THE DATA LINK LAYER

UNIT 3

Topics to be covered:

- Data Link layer Design Issues
- Error Detection & correction
- Elementary Data Link Protocols
- Sliding Window Protocol
- Example of DLL Protocol : HDLC

Scanned by CamScanner

DATA LINK LAYER

- Algorithm for achieving:
  - Reliable, +
  - Efficient, communication of a whole units - frame (as opposed to bits - physical layer) between two machines.
- Two machines are connected through a conceptual communication channel that acts conceptually like a wire (eg telephone line, coaxial cable or wireless channel)
- Essential property of a channel that makes it "wire-like" connection is that the bits are delivered in the same order in which they are sent.
- For ideal channel (no distortion, unlimited bandwidth and no delay) the job of data link layer would be trivial.
- However the limitations on bandwidth, distortion and delay makes the job difficult.

Scanned by CamScanner

DATA LINK LAYER DESIGN ISSUES

- NETWORK LAYER SERVICES
- ERROR CONTROL
- FRAMING
- FLOW CONTROL

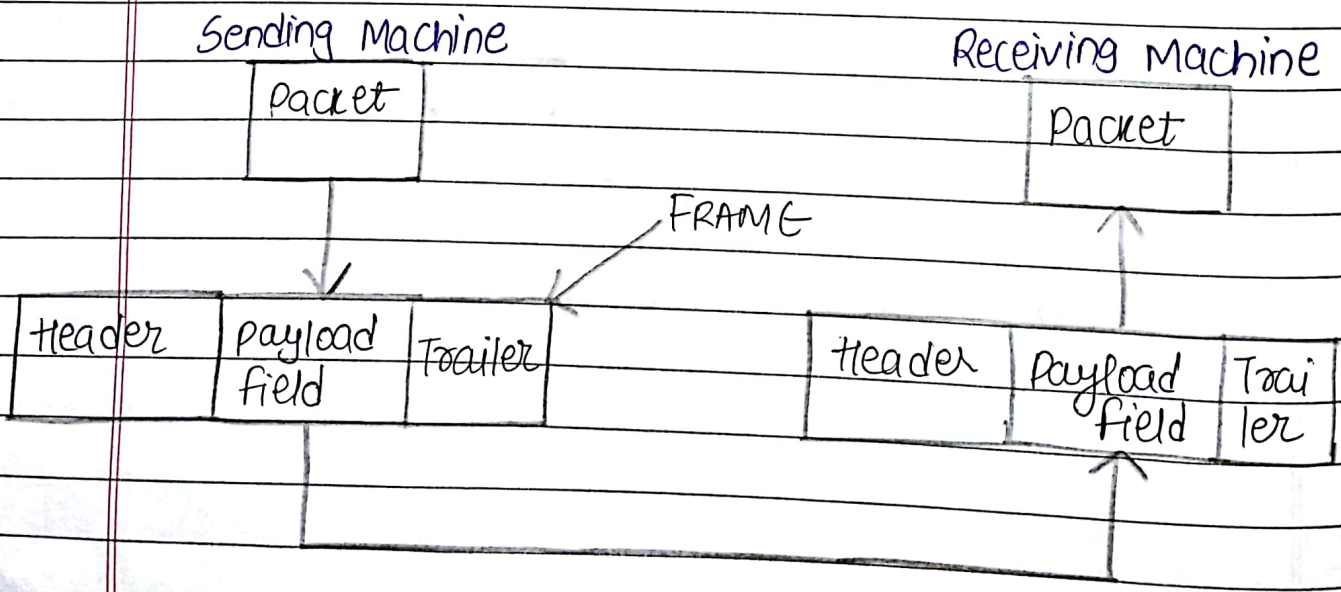
- Physical layer delivers bits of information to and from data link layer:

The functions of Data Link Layer:

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

- Data Link layer
  - Takes the packets from Physical Layer, and
  - Encapsulates them into frames
- Each frame has a
  - frame header - a field for holding packet, and
  - frame trailer
- Frame Management is what Data Link Layer Does.

Relationship between packets & frames



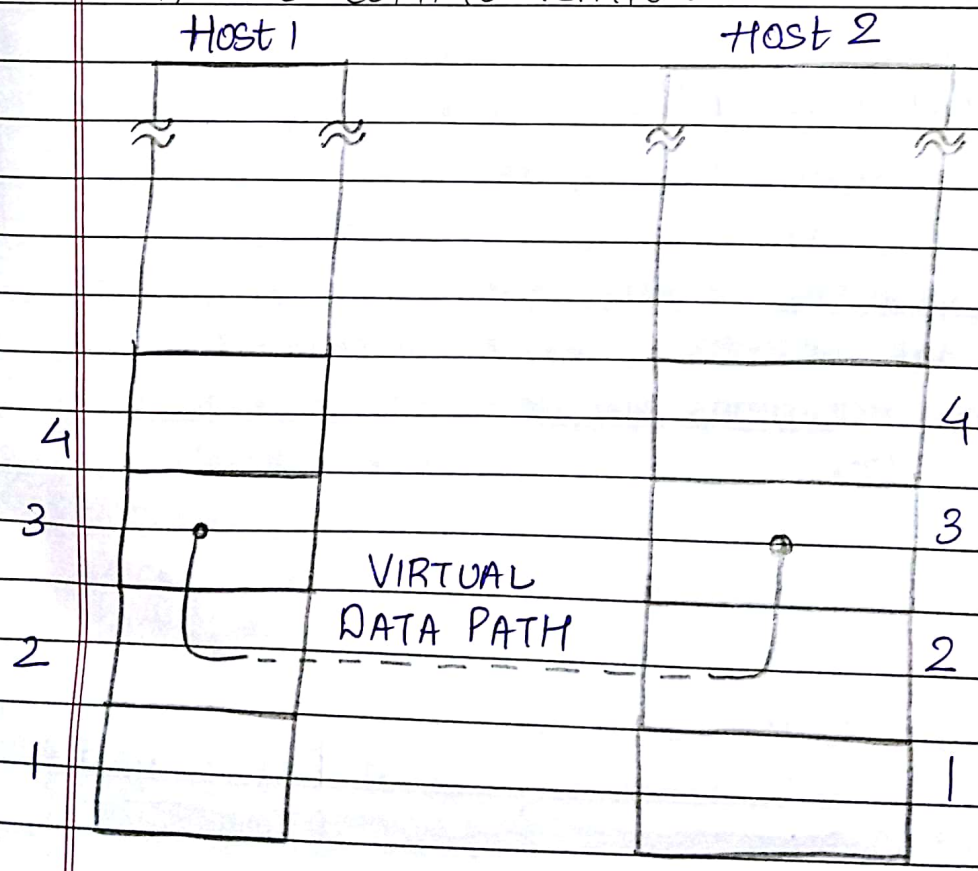
Scanned by CamScanner

SERVICE PROVIDED TO THE NETWORK LAYER

- Principal service function of the data link layer is to transfer the data from the network on the source machine to the network layer on the destination machine
  - Process in the network layer that hands some bits to the data link layer for transmission
  - Job of data link layer is to transmit the bits to the destination machine so they can be handed over to network layer.

