COMPUTER NETWORKS
MODULE 2

MS. JINCY J FERNANADEZ

ASST PROF, CSE

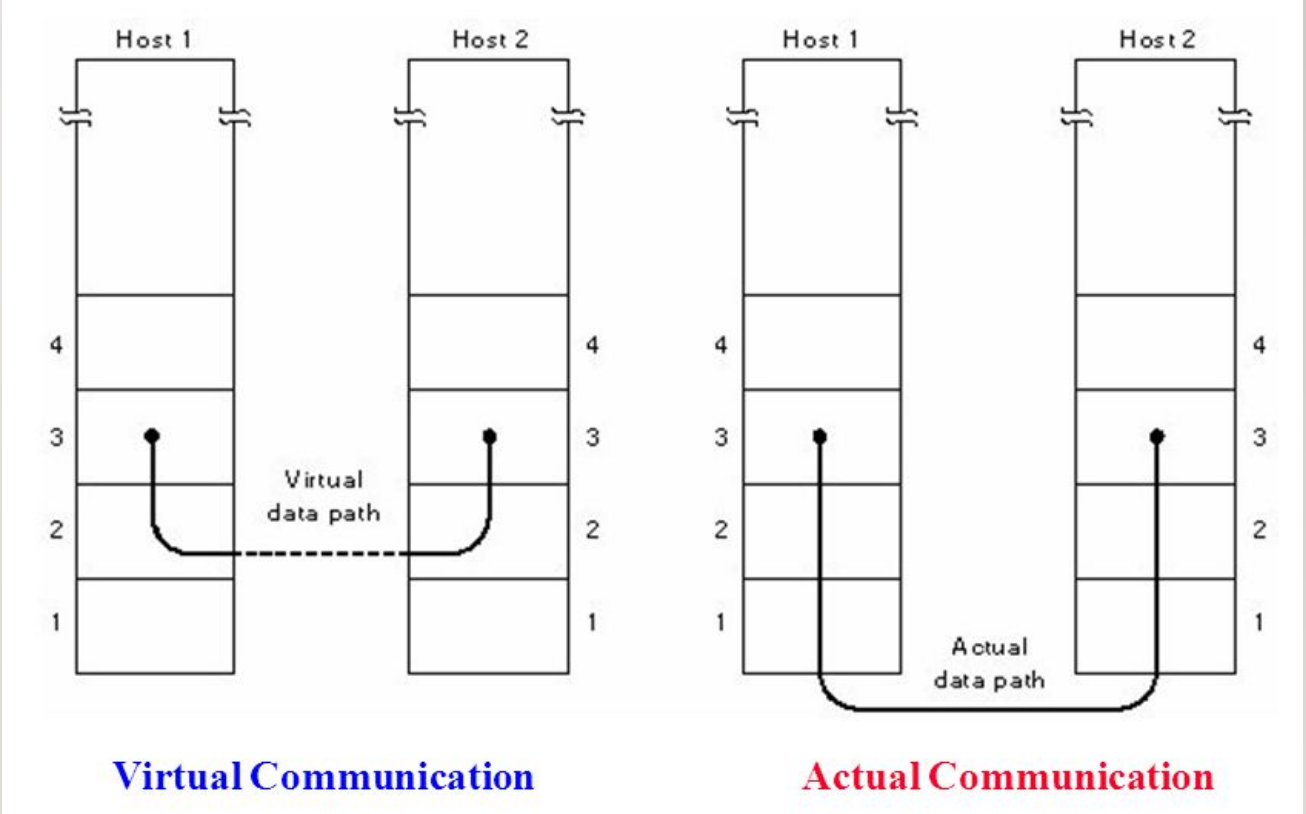
RSET

Data link layer- Design Issues

- Provide a well-defined service interface to the network layer.
- Deal with errors in transmission of frames.
- Regulate the flow of data so that slow receivers are not overwhelmed by fast senders.
- Framing of data.
- Detect and correct errors in frame data.

1. Service to Network layer

- Main service is transferring data from network layer on the source machine to network layer on the destination machine.



Service to Network layer

- Unacknowledged connectionless service
  - Source machine send independent frames to the destination machine without any acknowledgement policy.
  - No logical connection established or released.
  - If a frame is lost, no attempt to detect loss or recover it.
  - Used in LANs.

Service to Network layer

- Acknowledged connectionless service
  - No logical connections used here also.
  - Each frame sent is individually acknowledged.
  - In case of a time-out, data will be sent again.
  - Used in unreliable channels- wireless systems.

Service to Network layer

- Acknowledged connection-oriented service
  - Source and destination establish a connection before transferring data.
  - Release connection after transfer of data.
  - Each frame is numbered.
  - Guaranteed delivery of frames.
  - Ordered delivery of frames.
  - Used in WAN subnets containing routers connected by point to point leased telephone lines.

2. Error Control

- Need to ensure frames are delivered to the destination network layer correctly in proper order.
- Use acknowledgements: Positive or negative from receiver.
- Chance of lost messages. So, no positive or negative ACK will come from receiver.
- Use timers. Start a timer at sender when a frame is transmitted.
- Timer is set to expire after an interval long enough for frame to reach destination, get processed and ACK to propagate back to sender.
- If frame or ACK is lost. Then timer goes off. Then frame is retransmitted again.
- Chances of receiving same frame multiple times in time delays.
- Use sequence numbers for outgoing frames.

3. Flow Control

- Senders can be running on a fast computer and the receivers on a slow computer.
- Sender pumps out frames at a high rate and receiver gets overwhelmed.
- Even if transmission is error free destination will be unable to handle the frames and they get dropped off.
- Two solutions:
  - Feedback based flow control
  - Rate based flow control

Flow Control Techniques

- Feedback based control
- Receiver sends back information to the sender giving it permission to send more data.
- Depends on current status of receiver.

- Rate based control
- Protocol has a built-in mechanism to limit the rate at which sender may transmit data.
- No feedbacks given.

4. Framing

- Data link layer encapsulates data into frames for transmission.
- Each frame has a frame header, payload field for holding the packets, a frame trailer and flags.

Frames

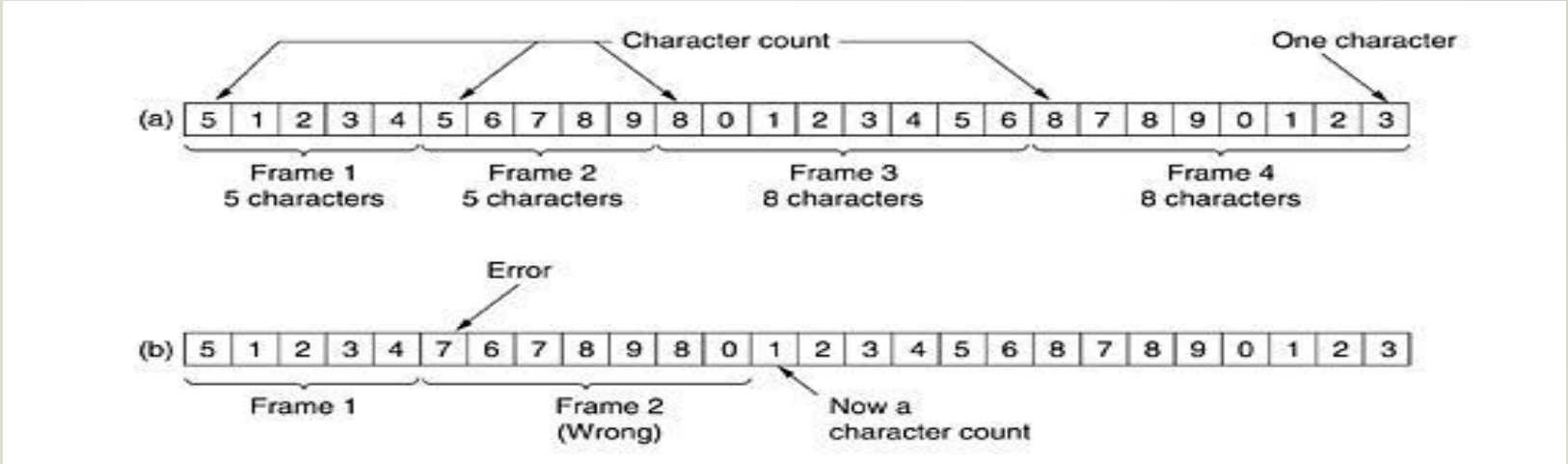
- A frame has the following parts:
  - Frame Header – It contains the source, and the destination addresses of the frame.
  - Payload field – It contains the message to be delivered.
  - Trailer – It contains the error detection and error correction bits.
  - Flag – It marks the beginning and end of the frame.
- Types of frame:
  - Fixed size frames
  - Variable size frames

Methods for Framing

- Byte count
- Flag bytes with byte stuffing
- Starting and ending flags with bit stuffing

1. Byte Count

- Use a field in the header to specify number of bytes in the frame.
- Count can get garbled by a transmission error: unable to locate the correct start of the next frame.
- Asking for retransmission to sender is of no use as well.
- Rarely used.



