Data Link Control

1. The two main functions of the data link layer are
2. data link control and media access control.

3. Data link control functions include

4. framing, flow and error control, and software implemented protocols that provide smooth and reliable transmission of frames between nodes

FRAMING

1. The data link layer needs to pack bits into frames, so that each frame is distinguishable from another.
2. Framing in the data link layer separates a message from one source to a destination, or from other messages to other destinations, by adding a sender address and a destination address.
3. The whole message could be packed in one frame, that is not normally done. One reason is that a frame can be very large, making flow and error control very inefficient. When a message is carried in one very large frame, even a single-bit error would require the retransmission of the whole message. When a message is divided into smaller frames, a single-bit error affects only that small frame.

I. Fixed-Size Framing

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

II. Variable-Size Framing

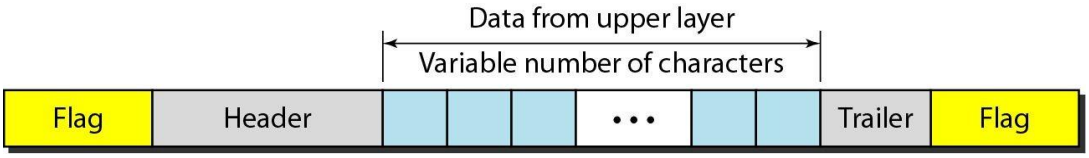
In variable-size framing, we need a way to define the end of the frame and the beginning of the next.

two approaches were used for this purpose:

I. ***A Character-Oriented Approach***

II. ***A Bit-Oriented Approach.***

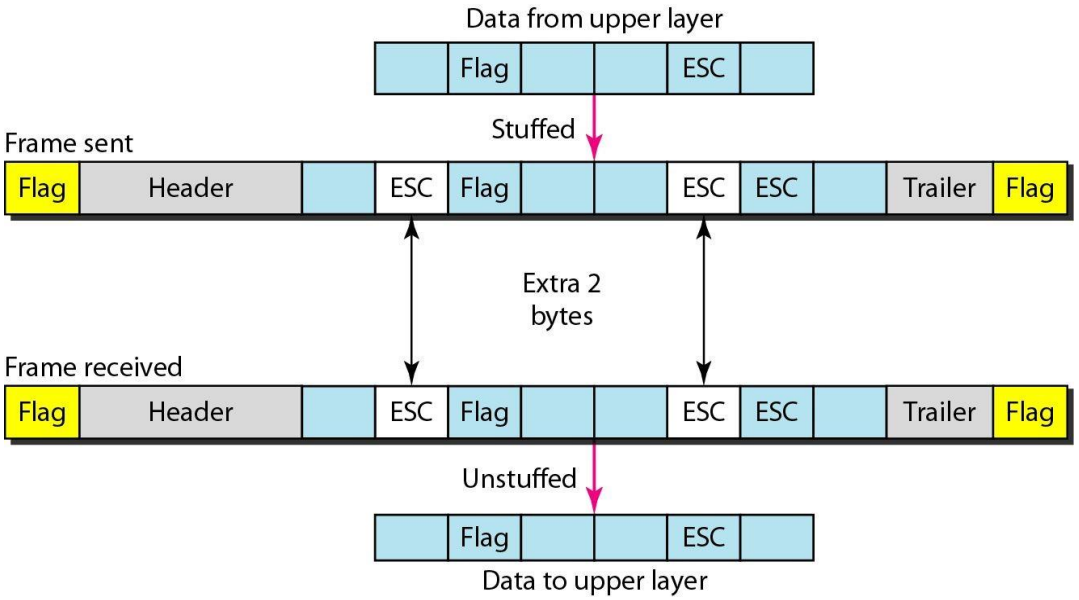
A frame in a character-oriented protocol



In a character-oriented protocol, data to be carried are 8-bit characters from a coding system such as ASCII (see Appendix A). The header, which normally carries the source and destination addresses and other control information, and the trailer, which carries error detection or error correction redundant bits, are also multiples of 8 bits. To separate one frame from the next, an 8-bit (I-byte) flag is added at the beginning and the end of a frame. The flag, composed of protocol-dependent special characters, signals the start or end of a frame. Figure 11.1 shows the format of a frame in a character-oriented protocol.