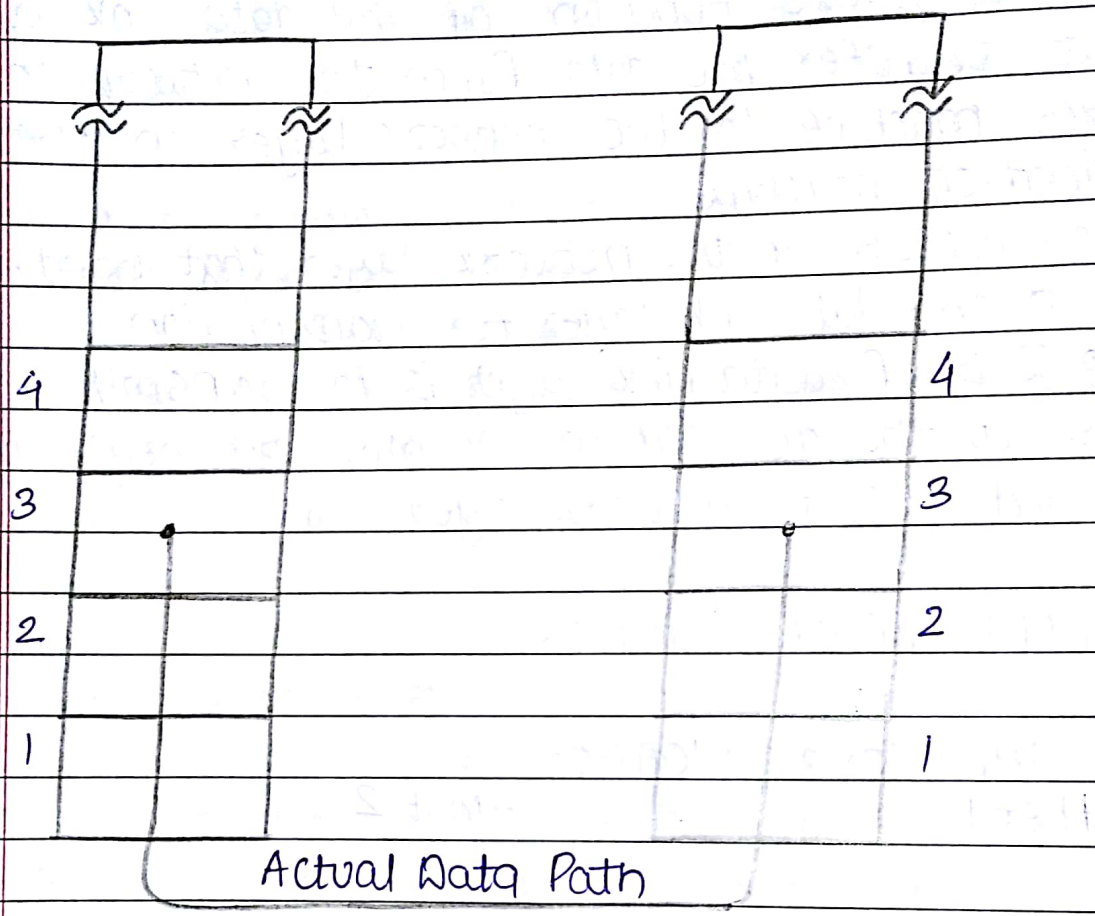
NETWORK LAYER SERVICES

a.) VIRTUAL COMMUNICATION



Scanned by CamScanner

b) Actual Communication



Scanned by CamScanner

POSSIBLE SERVICES OFFERED

1. UNACKNOWLEDGED CONNECTIONLESS SERVICE

- It consists of having the source machine send independent frames to the destination machine without having the destination machine acknowledge them.
- Example: Ethernet, Voice over IP, etc. in all the communication channel were real time operation is more important that quality of transmission.

2. ACKNOWLEDGED CONNECTIONLESS SERVICE

- Each frame send by the Data Link Layer is acknowledged and the sender knows if a specific frame has been received or lost
- Typically the protocol uses a specific time period that if has passed without getting acknowledgment it will re-send the frame
- This service is useful for communication when an unreliable channel is being utilized (e.g., 802.11 WiFi)
- Network layer does not know frame size of packets and other restriction of the data link layer. Hence it becomes neccessary for data link layer to have some mechanism to optimize the transmission.

Scanned by CamScanner

3. ACKNOWLEDGED CONNECTION ORIENTED SERVICE

- Source and destination establish a connection first.
- Each frame sent is numbered.
  - Data link layer gurantees that each frame sent is indeed received.
  - It gurantees that each frameme is received only once and that all frames are received in correct order.

- Examples:
  - Satellite channel communication
  - Long distance telephone communication etc

- THREE DISTINCT PHASES:
  - Connection is established by having both side intialize variables and counters needed to keep track of which frames have been received and which ones have not.
  - One or more frames are transmitted.
  - Finaly, the connection is released - freeing up the variables, buffers and other resources used to maintain connection.

Scanned by CamScanner

FRAMING

- To provide service to the network layer the data link layer must used the service provided to it by physical layer
- Stream of data bits provided to data link layer is not guaranteed to be without errors.
- Errors could be:
  - Number of received bits does not match number of transmitted bits (deletion or insertion)
  - Bit Value
  - It is upto data link layer whether to correct errors or not.
- Transmission of the data link layer starts with breaking up the bit stream
  - into discrete framed
  - Computation of checksum for each frame, and
  - Include the checksum into the frame before it is transmitted.
- Receiver computes its checksum error for a receiving frame and if it is different from the checksum that is being transmitted will have to deal with the error.