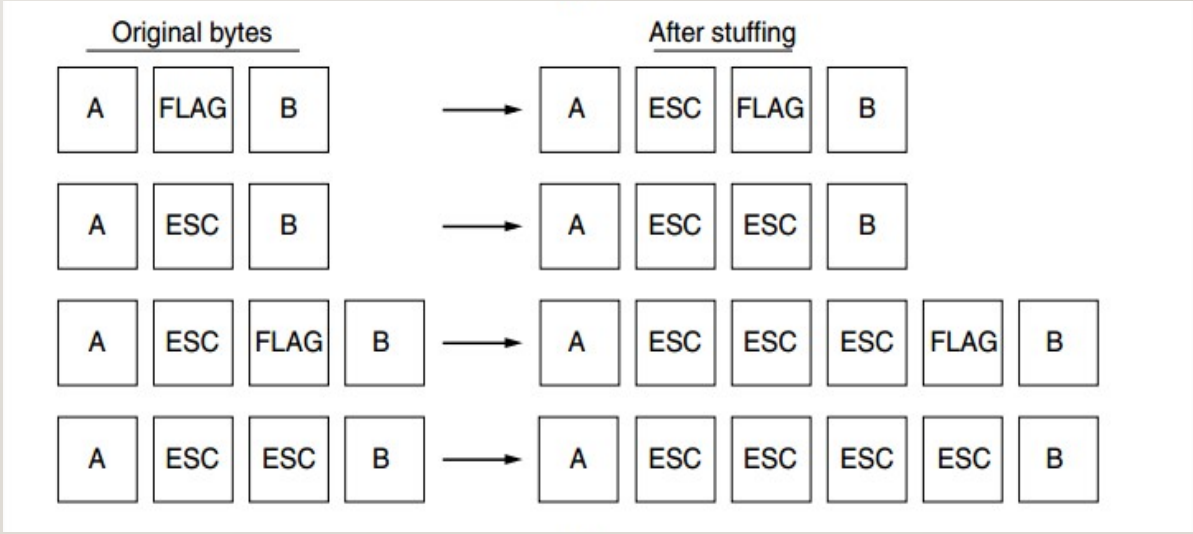
2. Flag Bytes

- Each frame starts and ends with special bytes- FLAG byte.

- Two consecutive flag bytes indicate end of one frame and start of next one.
- If receiver loses synchronization it can search for flag byte to get the end of the current frame and start of the next frame.
- Flag byte bit pattern may occur in the data.
- This interferes with framing.

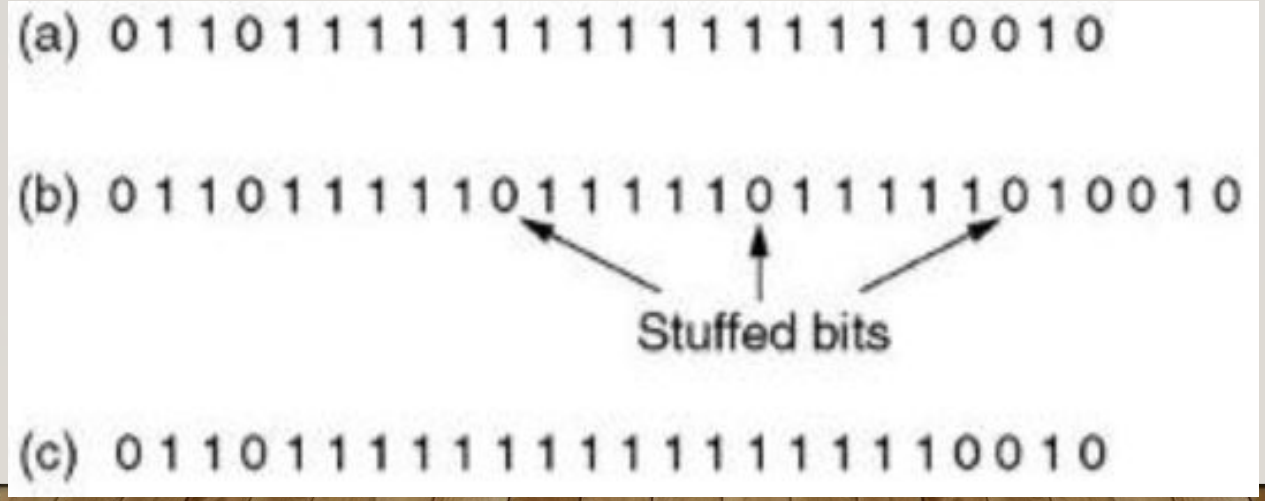
Flag Bytes with byte stuffing (Character oriented framing)

- Sender's data link layer inserts a special escape byte(ESC) just before each accidental flag byte in the data.
- Data link layer on receiving end removes the escape byte before the data is given to network layer.
- If escape byte occurs in the data??
- It is also stuffed with an extra escape byte.

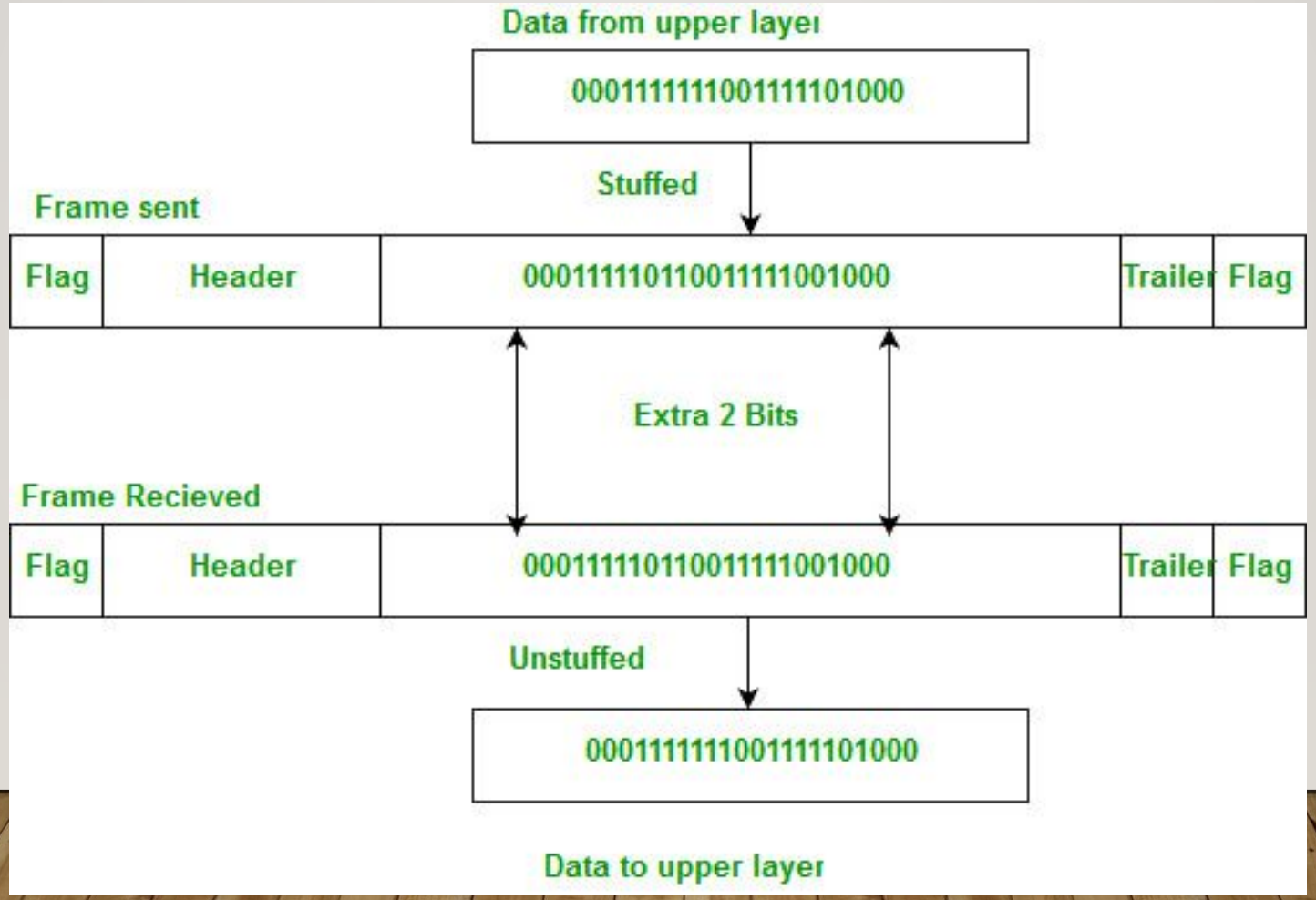


Flag Bytes with bit stuffing (Bit oriented framing)

- Each frame begins and ends with a special pattern 01111110 (flag pattern).
- Whenever there are 5 consecutive 1s in the data, sender automatically stuffs a 0 in the outgoing bit stream. It will be removed at destination data link layer.
- If user data contains the flag pattern 01111110 then it is transmitted as 011111010.