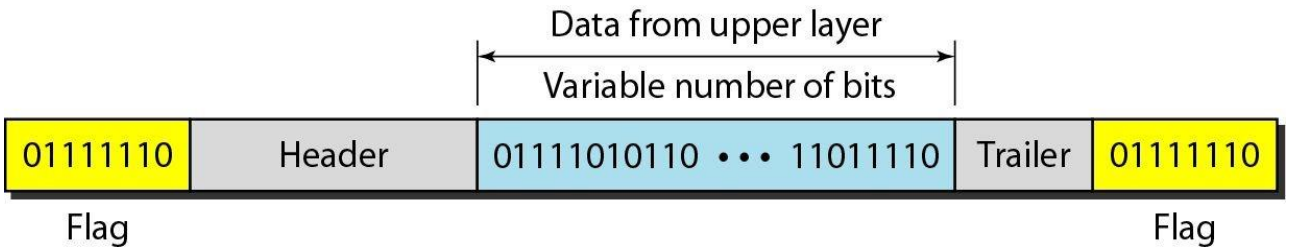
Byte stuffing is the process of adding 1extra byte whe there is a flag or escape character in the text.

Byte stuffing and unstuffing



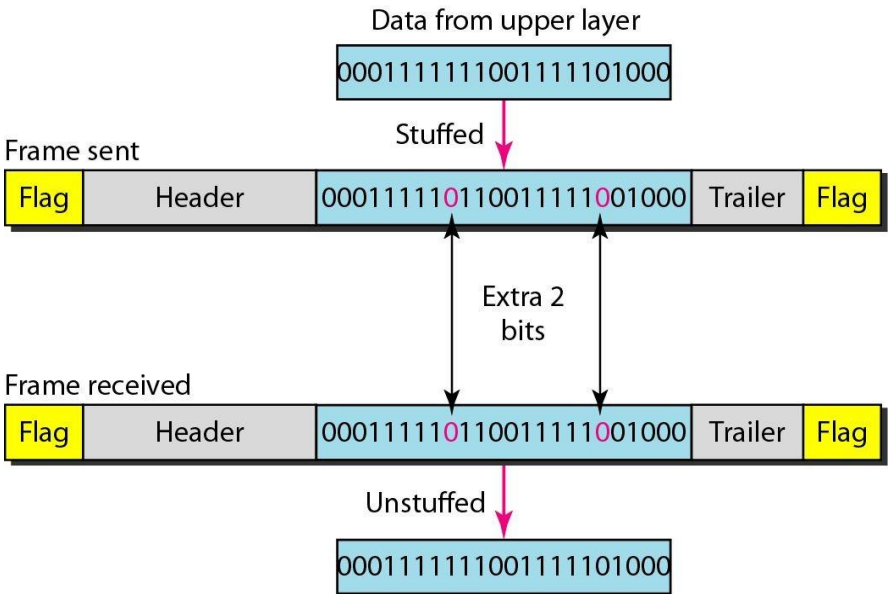
Bit-Oriented Protocols

1. In a bit-oriented protocol, the data section of a frame is a sequence of bits to be interpreted by the upper layer.
2. In addition to headers (and possible trailers), we still need a delimiter to separate one frame from the other. Most protocols use a special 8-bit pattern flag 01111110 as the delimiter to define the beginning and the end of the frame.
3. If the flag pattern appears in the data, we need to somehow inform the receiver that this is not the end of the frame, We do this by stuffing 1 single bit (instead of I byte) to prevent the pattern from looking like a flag. The strategy is called bit stuffing. In bit stuffing, if a 0 and five consecutive I bits are encountered, an extra 0 is added. This extra stuffed bit is eventually removed from the data by the receiver



Bit stuffing is the process of adding one extra 0 whenever five consecutive 1s follow a 0 in the data, so that the receiver does not mistake the pattern 0111110 for a flag.