FRAMING METHODS

- BYTE COUNT
- FLAG BYTES WITH BYTE STUFFING
- FLAG BITS WITH BIT STUFFING
- PHYSICAL LAYER CODING VIOLATIONS

A3 Size My Drawing

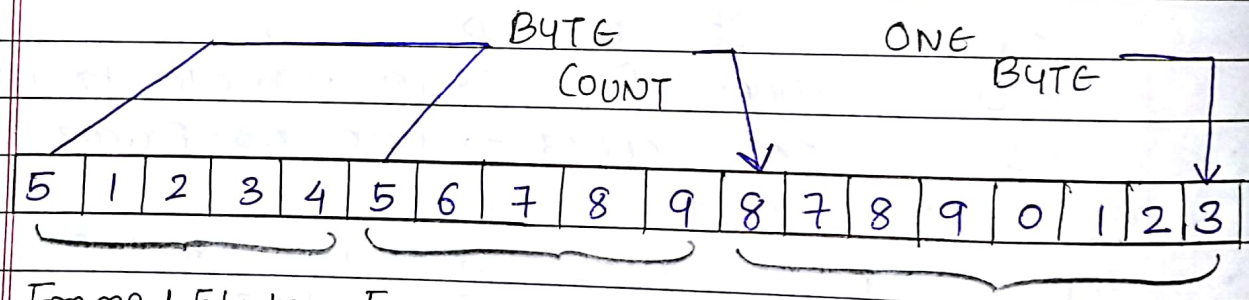
Scanned by CamScanner

BYTE COUNT FRAMING METHOD

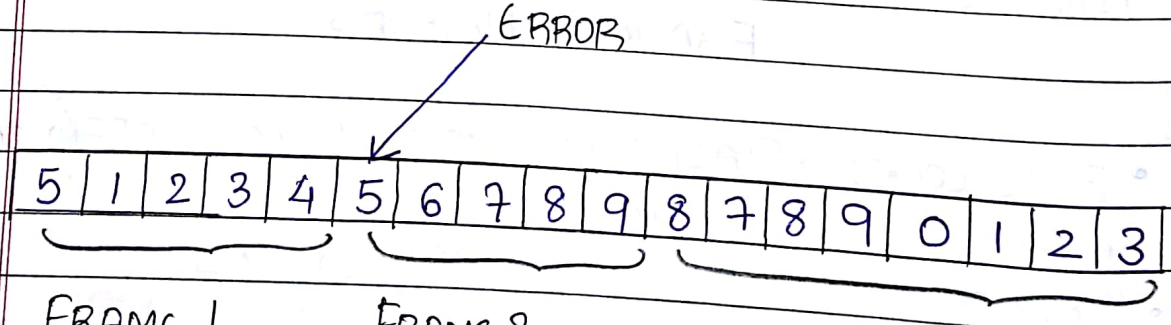
- It uses a field in the header to specify the number of bytes in frame
- Once the header information is being received it will be used to determine end of the frame.
- Trouble with this algorithm is that when the count is incorrectly received the destination will get out of synch with transmission.
  - Destination may be able to detect that the frame is in error but it does not have a means how to correct it.

A byte stream

(a) Without Errors



(b) With one Error

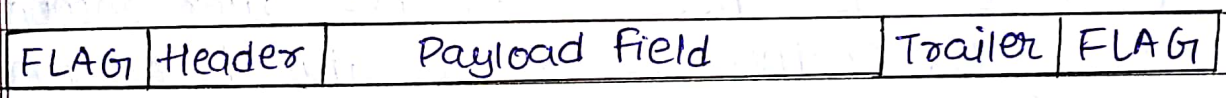


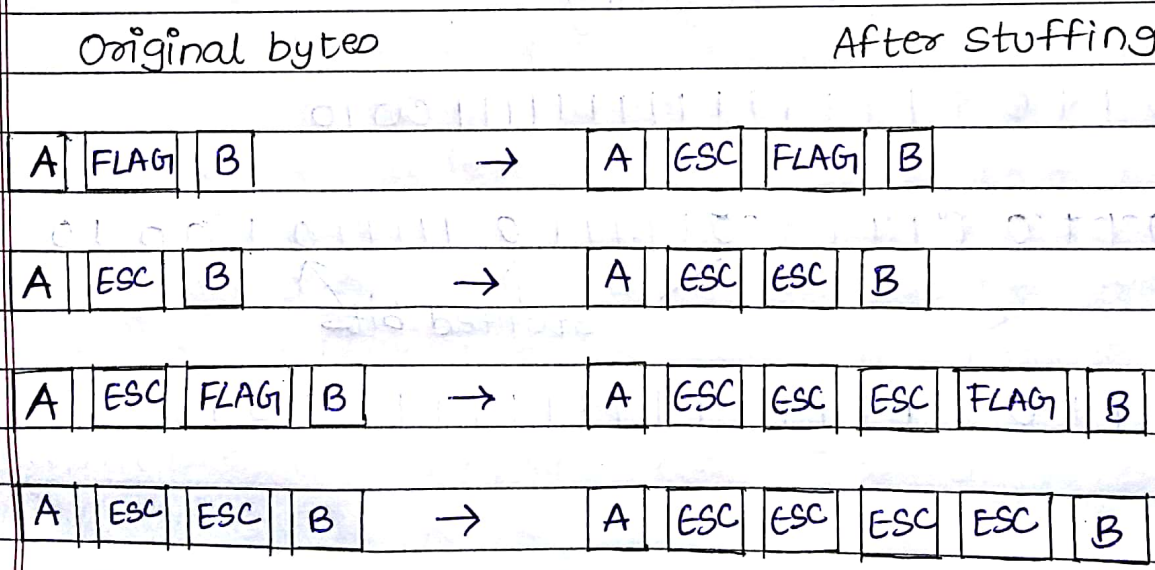
Scanned by CamScanner

FLAG BYTES WITH BYTE STUFFING FRAMING METHOD

- This method get around$ the boundary detection of the frame by having each appended by the frame start & frame end special bytes.
- If they are the same (beginning and ending byte in the frame) they are called flag byte.
- If the actual data contains a byte that is identical to the FLAG byte the convention that can be used is to have escape character inserted just before the "FLAG" character.

Diagram:





- A frame delimited by flag bytes
- Four examples of byte sequences before & after byte stuffing.

Scanned by CamScanner

FLAG BITS WITH BIT STUFFING FRAMING METHOD

- This method achieves the same thing as Byte Stuffing method by using Bits (1) instead of Bytes (8 bits)
- It was developed for High-Level Data Link Control (HDLC) protocol
- Each frame begins and ends with a special bit pattern: 01111110 or 0x7E ∠. 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 USB uses bit stuffing.

a) 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0

b) 0 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 0 0 1 0

Stuffed bits

c) 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0

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.

Scanned by CamScanner

Date
Page 11

PHYSICAL LAYER CODING VIOLATIONS

- Many data link protocols use a combination of presented methods for safety.
- For example in ethernet and 802.11 each frame begin with wel-defined pattern called a preamble
- Preamble is typicaly 72 bits long
- It is then followed by a length field.

ERROR CONTROL

- After solving the marking of the frame with start & end the data link layer has to handle eventual errors in transmission or detection.
  - Ensuring that all frames are delivered to the network layer at the destination in proper order.
- Unacknowledged connectionless service : it is Ok for the sender to o/p frames regardless of its reception
- Reliable connection-oriented service : it is NOT Ok
- Reliable connection-oriented service usualy will provide a sender with some feedback about what is happening at the other end of the line

Scanned by CamScanner

- Receiver Sends Back Special Control Frames
- If the Sender Received +ve Acknowledgment it will know that the frames has arrived safely
- Timer and Frame Sequence Number for the sender is Neccessary to handle the case when there is no response from the receiver

FLOW CONTROL

- Important Design Issue for the cases when the sender is running on a fast powerful computer and receiver is running on a slow low-end machine
- Two approaches:

1.] Feedback-based Flow Control
- Receiver sends back information to the sender giving it permission to send more data, or
- Telling sender how receiver is doing

2.] Rate-based Flow Control
- Built in mechanism that limits the rate at which sender may transmit data, without the need for feedback from the receiver.