Flag Bytes with bit stuffing (Bit oriented framing)

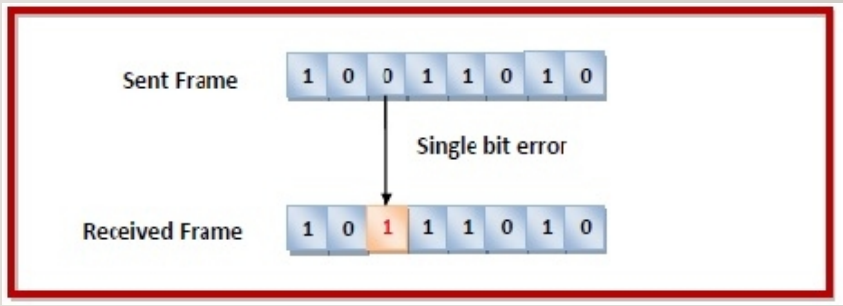


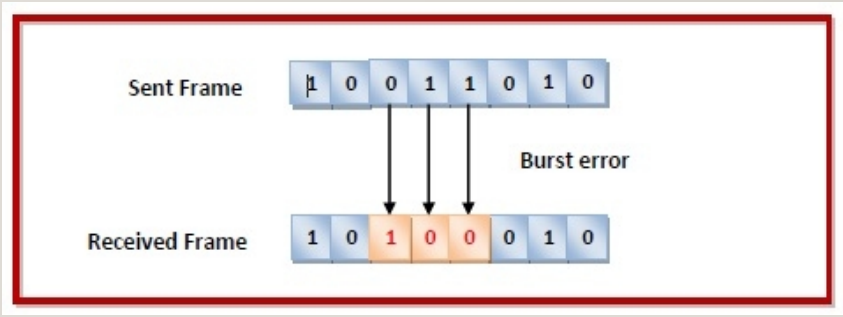
5. Error Detection and Correction

- Unpredictable changes during transfer of data due to interference.
- Noise usually occurs as bursts rather than independent, single bit errors.
- Detecting and correcting errors requires redundancy sending additional information along with the data.
- Redundancy is achieved through various coding schemes.

Error Detection and Correction

- There are two types of errors:
- Single bit error:
  - Only one bit has corrupted.
- Burst error:
  - More than one bits have corrupted.





Error Detection and Correction

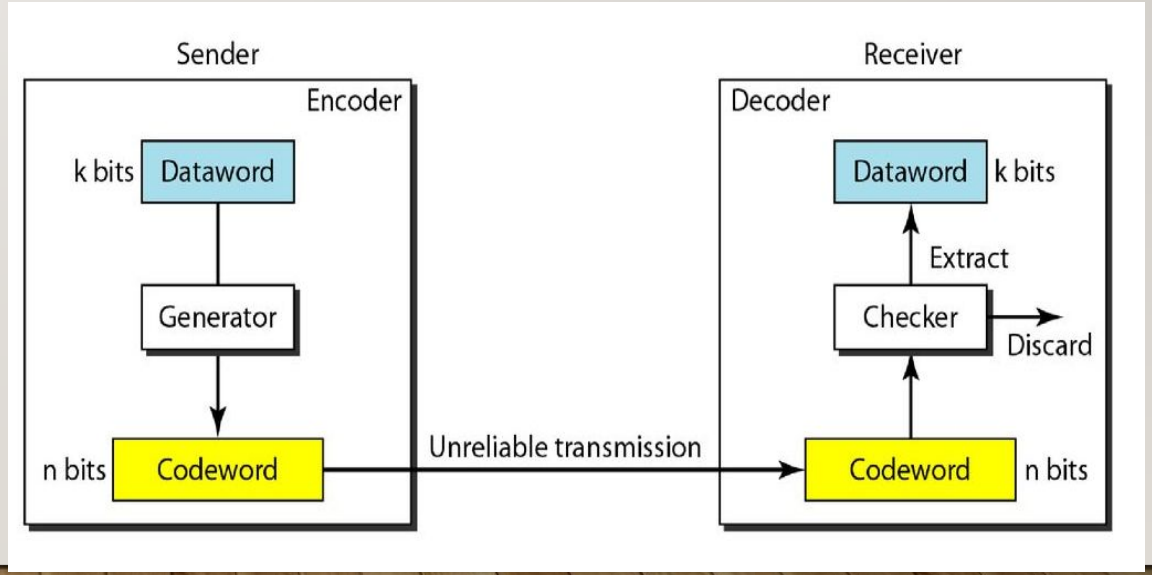
- There are two ways to control errors:
- Error Detecting Codes:
  - Check for whether error occurred or not.
  - The number of error bits and the type of error does not matter.
- Error Correcting Codes:
  - Need to know the exact number of bits that are corrupted and their location in the message.

Coding Scheme

- The sender adds redundant bits that creates a relationship between the redundant bits and the actual data. The receiver checks the relationships between the two sets of bits to detect errors.
- Important factors of coding scheme:
  - The ratio of redundant bits to the data bits
  - Robustness of the coding process
- Two types of Coding
  - Block Coding: information bits are immediately followed by parity bits.
  - Convolution Coding: information bits are not followed by parity bits instead spread along the sequence.

Coding Scheme

- Block Coding:
  - Divide the message into blocks, each of 'm' bits called data words.
  - Add 'r' redundant (check or parity) bits to each block (n=m+r). → codewords.
  - With 'm' bits, 2^m data words are possible.
  - One to one coding.
  - Code rate= fraction of the codeword that carries information that is not redundant.
  - Code rate= $\frac{m}{r}$



Coding Scheme

- Block Code:
  - Systematic Code:
    - ‘m’ data bits are sent directly, along with check bits.
  - Linear code:
    - ‘r’ check bits are computed as a linear function of the ‘m’ data bits.
    - XOR is a popular function.

Error Correcting Codes- Block codes

- Hamming Distance
  - Between two words is the number of distances between the corresponding bits.
  - The number of bit positions in which two codewords differ.
  - The Hamming distance d(000, 011) is 2

$$000 \oplus 011 \text{ is } 011 \text{ (two 1s)}$$

  - The Hamming distance d(10101, 11110) is 3

$$10101 \oplus 11110 \text{ is } 01011 \text{ (three 1s)}$$

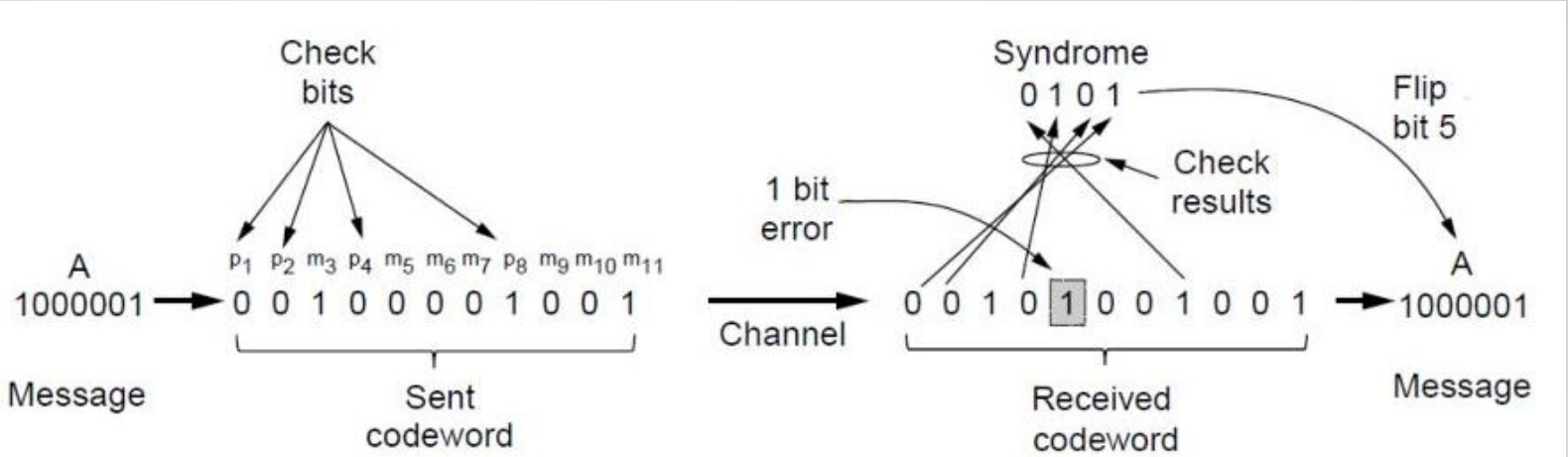
Error Correcting Codes- Block codes

- Hamming Distance
  - The bits of the codeword are numbered consecutively.
  - The bits that are powers of 2 (1,2,4,8,…) are check bits (parity bits) and rest are filled up with 'm' data bits.
  - (n,m) hamming code.