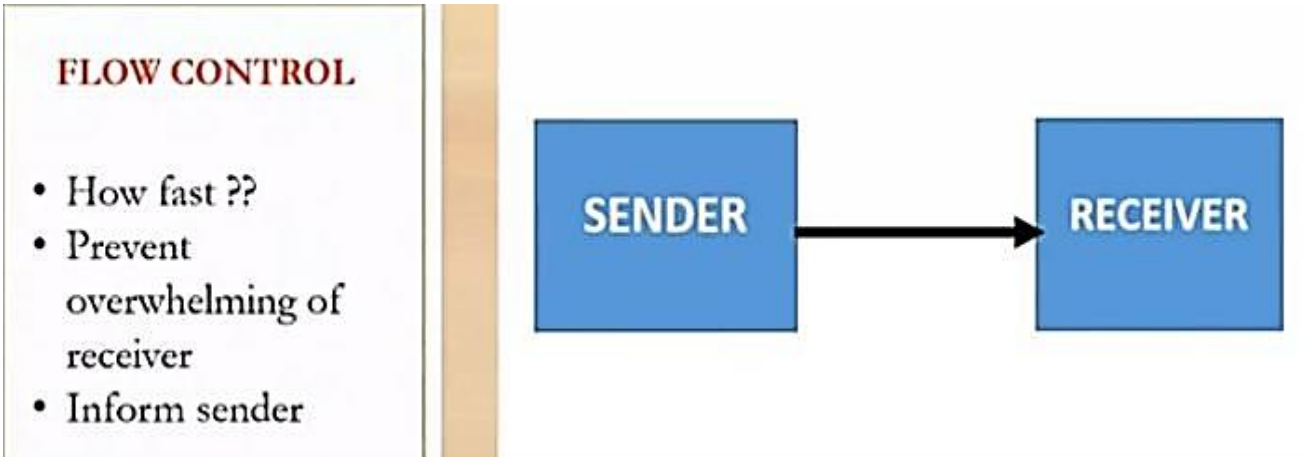
Bit stuffing and unstuffing



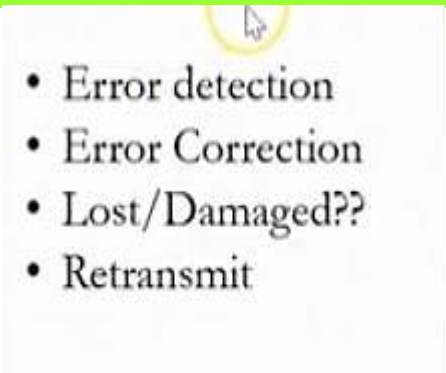
11-1 FLOW AND ERROR CONTROL

The most important responsibilities of the data link layer are flow control and error control. Collectively, these functions are known as data link control.

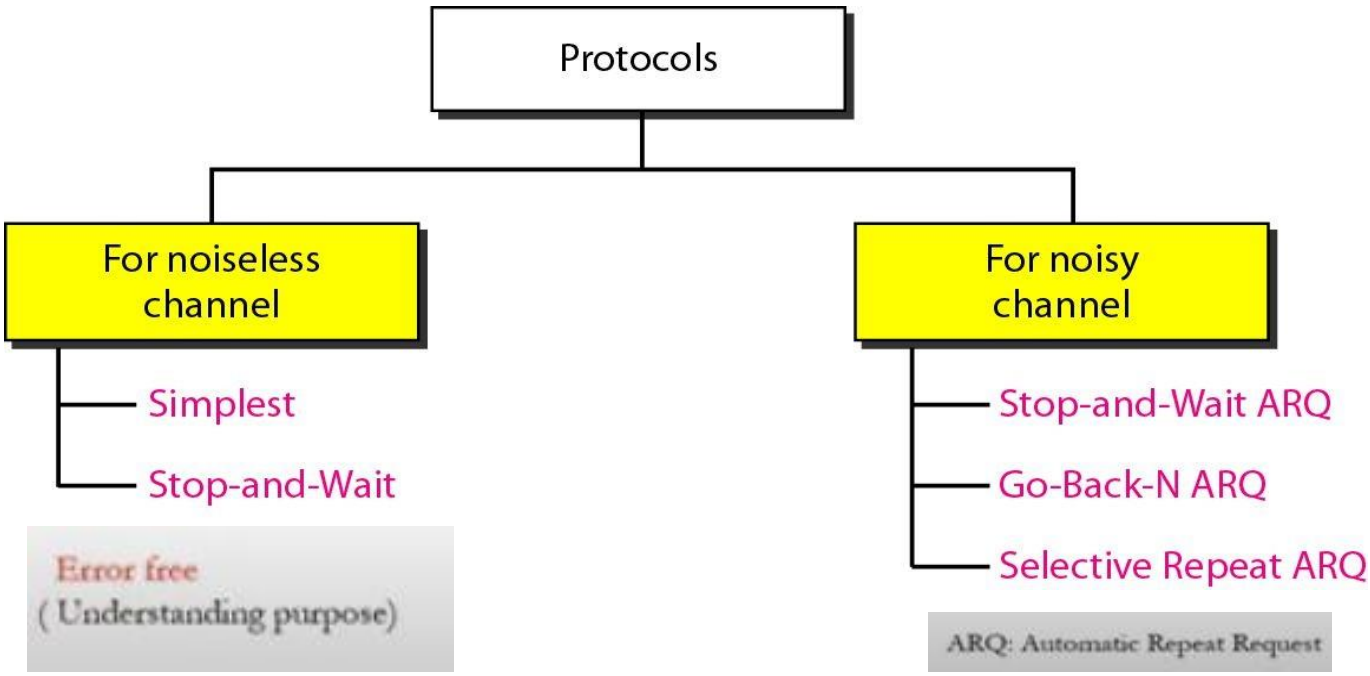
Flow control refers to a set of proceduresused to restrict the amount of data that the sender can send beforewaiting for acknowledgment.



