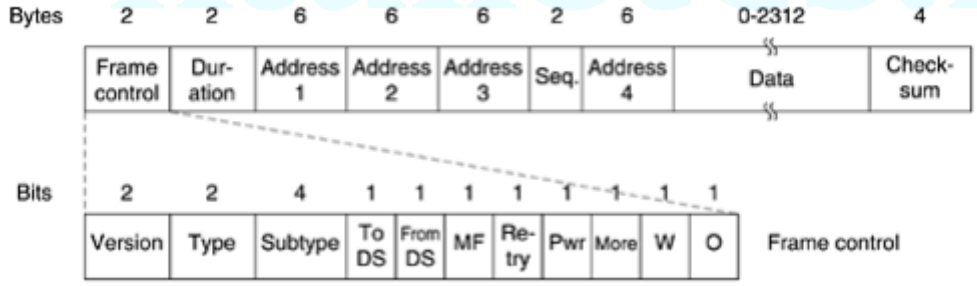
There is always exactly one station that is entitled to respond after a SIFS interval. If it fails to make use of its chance and a time **PIFS (PCF InterFrame Spacing)** elapses, the base station may send a beacon frame or poll frame. This mechanism allows a station sending a data frame or fragment sequence to finish its frame without anyone else getting in the way, but gives the base station a chance to grab the channel when the previous sender is done without having to compete with eager users.

If the base station has nothing to say and a time **DIFS (DCF InterFrame Spacing)** elapses, any station may attempt to acquire the channel to send a new frame. The usual contention rules apply, and binary exponential backoff may be needed if a collision occurs.

The last time interval, **EIFS (Extended InterFrame Spacing),** is used only by a station that has just received a bad or unknown frame to report the bad frame. The idea of giving this event the lowest priority is that since the receiver may have no idea of what is going on, it should wait a substantial time to avoid interfering with an ongoing dialog between two stations.

**The 802.11 Frame Structure**

The 802.11 standard defines three different classes of frames on the wire: data, control, and management.




- **The Frame Control field**. It itself has 11 subfields.
  - **Protocol version**, which allows two versions of the protocol to operate at the same time in the same cell.
  - **The Type** (data, control, or management) and **Subtype fields** (e.g., RTS or CTS).
  - The **To DS** and **From DS** bits indicate the frame is going to or coming from the intercell distribution system (e.g., Ethernet).
  - **The MF** bit means that more fragments will follow.
  - **The Retry** bit marks a retransmission of a frame sent earlier.
  - **The Power management** bit is used by the base station to put the receiver into sleep state or take it out of sleep state.
  - **The More bit** indicates that the sender has additional frames for the receiver.

    - **The W bit** specifies that the frame body has been encrypted using the WEP (Wired Equivalent Privacy) algorithm.
    - **The O bit** tells the receiver that a sequence of frames with this bit on must be processed strictly in order.
- the **Duration field**, tells how long the frame and its acknowledgement will occupy the channel.
- The frame header contains **four addresses**, all in standard IEEE 802 format. The source and destination. The other two addresses are used for the source and destination base stations for intercell traffic.
- **The Sequence field** allows fragments to be numbered. Of the 16 bits available, 12 identify the frame and 4 identify the fragment.
- **The Data field** contains the payload, up to 2312 bytes, followed by the usual **Checksum**.
- **Management frames** have a format similar to that of data frames, except without one of the base station addresses, because management frames are restricted to a single cell.
- **Control frames** are shorter still, having only one or two addresses, no Data field, and no Sequence field. The key information here is in the Subtype field, usually RTS, CTS, or ACK.

**services**

The 802.11 standard states that each conformant wireless LAN must provide nine services. These services are divided into two categories: five distribution services and four station services. The distribution services relate to managing cell membership and interacting with stations outside the cell. In contrast, the station services relate to activity within a single cell.

The five distribution services are provided by the base stations and deal with station mobility as they enter and leave cells, attaching themselves to and detaching themselves from base stations. They are as follows.