Parity Bit	Bit positions
1	1, 3, 5, 7, 9, 11, 13, 15 …
2	2, 3, 6, 7, 10, 11, 14, 15 …
4	4, 5, 6, 7, 12, 13, 14, 15 …
8	8, 9, 10, 11, 12, 13, 14, 15 …

Error Correcting Codes- Block codes

- Hamming Distance
  - The bits of the codeword are numbered consecutively.
  - The bits that are powers of 2 (1,2,4,8,…) are check bits and rest are filled up with ‘m’ data bits.
  - (n,r) hamming code.



Example of an (11, 7) Hamming code correcting a single-bit error.

Error Correcting Codes- Block codes

- If the 7-bit hamming word received by a receiver is 1011011. Assuming the even parity, state whether the received code word is correct or wrong. If wrong, locate the bit having error and do the correction.

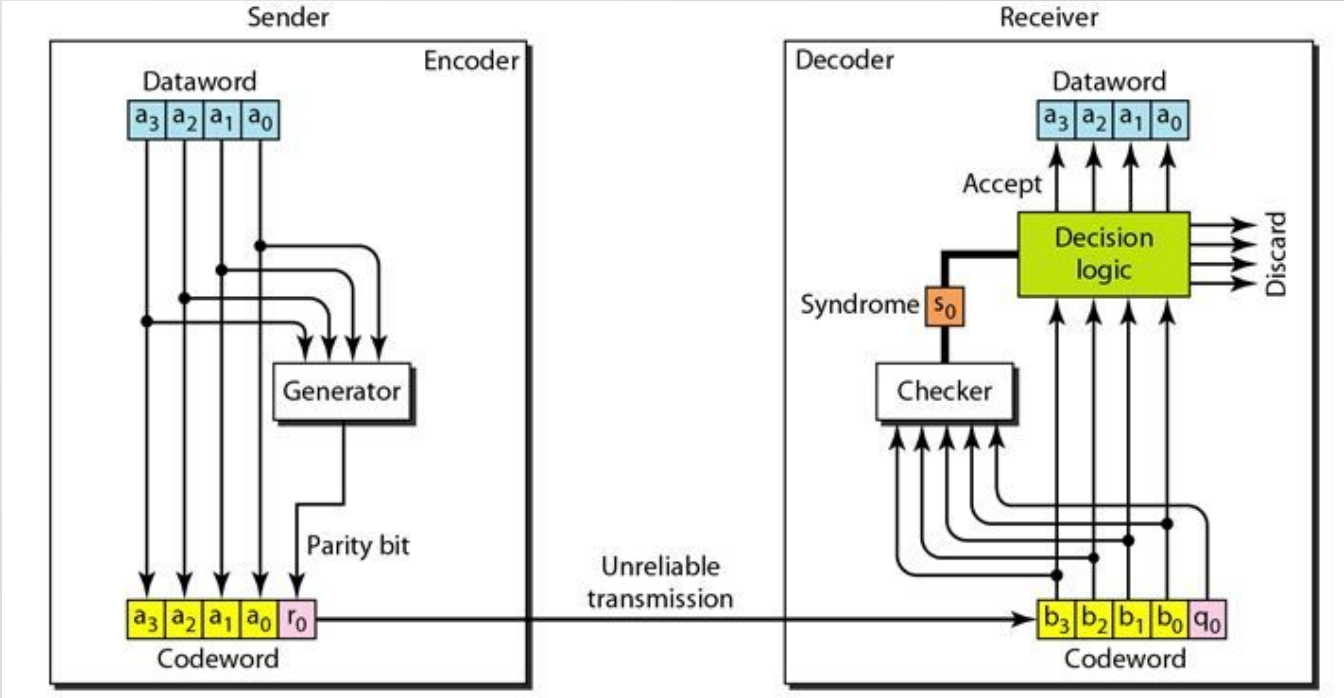
Error Detecting Codes

Minimum Hamming Distance

- In a set of codewords, the minimum hamming distance is the smallest hamming distance between all possible pairs of codewords.
- To guarantee the detection of up to 'n' errors, the minimum hamming distance must be $d_{min} = n + 1$.

Error Detecting Codes

- Parity
  - Sender adds parity bit.
  - Assume the coding scheme follows even parity.
  - If the data word contains odd number of 1's, then parity bit=1.
  - If the data word contains even number of 1's, then parity bit=0.
  - Detect only single bit errors.

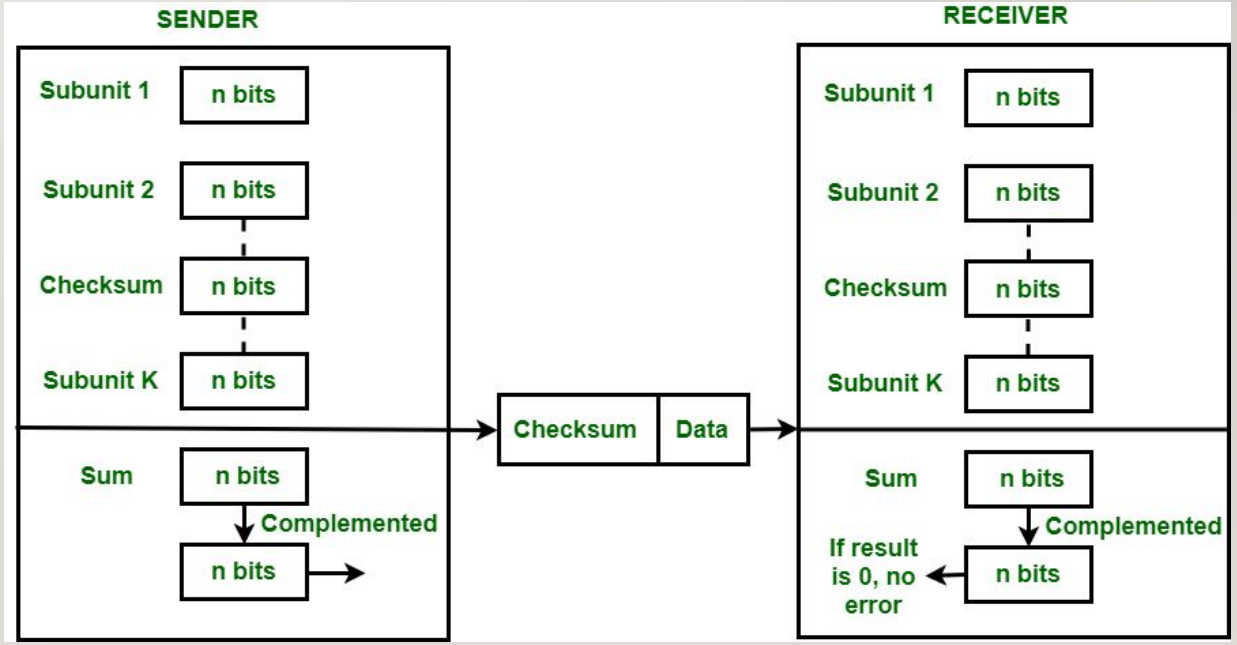


Error Detecting Codes-Checksum

- A **checksum** is a small-sized block of data derived from another block of digital data for the purpose of detecting errors.
- Error detecting technique that can be applied to message of any length.
- Sender side checksum creation
- Receiver side checksum validation