Error control in the data link layer is based on automatic repeat request, which is the retransmission of data.



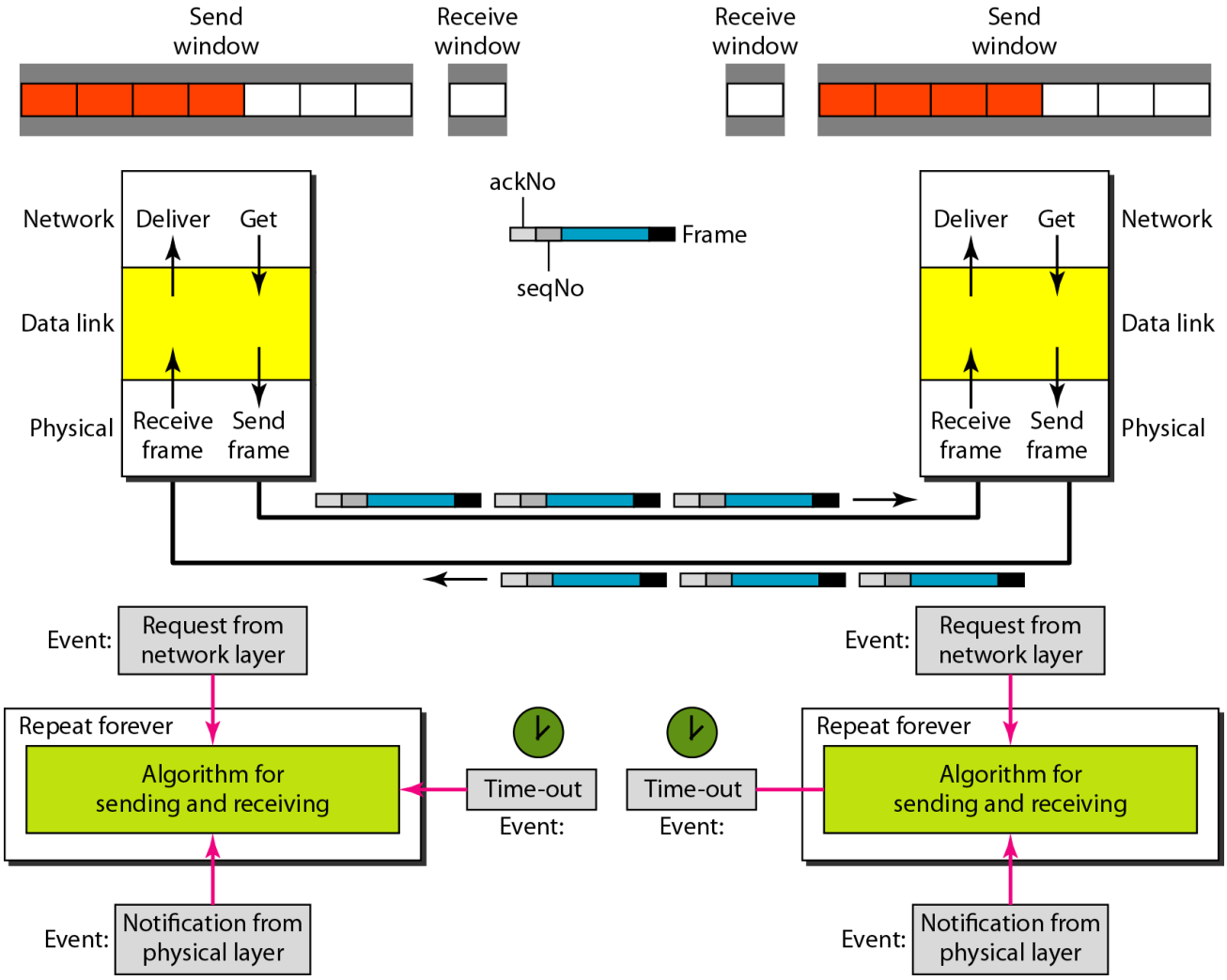
11-2 PROTOCOLS



Now let us see how the data link layer can combine framing, flow control, and error control to achieve the delivery of data from one node to another. The protocols are normally implemented in software by using one of the common programming languages. To make our discussions language-free, we have written in pseudocode a version of each protocol that concentrates mostly on the procedure instead of delving into the details of language rules.

