DATA LINK LAYER

1. Need of data link layer - (functions)
   (1) Access to media
   (2) Provide reliable transfer of data across media
   (3) Transmit frames node to node based on station address
   (4) Provide service interface to the network layer

2. Design Issues -
   (1) Service provided to network layer
   (2) Framing
   (3) Error Control
   (4) Flow Control

3. Services provided to network layer -
   (1) Unacknowledged connectionless service
   (2) Acknowledged connectionless service
   (3) Acknowledged connection-oriented service

4. Framing -
   Three forms of framing - byte count
   (1) Byte Delimit - 5 4 7 8 2
   (2) Flag byte with byte stuffing
   (3) Flag bit with bit stuffing

   Five consecutive incoming 1 bits followed by a 0 bit, it
   automatically deletes the 0 bit. Example: 0111110 \rightarrow 0111110 10

5. Flow Control -
   When sender is fast and receiver is slow. To prevent this jumble
   (1) Feedback based flow control
   (2) Rate based flow control
**Error Control**

Use of acknowledgment

A timer at sender and receiver end is introduced. Also keep numbers to the outgoing frames one maintained so that the receiver can distinguish retransmission from original.

**Data Link Layer Protocol**

- **Elementary and Sliding Window protocol**

  - Sender Sliding window: windows of frame that may be transmitted
  - Frame sequence number
  - Last frame transmitted
  - Window shrinks as frame are sent
  - Receiver Sliding window
  - Window before receiving frame 4
  - Window after receiving frame 4

- **One bit (Stop and wait ARQ) sliding window protocol**

  - Frame F<sub>0</sub>
  - A<sub>0</sub> F<sub>1</sub>
  - Frame A<sub>0</sub>
  - NA<sub>0</sub> F<sub>0</sub>
  - F<sub>1</sub> A<sub>1</sub>

  - Propagation delay: t<sub>prop</sub>
  - Processing delay: t<sub>proc</sub>
(2) Go back N ARQ - Sliding window protocol

(3) Selective Repeat ARQ

(4) Hybrid ARQ - Combination of above 3.

8. BERT - Bit Oriented protocols
   - SDLC (Synchronous Data Link Control) protocol
     No extra bit (start/stop). A defined bit pattern for preamble and postamble with the entire data frame. (E.g., \(1\) byte = \(8\) bit known as synchronizing bit). Uses Go-back-N, part of SNAP
   - BISYNCH (Binary Synchronous Control) protocol
     Support three codes - ASCII, EBCDIC, and traditional synchronous
     two-way alternate communication
   - LAP (Link Access Protocol)
     Used for packet-switched networks and ISDN. 946 used for framing and transmitting data across point to point link
     - LAPB (Bearer) - Updated version
     - HDLC (High-level data link control protocol)
       Uses on both point to point and multipoint data link...
Three types of stations are:
1. Primary Station → frame sent (command)
2. Secondary Station → frame sent (response)
3. Combined Station → both primary and secondary station features

Transfer Modes of HDLC:
1. Normal Response mode (NRM) → Only transmit data from secondary station to primary station in response.
2. Asynchronous Response mode (ARM) → Transmit data from secondary station without primary station's response but primary station is in control of connection.
3. Asynchronous Balanced mode (ABM) → Both primary and secondary station have equal status.

Frame Types:
1. Unnumbered frames (U-frames)
2. Information frames (I-frames)
3. Supervisory frame (S-frames) → used for error and flow control

Frame Format:
```
| FLAG | ADDRESS | CONTROL BTS | DATA | FCS | FLAG |
```

FCS → Frame Check Sequence

HDLC Operation: Initialization, Data Transfer, and Disconnect

Protocol Verification:
1. Finite state machine model:
   - States: Transitions, Initial State → (0, 0, 0)
   - x → sends frame, y → receives frame, z → state of channel.

   During normal operation, transitions 1, 2, 3, and 4 are repeated repeatedly. In each cycle, two packets are delivered, bringing the channel back to the initial state of trying to send a new frame.
<table>
<thead>
<tr>
<th>Transition</th>
<th>Who send?</th>
<th>Frame Accepted</th>
<th>Frame Emitted</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>—</td>
<td>(Frame host)</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>R</td>
<td>0</td>
<td>A</td>
</tr>
<tr>
<td>2</td>
<td>S</td>
<td>A</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>R</td>
<td>1</td>
<td>A</td>
</tr>
<tr>
<td>4</td>
<td>S</td>
<td>A</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>R</td>
<td>0</td>
<td>A</td>
</tr>
<tr>
<td>6</td>
<td>R</td>
<td>1</td>
<td>A</td>
</tr>
<tr>
<td>7</td>
<td>S</td>
<td>(time out)</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>S</td>
<td>(time out)</td>
<td>1</td>
</tr>
</tbody>
</table>

Sequence number 0.

During sending, line out system moves back to initial state (Transition 7). The loss of an ACK requires two transition 7 and 5 or 8 and 6 to repair the damage.

(2) PetriNet Models— (No composite state)

from basic elements: places, transitions, arcs, and tokens.

**Piggybacking** —

The technique of temporarily delaying outgoing acknowledgments so that they can be hooked onto the next outgoing data frame.