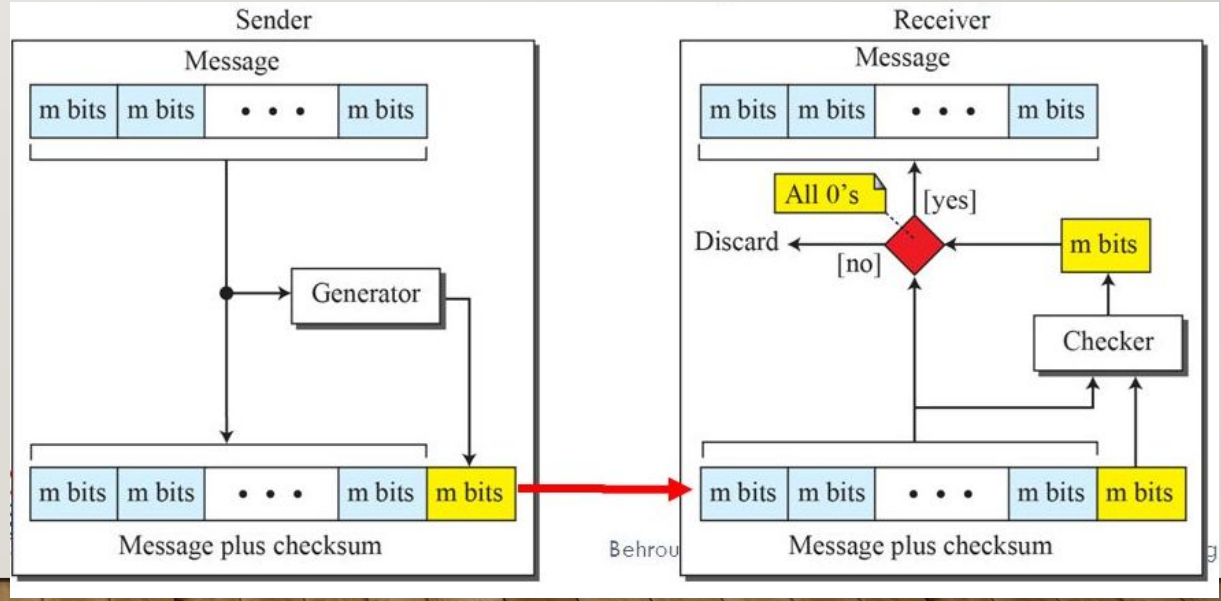
Error Detecting Codes-Checksum

- Steps
  - Break the original message into ‘k’ number of blocks with ‘n’ bits in each block.
  - Sum all the ‘k’ data blocks.
  - Add the carry to the sum, if any.
  - Checksum= 1’s complement of the sum obtained.



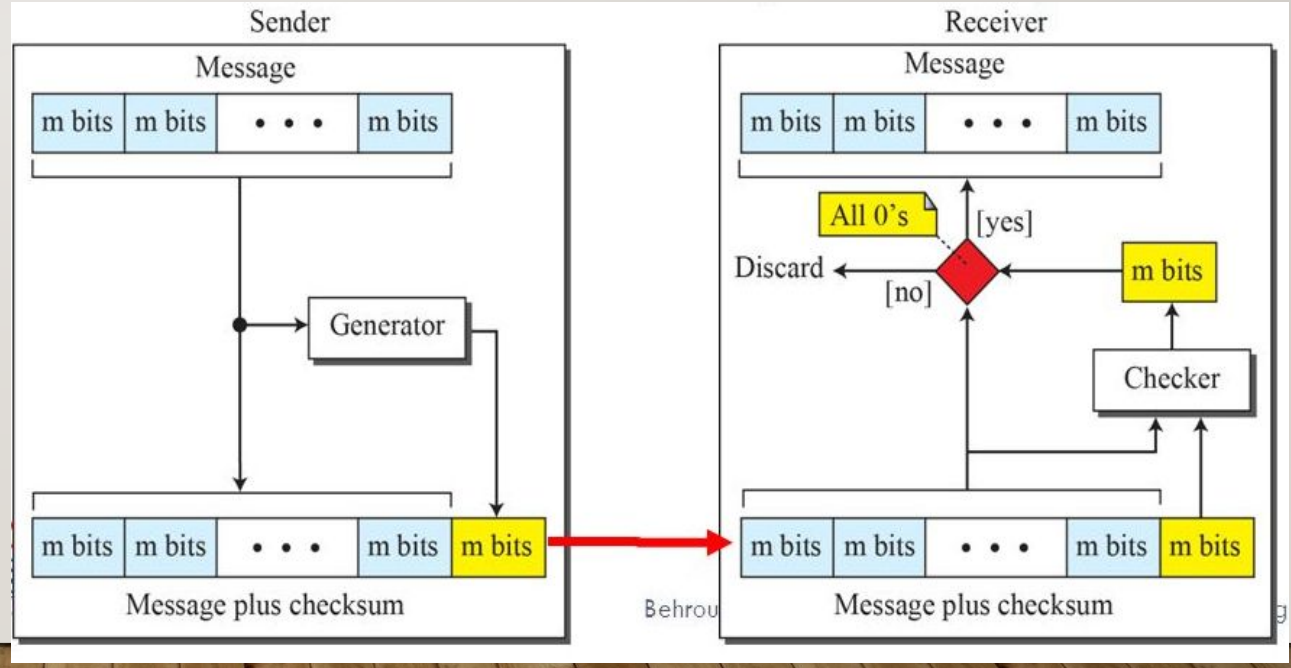
Error Detecting Codes-Checksum

- Steps: Sender Side
  - Break the original message into ‘k’ number of blocks with ‘m’ bits in each block.
  - Sum all the ‘k’ data blocks.
  - Add the carry to the sum, if any.
  - Checksum= 1’s complement of the sum obtained.



Error Detecting Codes-Checksum

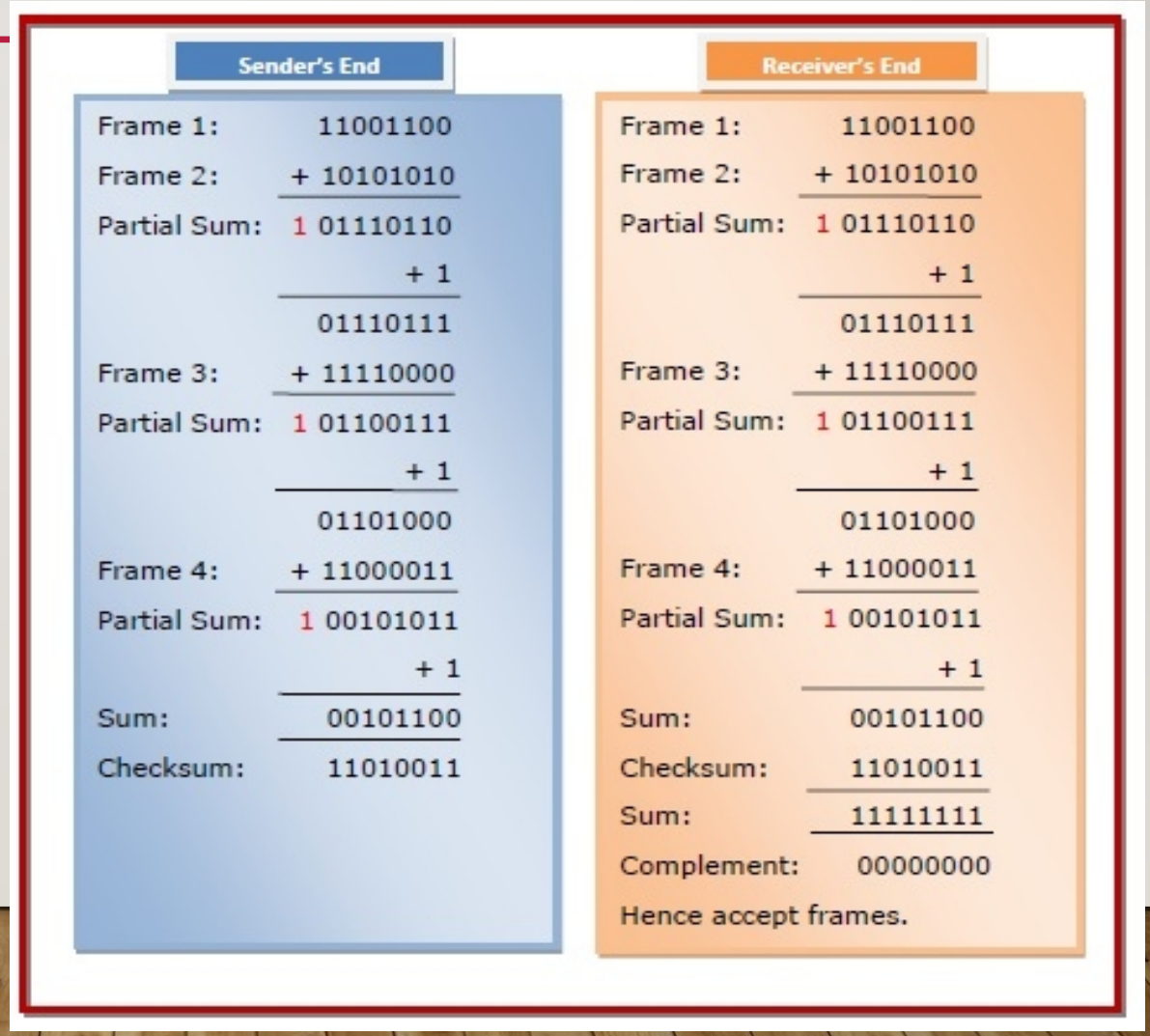
- Steps: Receiver Side
  - All received segments are added using 1's complement arithmetic to get the sum.
  - Complement the sum.
  - If the result is zero, the received data accepted; otherwise discarded.