Scanned by CamScanner

classmate
Date
Page 13

ERROR DETECTION AND CORRECTION

Two basic strategies to deal with errors:

1. Include enough redundant information to enable the receiver to deduce what the transmitted data must have been. Error ~~Detecting~~ Correcting Codes
2. Include only enough redundancy to allow the receiver to deduce that an error has occured (but not which error) Error Detecting Codes.

- Error codes are examined in Link Layer because this is the first place that we run up against the problem of reliability transmitting group of bits.
- Codes are reused because reliability is an overall concern
- The error correcting codes are also seen in the physical layer for noise channels.
- Commonly they are used in Link, network & transport layer.

Scanned by CamScanner

ERROR DETECTION & CORRECTION CODE

- All the codes presented in previous methods add redundancy to the information that is being sent.
- A frame consists of
  - m data bits (message) and
  - r redudant bits (check)
- Block Code - the r check bits are computed solely as function of m data bits with which they are associated.
- Systemic Code - the m data bits are send directly along with the check bits.
- Linear Code - the r check bits are computed as a linear function of the m data bits.

n - total length of a block (i.e, n=m+r)
(n,m) - code

n - bit codeword containing n bits

m/n - Code rate (range 1/2 for noisy channel & close to 1 for high-quality channel)

Example
Transmitted : 10001001
Received : 10110001
XOR operation gives no. of bits That are diff

XOR : 00111000

Inclusive of all taxes Official Use

Scanned by CamScanner

ERROR DETECTING CODES

Linear, Systematic and block codes

1. Parity
   - Single bit
   - Multiple bit
2. Checksums
3. Cycle Redundancy Checks (CRCs)

PARITY

1] SIMPLE PARITY CHECK

Blocks of data from the source are subjected to a check bit or parity bit generator form, where a parity of:

- 1 is added to the block if it contains odd number of 1's, and
- 0 is added if it contains even number of 1's
- This scheme makes the total number of 1's even, that is why it is called even parity checking.

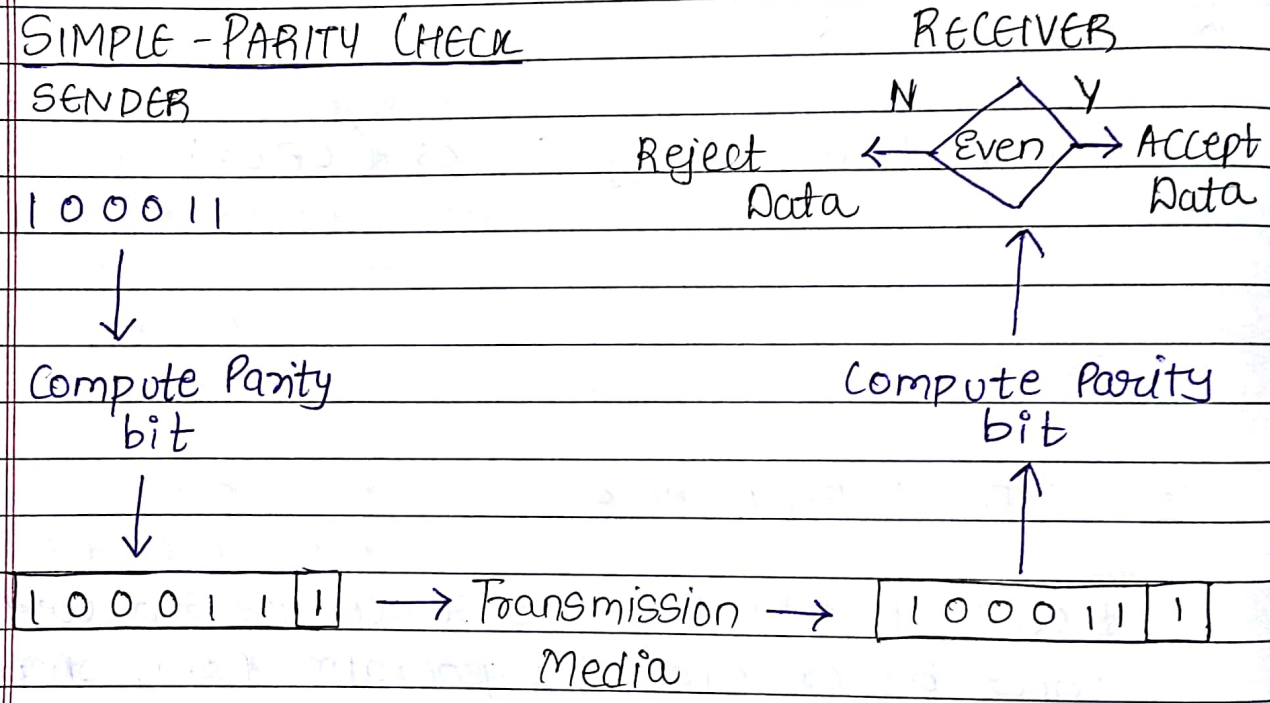
Scanned by CamScanner

2.] TWO-DIMENSIONAL PARITY CHECK

- Parity check bits are calculated for each row an columns, then both are sent along with the data

DIAGRAMATICALLY

SIMPLE-PARITY CHECK



TWO-DIMENSIONAL PARITY CHECK

original Data

10011001	11100010	00100100	10000100

	10011001	0
	11100010	0
	00100100	0
	10000100	0
Column Parities →	11011011	0

Scanned by CamScanner

Data to be sent

10011010	11100010	001001000	10001000	110110110

CHECK SUM

- In checksum error detection scheme, the data is divided into k segments each of m bits
- In the sender's end the segments are added using 1's complement airthmetic to get the sum. The sum is complemented to get the checksum
- The checksum segment is sent along with the data segments
- At the receiver's end, all received segments are added using 1's complement airthmetic to get the sum. The sum is complemented. If the result is zero, the received data is accepted; otherwise discarded.

Scanned by CamScanner

Original Data

10011001	11100010	00100100	10000100

k = 4 , m = 8

Sender

```
1     1 0 0 1 1 0 0 1
2     1 1 1 0 0 0 1 0
    1 0 1 1 1 1 0 1 1
      →             1
    ---------------------
      0 1 1 1 1 1 0 0
3     0 0 1 0 0 1 0 0
    ---------------------
      1 0 1 0 0 0 0 0
4     1 0 0 0 0 1 0 0
    ---------------------
    1 0 0 1 0 0 1 0 0
      →             1
    ---------------------
Sum: 0 0 1 0 0 1 0 1
```

↓ 1's complement

Checksum: 1 1 0 1 1 0 1 0

Now,

Sum + Checksum = 1 1 1 1 1 1 1 1

↓ 1's complement

00000000

Conclusion : Accept Data

Scanned by CamScanner

CYCLIC REDUNDANCY CHECK [CRC]

- Unlike checksum scheme, which is based on addition, CRC is based on binary division
- In CRC, a sequence of redundant bits, called cyclic redundancy check bits, are appended at the end of data unit so that the resulting data unit becomes exactly divisible by a second, predetermined binary number.
- At the destination, the incoming data units is divided by the same number. If at this step there is no remainder, the data unit is assumed to be correct and it is therefore accepted.
- A remainder indicates that the data unit has been damaged in transit and therefore must be rejected

Original message - 1010000 M(x)
Code / key - 1001 G(x)

Sender

```
        ↓
1001 | 10100000000
       1001
       ----
       xx1100
         1001
         ----
         x1010
          1001
          ----
          xx1100
            1001
            ----
            x1010
             1001
             ----
             xx11
```

10100000000 + 11 = 10100000011

Receiver

↓

```
1001 | 10100011
       1001
       ----
       xx1100
         1001
         ----
         x1010
          1001
          ----
          xx1101
            1001
            ----
            x1001
             1001
             ----
             0000  ← Zero means data has been accepted
```

Scanned by CamScanner

FLOW CONTROL AND ERROR CONTROL

- FLOW CONTROL

- A set of procedures that tells the sender how much data can be sent before waiting for acknowledgement.