1. PIGGYBACKING

1) Mostly the transfer of data is bi-directional [Full duplex transmission].From node A to node B and from node B to node A.This means that the control information also needs to flow in both directions.
2) Piggybacking is a method of attaching acknowledgement to the outgoing data frame.
3) When a frame is carrying data from A to B, it can also carry control information about arrived (or lost) frames from B; when a frame is carrying data from B to A, it can also carry control information about the arrived (or lost) frames from A.
4) When host A sends a data frame to B, then B does not send acknowledgment immediately.
5) The acknowledgement is delayed until the next frame of host B available for transmission.
6) The process of delaying acknowledgement, so that it can be attached to the outgoing data frame is called piggybacking.
7) In the design for a Go-Back-N ARQ using piggybacking , each node now has two windows: one send window and one receive window. Both also need to use a timer. Both are involved in three types of events: request, arrival, and time-out. However, the arrival event here is complicated; when a frame arrives, the site needs to handle control information as well as the frame itself



III. HDLC

1) High-level Data Link Control (HDLC) is a group of communication protocols of the data link layer for transmitting data between network points or nodes.
2) Since it is a data link protocol, data is organized into frames.
3) A frame is transmitted via the network to the destination that verifies its successful arrival.
4) It is a bit - oriented protocol that is applicable for both point - to - point and multipoint communications

High-level Data Link Control (HDLC) is a bit-oriented protocol for communication over point-to-point and multipoint links. It implements the ARQ mechanisms we discussed in this chapter.

Topics discussed in this section:

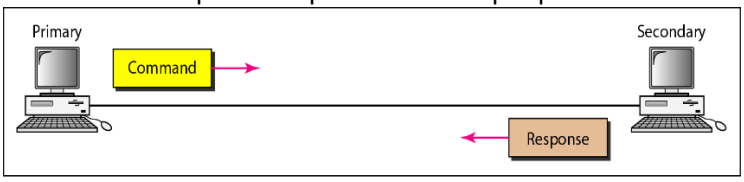
1. *Configurations and Transfer Modes*
2. *Frames*
3. *Control Field*

2. Configurations and Transfer Modes

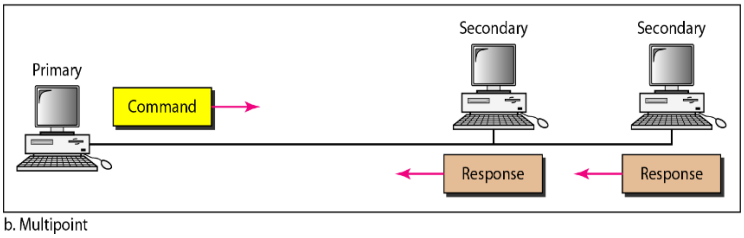
HDLC provides two common transfer modes : normal response mode (NRM) asynchronous balanced mode (ABM).

1. Normal response mode (NRM) :

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



a. Point-to-point



b. Multipoint

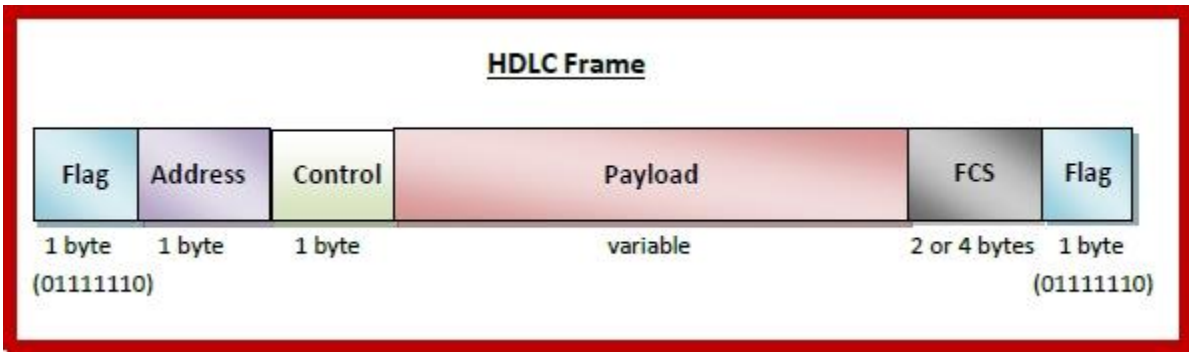
2. Asynchronous Balanced Mode

In asynchronous balanced mode (ABM), the configuration is balanced. The link is point-to-point, and each station can function as a primary and a secondary