1. Association. This service is used by mobile stations to connect themselves to base stations.
2. Disassociation. Either the station or the base station may disassociate, thus breaking the relationship.
3. Reassociation. A station may change its preferred base station using this service.
4. Distribution. This service determines how to route frames sent to the base station.
5. Integration. If a frame needs to be sent through a non-802.11 network with a different addressing scheme or frame format, this service handles the translation from the 802.11 format to the format required by the destination network.

The remaining four services are intracell (i.e., relate to actions within a single cell). They are used after association has taken place and are as follows.

1. Authentication. Because wireless communication can easily be sent or received by unauthorized stations, a station must authenticate itself before it is permitted to send data. After a mobile station has been associated by the base station (i.e., accepted into its cell), the base station sends a special challenge frame to it to see if the mobile station knows the secret key (password) that has been assigned to it. It proves its knowledge of the secret key by encrypting the challenge frame and sending it back to the base station. If the result is correct, the mobile is

fully enrolled in the cell. In the initial standard, the base station does not have to prove its identity to the mobile station.
2. Deauthentication. When a previously authenticated station wants to leave the network, it is deauthenticated. After deauthentication, it may no longer use the network.
3. Privacy. For information sent over a wireless LAN to be kept confidential, it must be encrypted. This service manages the encryption and decryption.
4. Data delivery. Finally, data transmission is what it is all about, so 802.11 naturally provides a way to transmit and receive data.

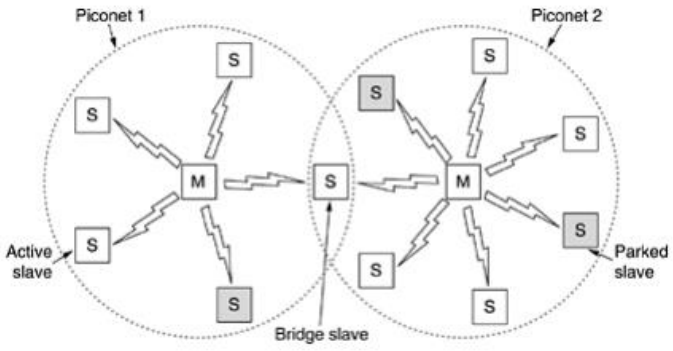
**802.15 / BLUETOOTH**

Bluetooth is a **wireless LAN** technology designed to connect devices of different functions such as telephones, notebooks, computers (desktop and laptop), cameras, printers, coffee makers, and so on. A Bluetooth LAN is an **ad hoc network**, which means that the network is formed spontaneously; the devices, sometimes called **gadgets**, find each other and make a network called a **piconet**.

Bluetooth technology is the implementation of a protocol defined by the IEEE 802.15 standard. The standard defines a wireless personal-area network (PAN) operable in an area the size of a room or a hall.

Architecture Bluetooth defines two types of networks: **piconet and scatternet.**
The basic unit of a Bluetooth system is a **piconet**, which consists of a master node and up to seven active slave nodes within a distance of 10 meters. Multiple piconets can exist in the same (large) room and can even be connected via a bridge node. An interconnected collection of piconets is called a scatternet.
All the secondary stations synchronize their clocks and hopping sequence with the primary. Note that a piconet can have only one primary station. The communication between the primary and the secondary can be one-to-one or one-to-many. Figure 1 shows a piconet.

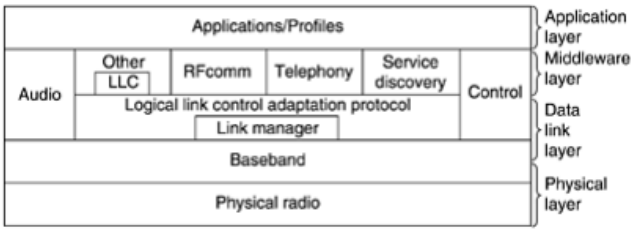
***Figure 4-35. Two piconets can be connected to form a sca***




In addition to the seven active slave nodes in a piconet, there can be up to 255 parked nodes in the net. These are devices that the master has switched to a low-power state to reduce the drain on their batteries. In parked state, a device cannot do anything except respond to an activation or beacon signal from the master. There are also two intermediate power states, hold and sniff, but these will not concern us here.