- ERROR CONTROL

- Included both error detection & correction
- Allows receiver to inform sender of lost or duplicate frames.
- Mostly based on Automatic Repeat Request (ARQ)

B PROTOCOLS

- The protocols are normally implemented in software by using one of the common programming languages
- To make our discussions language-free, we have written is pseudocode version of each protocol that concentrates mostly on the procedure instead of delving into the details of language rules

Scanned by CamScanner

PROTOCOLS

- FOR NOISELESS CHANNEL
  - SIMPLEST
  - STOP-and-WAIT
- FOR NOISY CHANNEL
  - Stop-and-Wait ARQ
  - Go-Back-N ARQ
  - Selective Repeat ARQ

NOISELESS CHANNEL

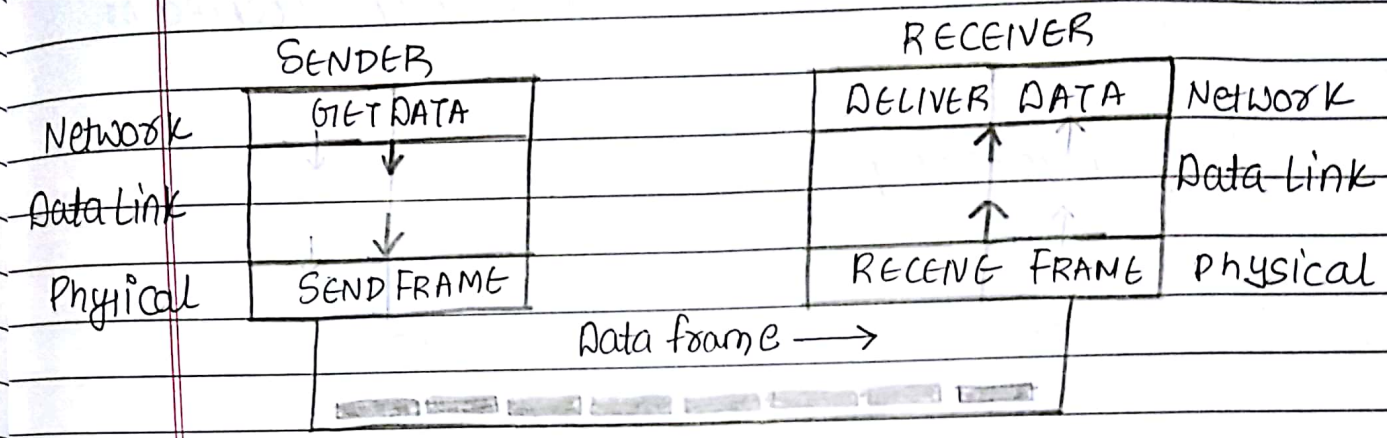
- Let us first assume we have an ideal channel in which no frames are lost, duplicated, or corrupted
  we introduce two types of protocol for this channel

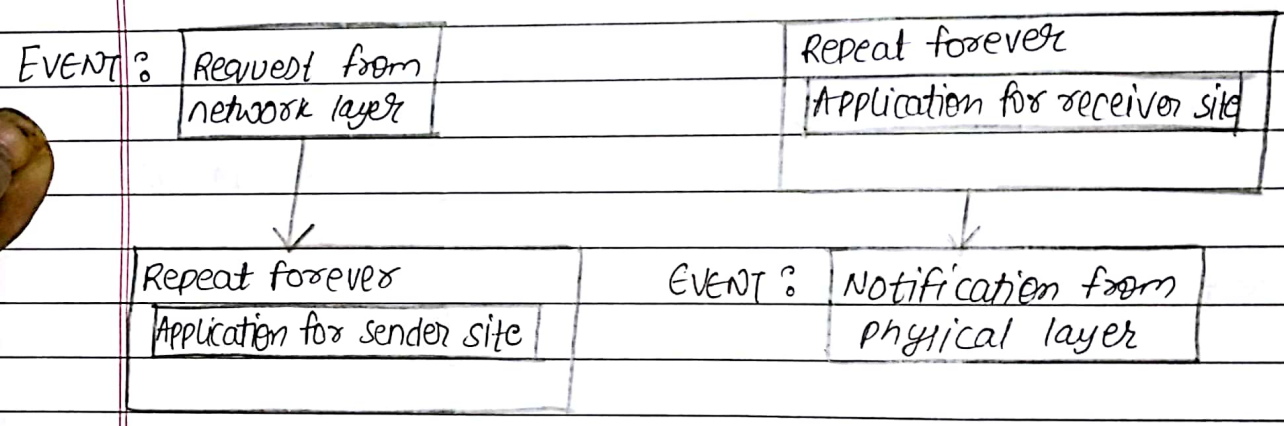
- SIMPLEST PROTOCOL
  - Noiseless channel
  - unlimited buffer and speed for the receiver

- STOP-and-WAIT PROTOCOL
  - Still noiseless channel
  - Receiver has limited buffer
    - Requires flow control
  - Sender sends 1 frame at a time and waits for acknowledgement.