3. HDLC frames

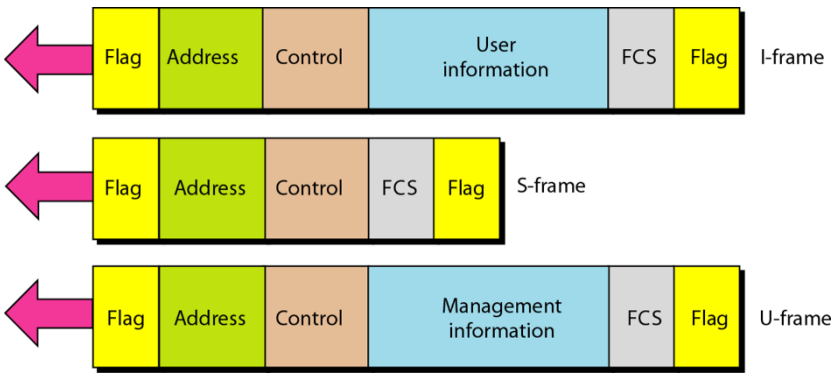
HDLC is a bit - oriented protocol where each frame contains up to six fields. The structure varies according to the type of frame. The fields of a HDLC frame are –

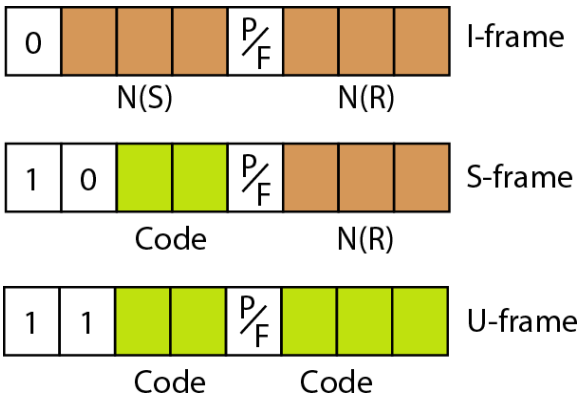
- **Flag** – It is an 8-bit sequence that marks the beginning and the end of the frame. The bit pattern of the flag is 01111110.
- **Address** – It contains the address of the receiver. If the frame is sent by the primary station, it contains the address(es) of the secondary station(s). If it is sent by the secondary station, it contains the address of the primary station. The address field may be from 1 byte to several bytes.
- **Control** – It is 1 or 2 bytes containing flow and error control information.
- **Payload** – This carries the data from the network layer. Its length may vary from one network to another.
- **FCS** – It is a 2 byte or 4 bytes frame check sequence for error detection. The standard code used is CRC (cyclic redundancy code)



4. Types of HDLC Frames

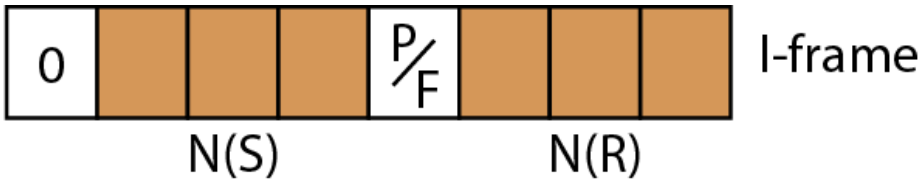
There are three types of HDLC frames. The type of frame is determined by the control field of the frame





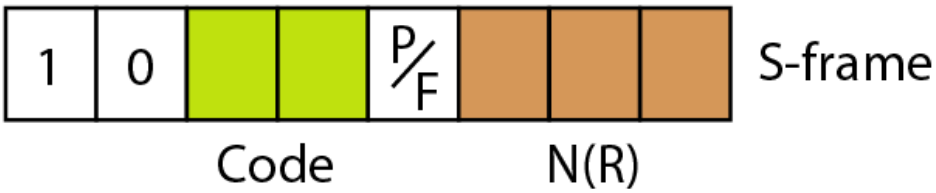
5. Control Field for I-Frames :

1) I-frames are designed to carry user data from the network layer. In addition, they can include flow and error control information (piggybacking).
2) The subfields in the control field are :
    a. **The first bit defines the type**.
    If the first bit of the control field is 0, this means the frame is an I-frame.
    b. **The next 3 bits, called N(S),**
    define the sequence number of the frame. Note that with 3 bits, we can define a sequence number between ° and 7; but in the extension format, in which the control field is 2 bytes, this field is larger.
    c. **The last 3 bits, called N(R),**
    correspond to the acknowledgment number when piggybacking is used.
    d. **The single bit between N(S) and N(R) is called the PIF bit.**
    The PIF field is a single bit with a dual purpose. It has meaning only when it is set (bit = 1) and can mean poll or final. It means poll when the frame is sent by a primary station to a secondary (when the address field contains the address of the receiver). It means final when the frame is sent by a secondary to a primary (when the address field contains the address of the sender).