**Bluetooth protocol stack**

***Figure 4-37. The 802.15 version of the Bluetooth protocol architecture.***




- **physical radio layer**, deals with radio transmission and modulation. The goal of making the system inexpensive.

- The **baseband layer** is somewhat analogous to the MAC sublayer but also includes elements of the physical layer. It deals with how the master controls time slots and how these slots are grouped into frames.Each frame is transmitted over a logical channel, called **a link,** between the master and a slave. Two kinds of links exist.
  - The first is the **ACL (Asynchronous Connection-Less) link,** which is used for packet-switched data available at irregular intervals. These data come from the L2CAP layer on the sending side and are delivered to the L2CAP layer on the receiving side. ACL traffic is delivered on a best-efforts basis. No guarantees are given. Frames can be lost and may have to be retransmitted. A slave may have only one ACL link to its master.
  - The other is the **SCO (Synchronous Connection Oriented) link**, for real-time data, such as telephone connections. This type of channel is allocated a fixed slot in each direction. Due to the time-critical nature of SCO links, frames sent over them are never retransmitted. Instead, forward error correction can be used to provide high reliability. A slave may have up to three SCO links with its master.

- **The link manager** handles the establishment of logical channels between devices, including power management, authentication, and quality of service.

- **The logical link control adaptation protocol** (often called L2CAP) shields the upper layers from the details of transmission.The L2CAP layer has three major functions.
  - First, it accepts packets of up to 64 KB from the upper layers and breaks them into frames for transmission. At the far end, the frames are reassembled into packets again.
  - Second, it handles the multiplexing and demultiplexing of multiple packet sources. When a packet has been reassembled, the L2CAP layer determines which upper-layer protocol to hand it to.
  - Third, L2CAP handles the quality of service requirements, both when links are established and during normal operation. Also negotiated at setup time is the maximum payload size allowed, to prevent a large-packet device from drowning a small-packet device. This feature is needed because not all devices can handle the 64-KB maximum packet.

- **The audio and control protocols** deal with audio and control.

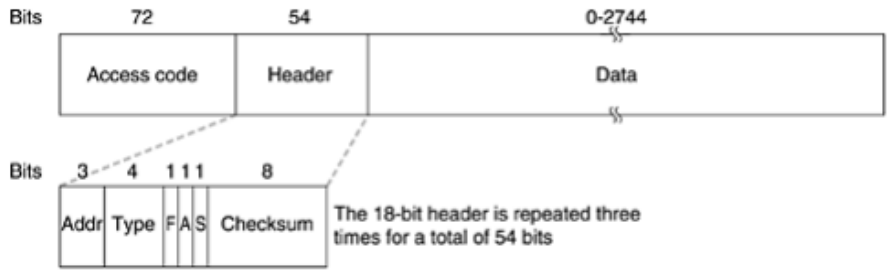
- **The middleware layer**, which contains a mix of different protocols.
  - **The 802 LLC** was inserted here by IEEE for compatibility with its other 802 networks.
  - **RFcomm (Radio Frequency communication**) is the protocol that emulates the standard serial port found on PCs for connecting the keyboard, mouse, and modem, among other devices. It has been designed to allow legacy devices to use it easily.
  - **The telephony protocol** is a real-time protocol used for the three speech-oriented profiles. It also manages call setup and termination.
  - **The service discovery** protocol is used to locate services within the network.

- **The top layer** is where the **applications** and profiles are located. They make use of the protocols in lower layers to get their work done. Each application has its own dedicated subset of the protocols. Specific devices, such as a headset, usually contain only those protocols needed by that application and no others.



**The Bluetooth Frame Structure**

***Figure 4-38. A typical Bluetooth data frame.***




It begins with **an access code** that usually identifies the master so that slaves within radio range of two masters can tell which traffic is for them.
Next comes a **54-bit header** containing typical MAC sublayer fields.
Then comes **the data field**, of up to 2744 bits (for a five-slot transmission). For a single time slot, the format is the same except that the data field is 240 bits.