Error Detecting Codes-Checksum

- Input: 1100110010101010

1111000011000011



Sender's End		Receiver's End	
Frame 1:	11001100	Frame 1:	11001100
Frame 2:	+ 10101010	Frame 2:	+ 10101010
Partial Sum:	1 01110110	Partial Sum:	1 01110110
	+ 1		+ 1
	01110111		01110111
Frame 3:	+ 11110000	Frame 3:	+ 11110000
Partial Sum:	1 01100111	Partial Sum:	1 01100111
	+ 1		+ 1
	01101000		01101000
Frame 4:	+ 11000011	Frame 4:	+ 11000011
Partial Sum:	1 00101011	Partial Sum:	1 00101011
	+ 1		+ 1
Sum:	00101100	Sum:	00101100
Checksum:	11010011	Checksum:	11010011
		Sum:	11111111
		Complement:	00000000
		Hence accept frames.	

Error Detecting Codes-Checksum

- Compute the checksum value of 10010011 10010011 10011000 01001101 of 16 bit segment.

- Ans: 1101010000011110

Error Detecting Codes-CRC

- Stands for Cyclic Redundancy Check (Polynomial code).
- Cyclic code: if a codeword is cyclically shifted (rotated), the result is another codeword.
- Polynomial code: treat bit strings as representations of polynomials with coefficients 0 and 1 only.
- k- bit frame→ polynomial with 'k' terms, ranging from x^{k-1} to x^0.
- E. g. $110001 \rightarrow 1x^5 + 1x^4 + 0x^3 + 0x^2 + 0x^1 + 1x^0$

Error Detecting Codes-CRC

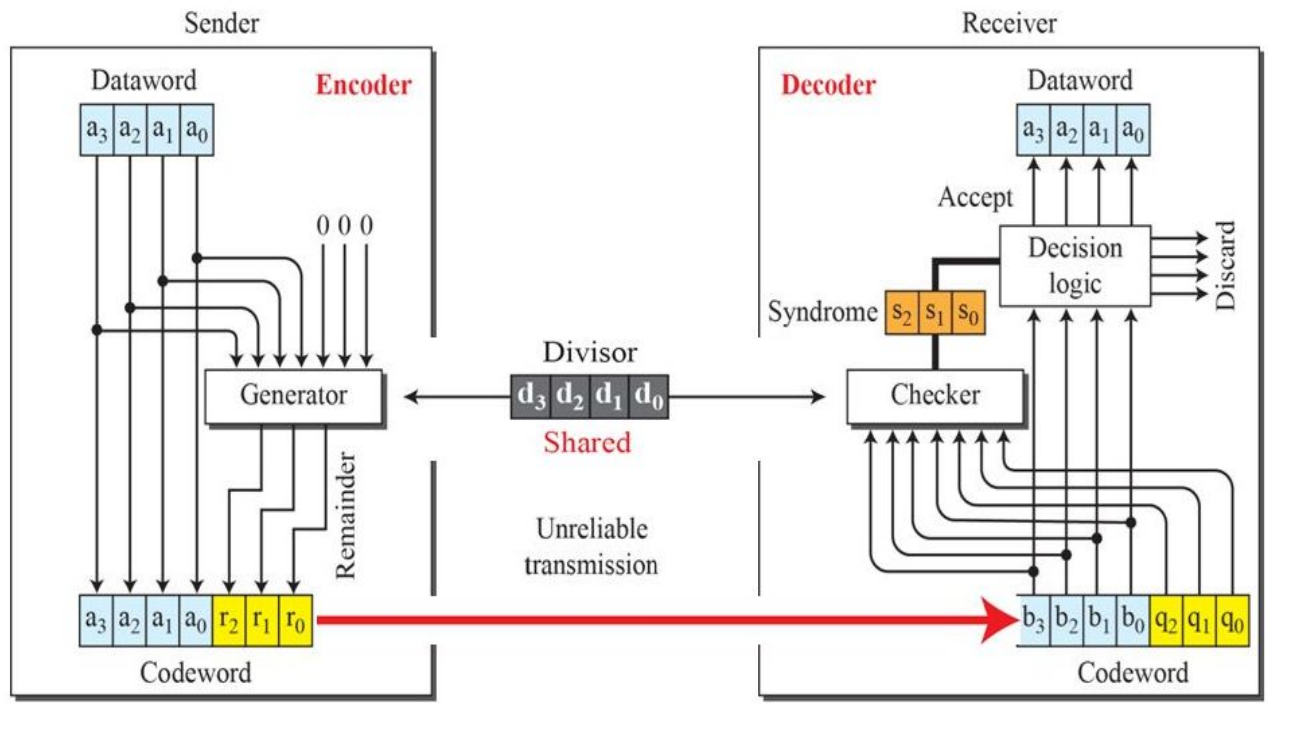
- Uses Modulo 2 arithmetic
  - No carry for addition or borrows for subtraction.
  - Addition and subtraction are identical to XOR.
  - CRC uses **Generator Polynomial, G(x)** which is agreed by both sender and receiver side.
  - An example generator polynomial is of the form like $x^3 + x + 1$, represents key 1011.
  - Both high and low order bits of the generator polynomial must be 1.

Error Detecting Codes-CRC

- To compute CRC for some frame with ‘m’ bits, the frame must be longer than the generator polynomial.
- Idea is to append a CRC to the end of the frame such that polynomial represented by the frame is divisible by G(x).
- Receiver tries dividing it by G(x). If there is any remainder □ transmission error.
- Let ‘r’ be the degree of G(x). Append ‘r’ zero bits to the low order end of the frame so that it contains m+r bits.

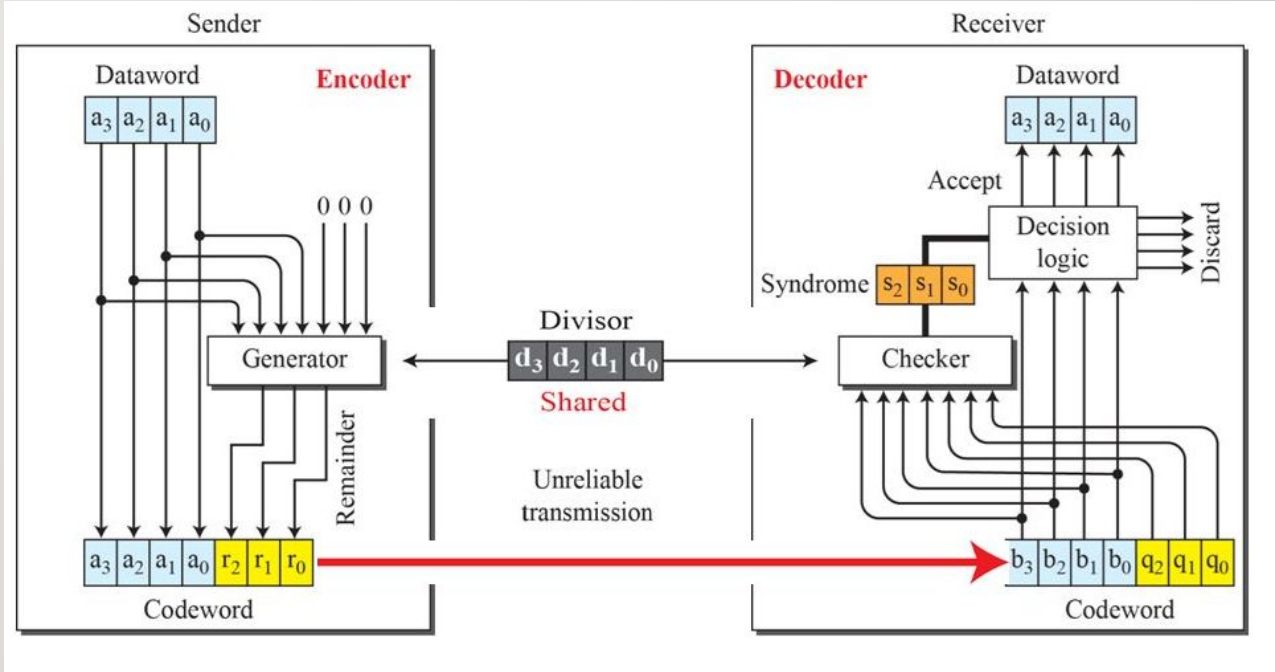
Error Detecting Codes-CRC

- Unlike checksum scheme, which is based on addition, CRC is based on binary division.
- Divisor is known to the sender and receiver in advance.
- Perform modulo-2 division.
- Discard the quotient.
- Append the remainder $(r_2 r_1 r_0)$ to the data word to form the code word.