6. Control Field for S-Frames :

1) Supervisory frames are used for flow and error control
2) S-frames do not have information fields.
3) **If the first 2 bits** of the control field is 10, this means the frame is an S- frame.
4) **The last 3 bits, called N(R),** corresponds to the acknowledgment number (ACK) or negative acknowledgment number (NAK) depending on the type of S-frame.
5) The 2 bits called code is used to define the type of S-frame itself. With 2 bits, we can have four types of S-frames:
    i. **Receive ready (RR):**
    If the value of the code subfield is 00, it is an RR S-frame. This kind of frame acknowledges the receipt of a safe and sound frame or group of frames. In this case, the value N(R) field defines the acknowledgment number.
    ii. **Receive not ready (RNR):**
    If the value of the code subfield is 10, it is an RNR S-frame. This kind of frame is an RR frame with additional functions. It acknowledges the receipt of a frame or group of frames, and it announces that the receiver is busy and cannot receive more frames. It acts as a kind of congestion control mechanism by asking the sender to slow down.
    iii. **Reject (REJ):**
    If the value of the code subfield is 01, it is a REJ S-frame. This is a NAK frame, but not like the one used for Selective Repeat ARQ. It is a NAK that can be used in Go-Back-N ARQ to improve the efficiency of the process by informing the sender, before the sender time expires, that the last frame is lost or damaged. The value of N(R) is the negative acknowledgment number.
    iv. **Selective reject (SREJ):**
    If the value of the code subfield is 11, it is an SREJ S-frame. This is a NAK frame used in Selective Repeat ARQ. Note that the HDLC Protocol uses the term selective reject instead of selective repeat. The value of N(R) is the negative acknowledgment number



7. Control Field for U-Frames :

1) Unnumbered frames are used to exchange session management and
2) control information between connected devices.
3) Unlike S-frames, U-frames contain an information field, but one used for system management information, not user data.
4) U-frame codes are divided into two sections: a 2-bit prefix before the P/F bit and a 3-bit suffix after the P/F bit.
5) Together, these two segments (5 bits) can be used to create up to 32 different types of U-frames.

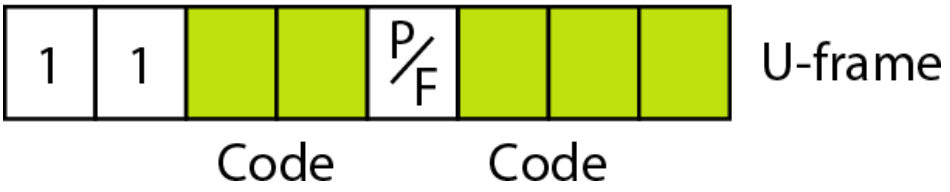


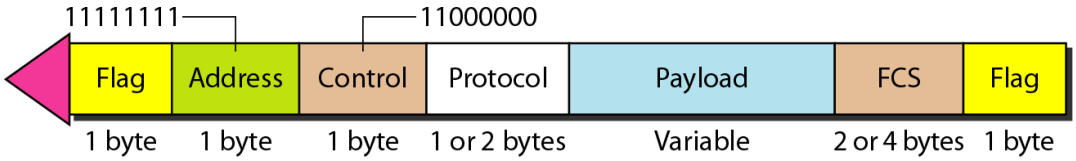
Table 11.1 U-frame control command and response

Code	*Command*	*Response*	*Meaning*
00 001	SNRM		Set normal response mode
11 011	SNRME		Set normal response mode, extended
11 100	SABM	**DM**	Set asynchronous balanced mode or **disconnect mode**
11 110	SABME		Set asynchronous balanced mode, extended
00 000	UI	**UI**	Unnumbered information
00 110		**UA**	**Unnumbered acknowledgment**
00 010	DISC	**RD**	Disconnect or **request disconnect**
10 000	SIM	**RIM**	Set initialization mode or **request information mode**
00 100	UP		Unnumbered poll
11 001	RSET		Reset
11 101	XID	**XID**	Exchange ID
10 001	FRMR	**FRMR**	Frame reject

IV. POINT-TO-POINT PROTOCOL

1) Although HDLC is a general protocol that can be used for both point-to-point and multipoint configurations, one of the most common protocols for point-to-point access is the Point-to-Point Protocol (PPP). PPP is a byte-oriented protocol.
2) Millions of Internet users who need to connect their home computers
3) to the server of an Internet service provider use PPP. PPP provides several services:
4) PPP defines the format of the frame to be exchanged between devices.
5) PPP defines how two devices can negotiate the establishment of the link and the exchange of data.
6) PPP defines how network layer data are encapsulated in the data link frame.
7) PPP defines how two devices can authenticate each other.
8) Services not provided by PPP :
   i. PPP does not provide flow control.
   ii. PPP has a very simple mechanism for error control. A CRC field is used to detect errors. If the frame is corrupted, it is silently discarded

PPP frame format



1. Flag :

A PPP frame starts and ends with a I-byte flag with the bit pattern 01111110. Although this pattern is the same as that used in HDLC, there is a big difference. PPP is a byte-oriented protocol; HDLC is a bit-oriented protocol. The flag is treated as a byte.