**at the header**. **The Address field** identifies which of the eight active devices the frame is intended for. **The Type field** identifies the frame type (ACL, SCO, poll, or null), the type of error correction used in the data field, and how many slots long the frame is. **The Flow bit** is asserted by a slave when its buffer is full and cannot receive any more data. This is a primitive form of flow control. **The Acknowledgement bit** is used to piggyback an ACK onto a frame.

**The Sequence bit** is used to number the frames to detect retransmissions. The protocol is stop-and-wait, so 1 bit is enough. Then comes the 8-bit header **Checksum.**

## PPP

Within a single building, LANs are widely used for interconnection, but most wide-area network infrastructure is built up from point-to-point lines.

The data link protocols found on point-to-point lines in the Internet in two common situations.

- The first situation is when packets are sent over SONET optical fiber links in wide-area networks. These links are widely used, for example, to connect routers in the different locations of an ISP's network.
- The second situation is for ADSL links running on the local loop of the telephone network at the edge of the Internet. These links connect millions of individuals and businesses to the Internet

The Internet needs point-to-point links for these uses, as well as dial-up modems, leased lines, and cable modems, and so on. A standard protocol called PPP (Point-to-Point Protocol) is used to send packets over these links. PPP is defined in RFC 1661 and further elaborated in RFC 1662 and other RFCs. SONET and ADSL links both apply PPP, but in different ways.

PPP improves on an earlier, simpler protocol called **SLIP (Serial Line Internet Protocol)** and is used to handle error detection link configuration, support multiple protocols, permit authentication, and more. With a wide set of options, PPP provides three main features:

1. A framing method that unambiguously delineates the end of one frame and the start of the next one. The frame format also handles error detection.
2. A link control protocol for bringing lines up, testing them, negotiating options, and bringing them down again gracefully when they are no longer needed. This protocol is called LCP (Link Control Protocol).
3. A way to negotiate network-layer options in a way that is independent of the network layer protocol to be used. The method chosen is to have a different NCP (Network Control Protocol) for each network layer supported.

### PPP Frame Format:

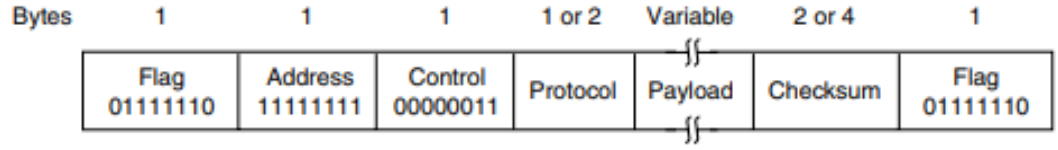



**Figure 3-24.** The PPP full frame format for unnumbered mode operation.

All PPP frames begin with the **standard HDLC flag byte** of 0x7E (01111110). The flag byte is stuffed if it occurs within the Payload field using the escape byte 0x7D

After the start-of-frame flag byte comes the **Address field**. This field is always set to the binary value 11111111 to indicate that all stations are to accept the frame. Using this value avoids the issue of having to assign data link addresses.

The Address field is followed by the **Control field**, the default value of which is 00000011. This value indicates an unnumbered frame.
Since the Address and Control fields are always constant in the default configuration, LCP provides the necessary mechanism for the two parties to negotiate an option to omit them altogether and save 2 bytes per frame.

The fourth PPP field is the **Protocol field.** Its job is to tell what kind of packet is in the Payload field. Codes starting with a 0 bit are defined for IP version 4, IP version 6, and other network layer protocols that might be used. Codes starting with a 1 bit are used for PPP configuration protocols, including LCP and a different NCP for each network layer protocol supported. The default size of the Protocol field is 2 bytes, but it can be negotiated down to 1 byte using LCP.

The **Payload field** is variable length, up to some negotiated maximum. If the length is not negotiated using LCP during line setup, a default length of 1500 bytes is used. Padding may follow the payload if it is needed.

After the Payload field comes the **Checksum field**, which is normally 2 bytes, but a 4-byte checksum can be negotiated