Error Detecting Codes-CRC

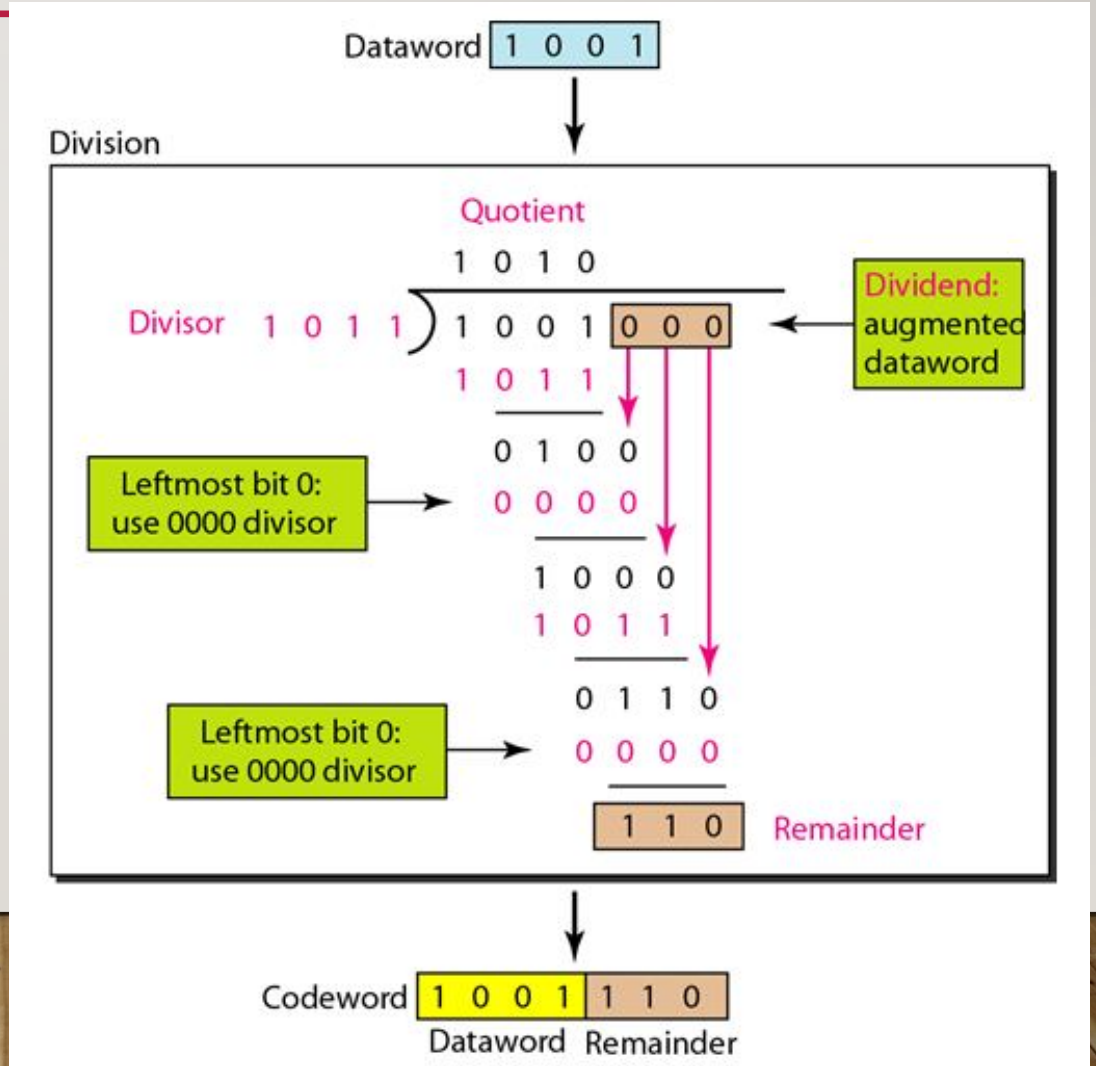
- Decoder
  - n- bits are fed to the checker.
  - The remainder produced by the checker is a syndrome of n-k bits and fed to the decision logic analyzer.
  - Accept if the syndrome bits are all zeros



Error Detecting Codes-CRC

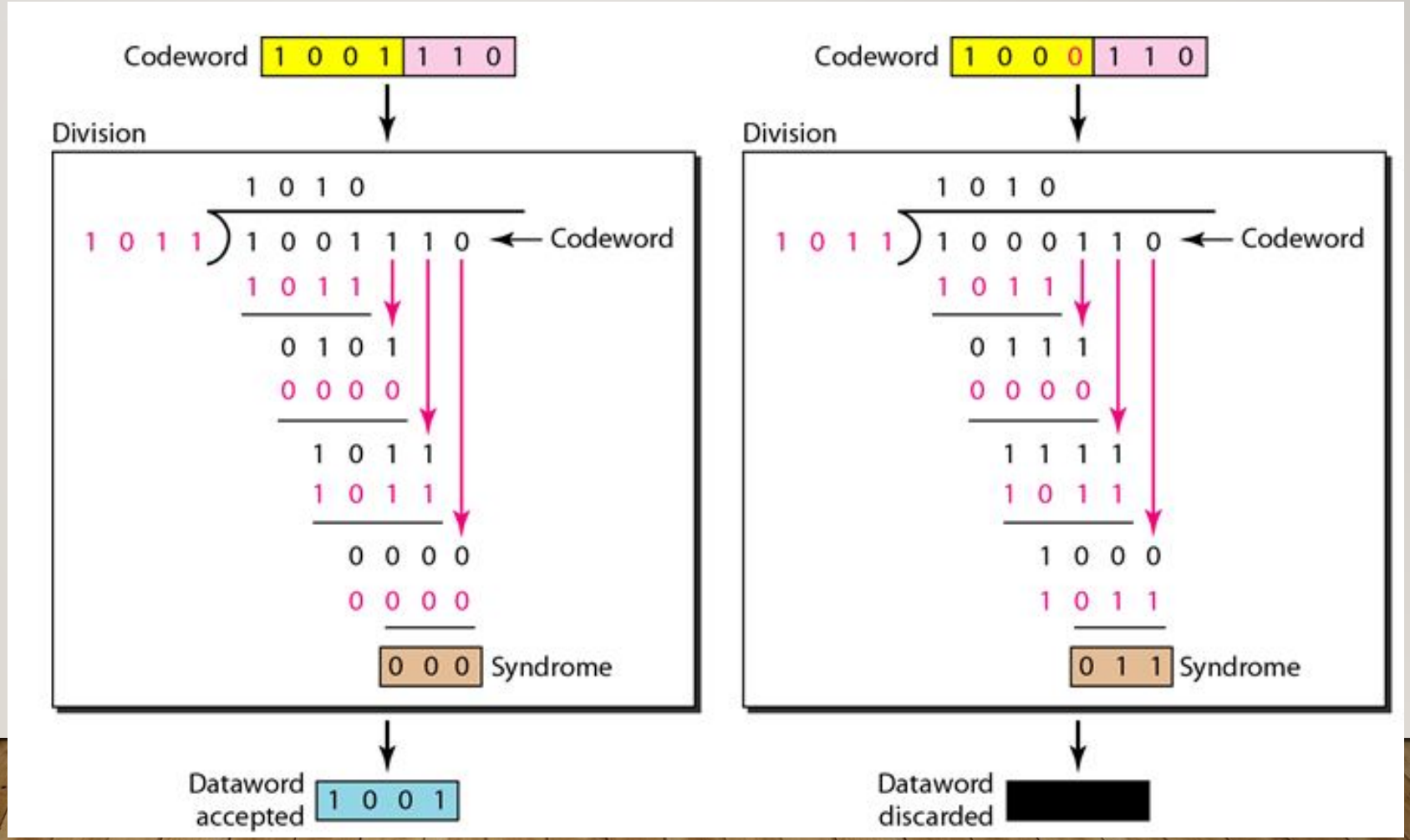
Encoder:

- Modulo 2 Arithmetic:



Error Detecting Codes-CRC

Decoder



Error Detecting Codes-CRC

- Q1. Station A wants to send a dataword 1101011111 to station B using CRC .The generator polynomial agreed by both A and B is $x^4 + x^1 + 1$. Find the transmitted codeword?