Scanned by CamScanner

SIMPLEST PROTOCOL





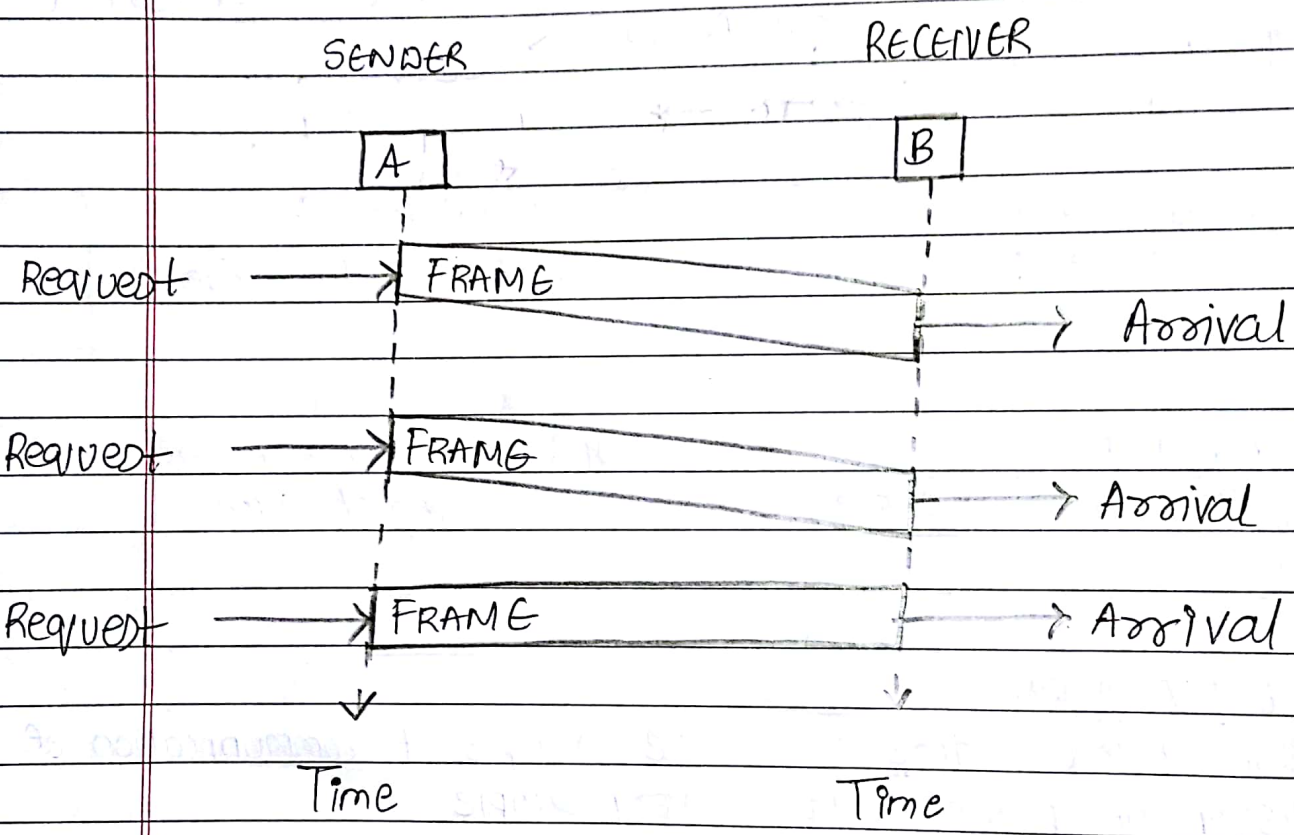
EXPLANATION:

- The below figure shows the example of communication of simplest protocol. It is very simple
- The sender sends a sequence of frames without even thinking about the receiver
- To send three frames, three events occur at the sender site and ~~there~~ three events at the receiver site
- The height of the box defines the transmission time and difference between first bit and the last bit in the frame.

Scanned by CamScanner

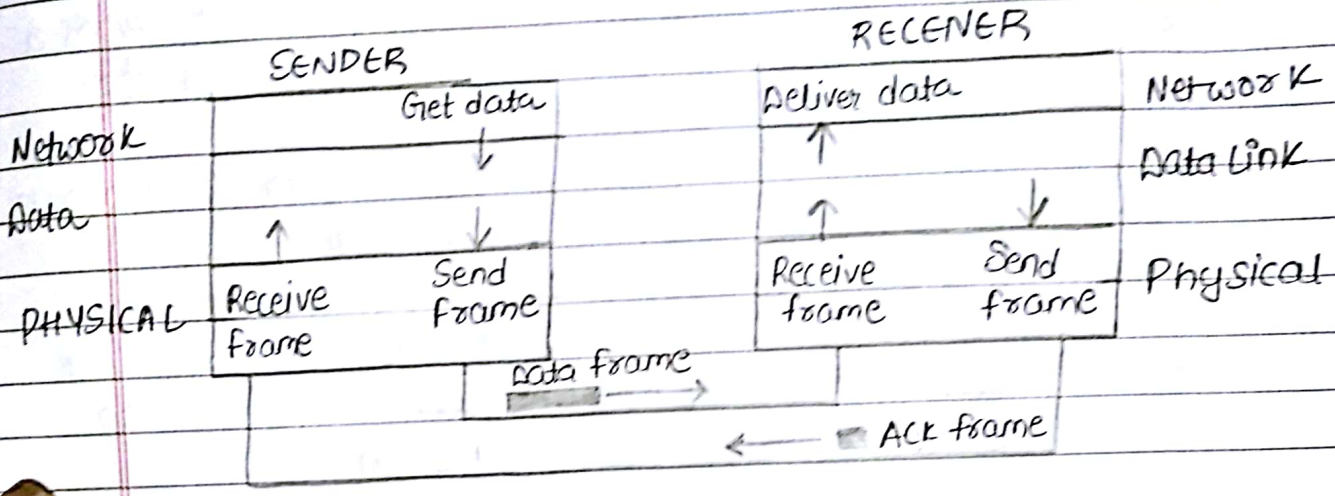
- There is no acknowledgment sent by the receiver to the sender for informing that whether the packet is sucessfully received or not.

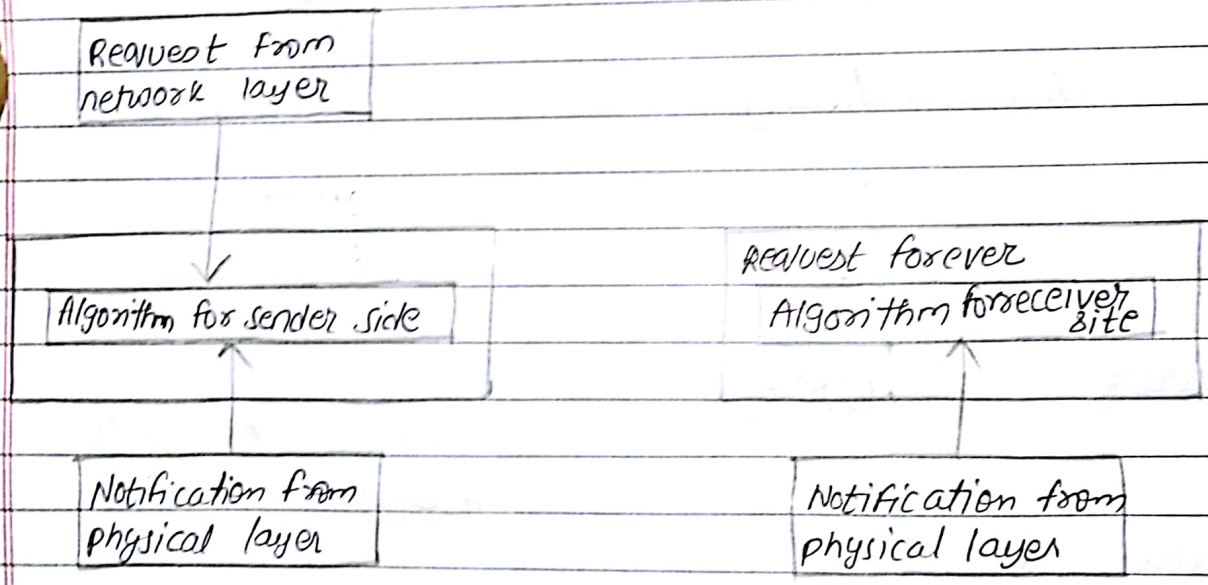
Diagramatically



Scanned by CamScanner

STOP-and-WAIT OVERVIEW:

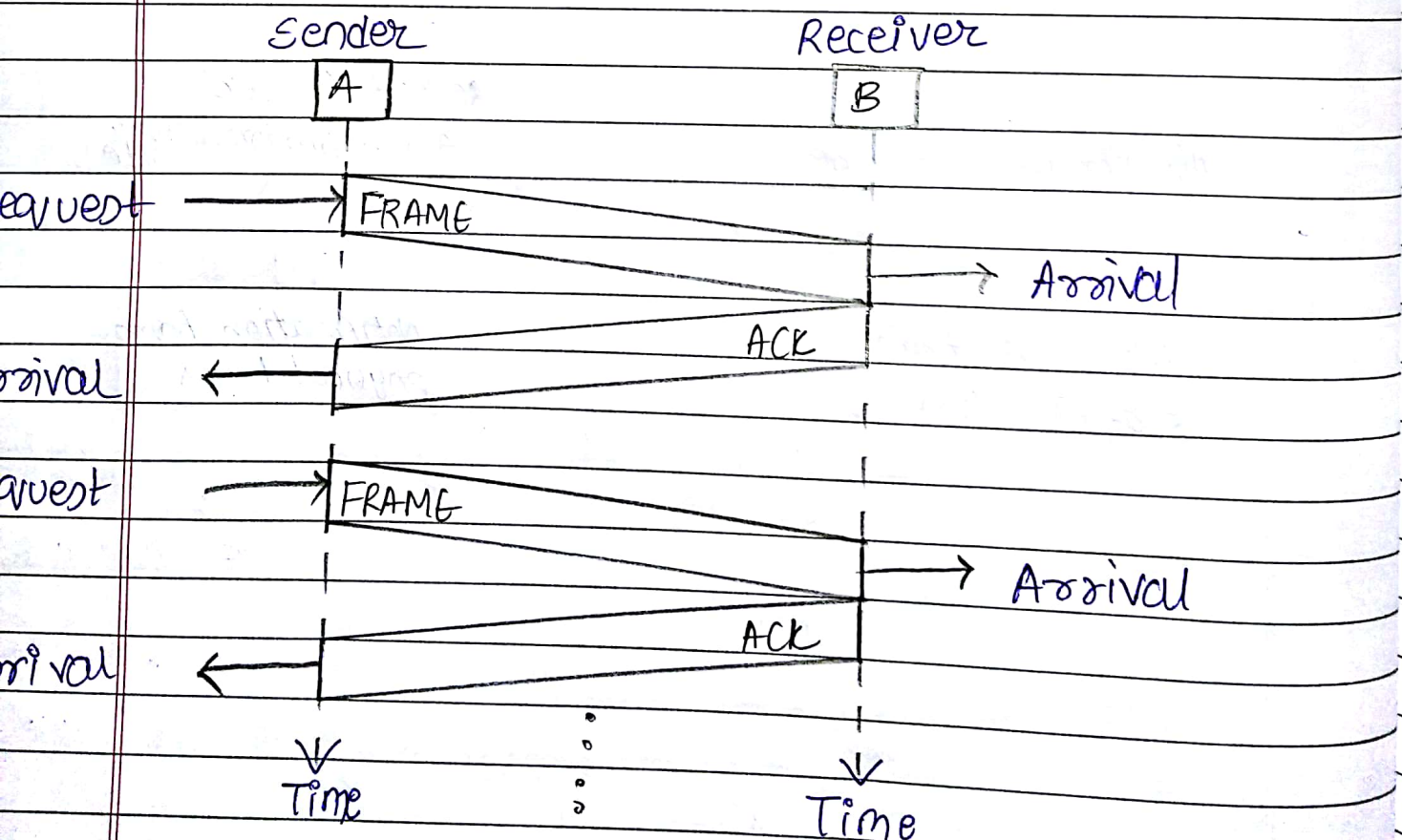




Scanned by CamScanner

- The below figure shows the communication using stop-and-wait protocol.
- In this method of flow control, sender sends a single frame to receiver & then waits for an acknowledgment
- The next frame is sent by sender only when acknowledgment of previous frame is received
- This process of sending a frame & waiting for an acknowledgment continues as long as the sender has data to send.
- To end up the transmission sender transmits end of transmission [EOT] frame

Diagramatically



Scanned by CamScanner

HIGH-LEVEL DATA LINK CONTROL [HDLC]

- High-level Data Link Control is a bit-oriented protocol for communication over point-to-point & multipoint links
- It implements the ARQ mechanisms which we

Configuration and Transfer Modes

- HDLC provides two common transfer modes that can be used in different configuration
  - Normal Response mode (NRM) and
  - Asynchronous Balanced Mode (ABM)

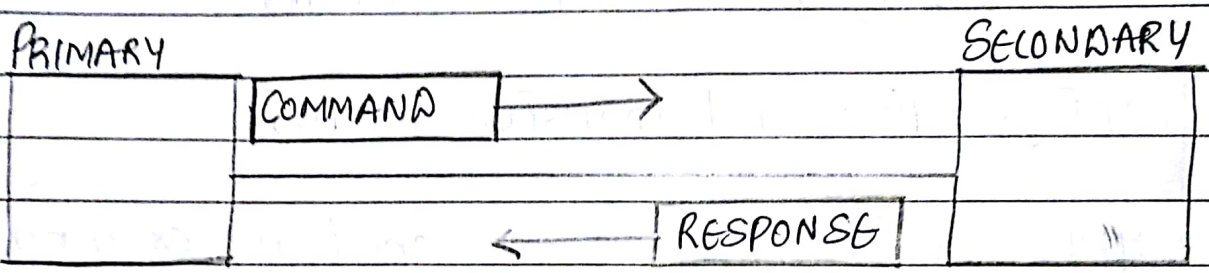
1. NORMAL RESPONSE MODE

- In Normal Response mode, 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 & multipoint a links, as shown in figure below.