2. Address :

The address field in this protocol is a constant value and set to 11111111 (broadcast address). During negotiation, the two parties may agree to omit this byte.

3. Control:

This field is set to the constant value 11000000 (imitating unnumbered frames in HDLC). PPP does not provide any flow control. Error control is also limited to error detection. This means that this field is not needed at all, and again, the two parties can agree, during negotiation, to omit this byte.

4. Protocol:

The protocol field defines what is being carried in the data field: either user data or other information. This field is by default 2 bytes long, but the two parties can agree to use only I byte.

5. Payload field:

This field carries either the user data or other information. The data field is a sequence of bytes with the default of a maximum of 1500 bytes.

6. FCS

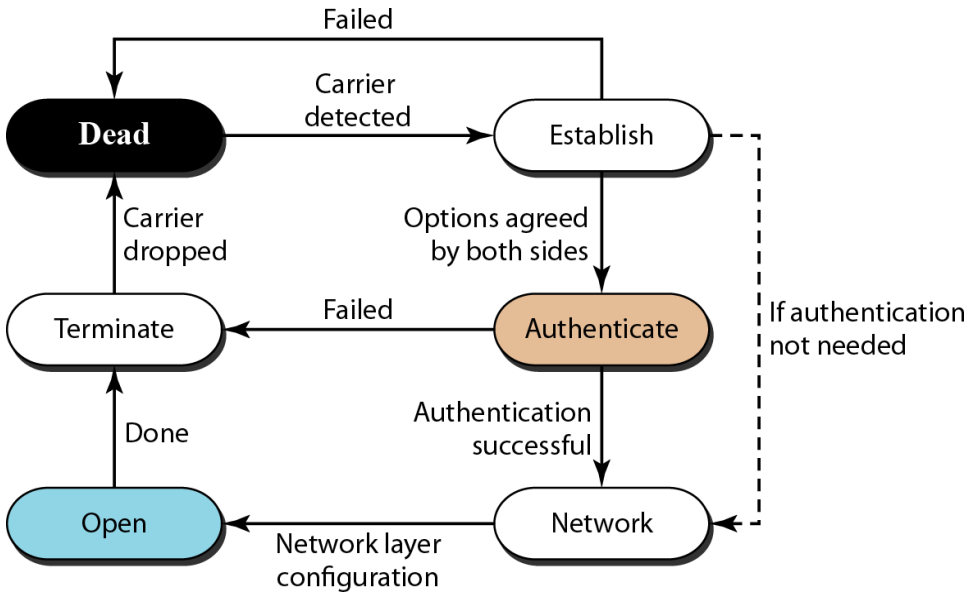
The frame check sequence (FCS) is simply a 2-byte or 4-byte standard CRC

PPP is a byte-oriented protocol using byte stuffing with the escape byte 01111101

8. PPP Transition Phase

PPP connection generally goes through different phases that can be seen in Transition Phase Diagram as shown below :

Figure 11.33 Transition phases



1.	**Dead**	In this phase, link basically starts and stops. Carrier Detection is an event that is used to indicate that physical layer is ready, and now the PPP will proceed towards establishment phase. Disconnection from modem line must bring back the line or connection to this phase
2.	**Establish**	When one of the nodes starts the communication, the connection goes into this phase. In this phase, options are negotiated between the two parties. If the negotiation is successful, the system goes to the authentication phase (if authentication is required) or directly to the networking phase.
3.	**Authenticate**	The authentication phase is optional; the two nodes may decide, during the establishment phase, not to skip this phase. However, if they decide to proceed with authentication, they send several authentication packets,. If the result is successful, the connection goes to the networking phase; otherwise, it goes to the termination phase
4.	**Network**	In the network phase, negotiation for the network layer protocols takes place. PPP specifies that two nodes establish a network layer agreement before data at the network layer can be exchanged. The reason is that PPP supports multiple protocols at the network layer. If a node is running multiple protocols simultaneously at the network layer, the receiving node needs to know which protocol will receive the data.
5.	**Open**	In the open phase, data transfer takes place. The connection remains in this phase until one of the endpoints wants to terminate the connection.
6.	**Terminate.**	Connection can be terminated at any point of time as per the request of either of the endpoints. LCP is basically required to close or terminate link through the exchange of terminate packets