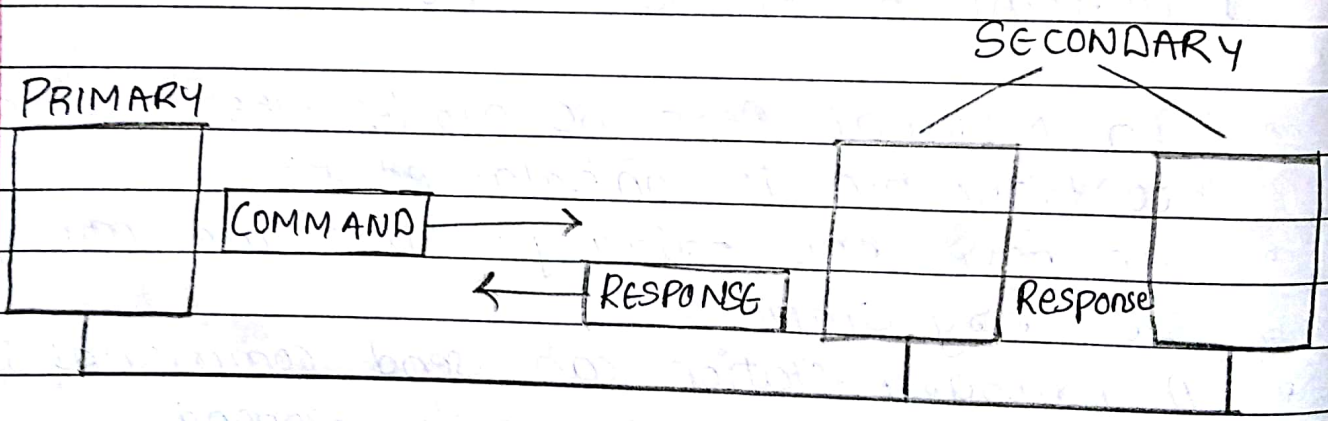
Scanned by CamScanner

NORMAL RESPONSE MODE [NRM]

1. POINT-TO-POINT



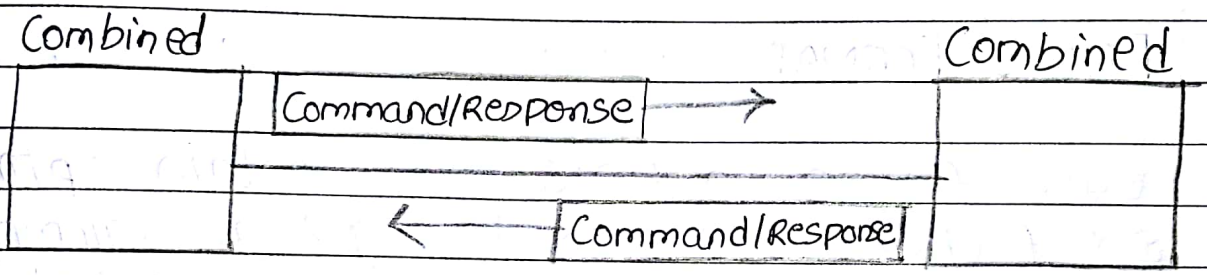
2. MULTIPOINT



Scanned by CamScanner

ASYNCHRONOUS BALANCED MODE

- In ABM, the configuration is balanced.
- The link is poin-to-point, and each station can function as primary and a secondary (acting as peers)
- This is the common mode today.



HDLC - Frames

- To provide the flexibility neccessary to support all the option possible in the modes and configuration described, HDLC defines three types of frames:
  - information frames (I frames),
  - supervisory frames (S-frames) and
  - unnumbered frames (U-frames).

Each type of frame serves as an envelope for the transmission of a different type of message

- Iframes are used to data-link user data and control information relating to user data (Piggybacking)

Scanned by CamScanner

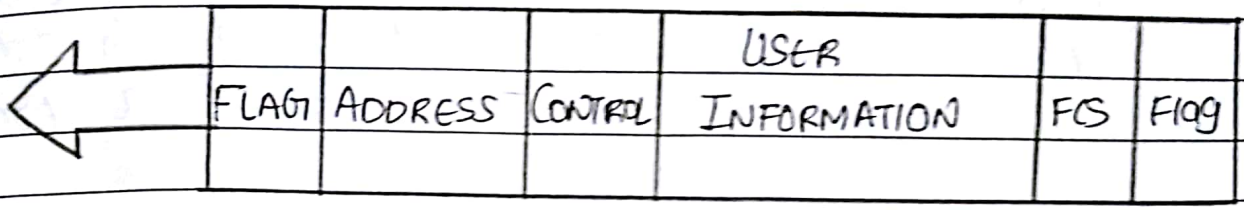
- S-frames are used only to transport control information
- U-frames are reserved for system management
- Information carried by U-frames is intended for managing the link itself

FRAME FORMAT

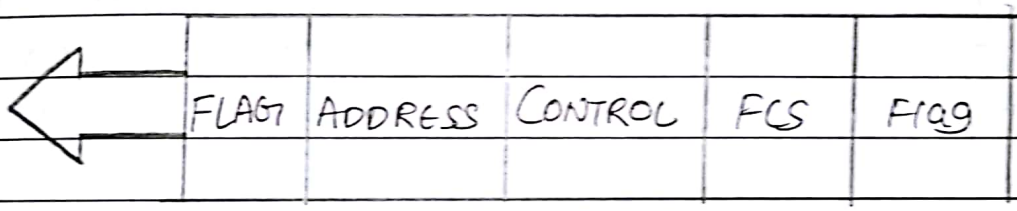
- Each frame in HDLC may contain upto six fields, as shown in fig: a beginning flag field, an address field, a control field, an information field, a frame check sequence, and an ending flag field
- In multiple-frame transmissions, the ending flag of one frame can serve as the beginning flag of the next frame.

Scanned by CamScanner

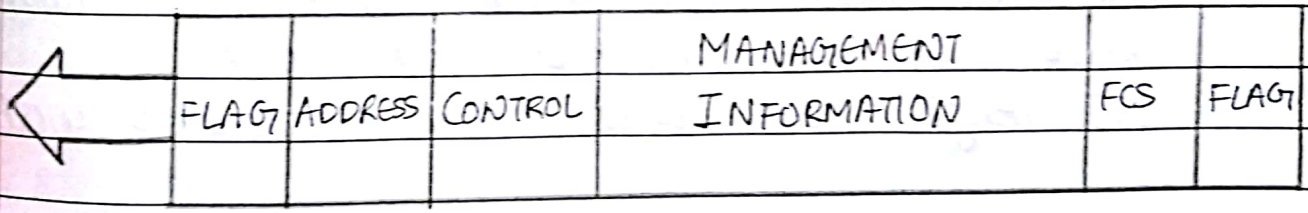
I-Frames



S-Frames



U-Frames



Scanned by CamScanner

FLAG FIELD : This field contains synchronization pattern 01111110, which identifies both the beginning and the end of a frame.

ADDRESS FIELD: This field contain the address of secondary station

If primary station created the frame, it contains a to address. If a secondary station creates a frame, it contains a from address.

CONTROL FIELD: It is of one or two byte and used for flow or error control

INFORMATION FIELD: The information field contains the user's data from the network layer or management information.
Its length can vary from network to another

FCS field: It is the ~~error~~ HDLC error detection field.

- It can contain either a 2-or-4 byte CRC
- The control field determines the type of frame and defines its functionality.

Scanned by CamScanner