- Ans: 11010111110010

THANK YOU!!!

COMPUTER NETWORKS
MODULE 2

MS. JINCY J FERNANADEZ

ASST PROF, CSE

RSET

Introduction to DLL Protocols

- A link level protocol that wants to deliver frames reliably must recover the lost frames.

- Two fundamental mechanisms:
  - Acknowledgements
  - Time Outs

Acknowledgement

- An acknowledgement (ACK for short) is a small control frame that a protocol sends back to its peer saying that it has received the earlier frame.
- A control frame is a frame with header only (no data).
- The receipt of an acknowledgement indicates to the sender of the original frame that its data frame was successfully delivered.

Timeout

- If the sender does not receive an *acknowledgment* after a reasonable amount of time, then it retransmits the original frame.
- The action of waiting a reasonable amount of time is called a *timeout*.
- The general strategy of using *acknowledgements* and *timeouts* to implement reliable delivery is called Automatic Repeat reQuest (ARQ).

Protocols in ARQ

- Stop and Wait Protocol
- Sliding Window Protocol
  - Go Back N
  - Selective Repeat

Stop and Wait Protocol

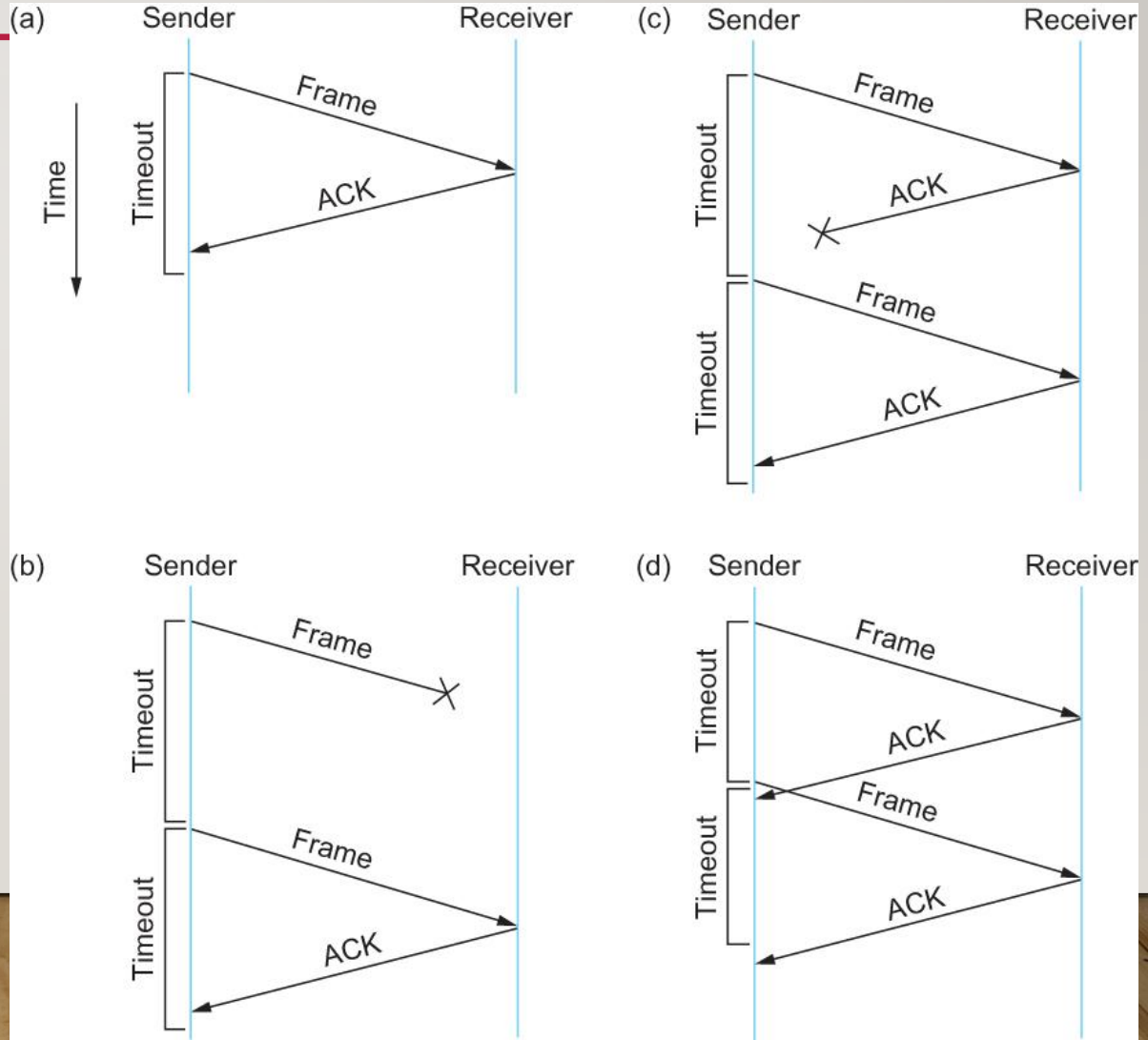
- Idea of stop-and-wait protocol is straightforward.
  - After transmitting one frame, the sender waits for an acknowledgement before transmitting the next frame.
  - If the acknowledgement does not arrive after a certain period of time, the sender times out and retransmits the original frame.

Stop and Wait Protocol

Timeline showing four different scenarios for the stop-and-wait algorithm.

(a) The ACK is received before the timer expires; (b) the original frame is lost;

(c) The ACK is lost; (d) the timeout fires too soon



Stop and Wait Protocol

- If the acknowledgment is lost or delayed in arriving:
  - The sender times out and retransmits the original frame, but the receiver will think that it is the next frame since it has correctly received and acknowledged the first frame.
  - As a result, duplicate copies of frames will be delivered.
- How to solve this??
  - Use 1 bit sequence number (0 or 1).
  - When the sender retransmits frame 0, the receiver can determine that it is seeing a second copy of frame 0 rather than the first copy of frame 1 and therefore can ignore it (the receiver still acknowledges it, in case the first acknowledgement was lost).

Stop and Wait Protocol

- Sender's Window size (Ws) = Receiver's Window size (Wr)= 1.
- Sequence number= [0,1]

Stop and Wait Protocol- Features

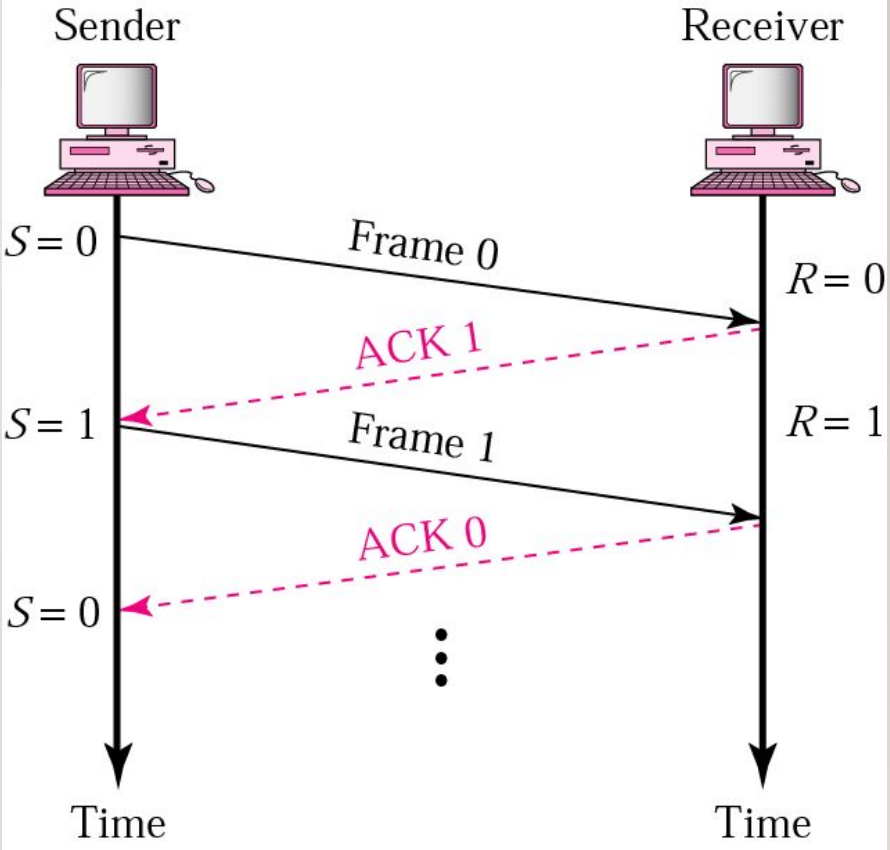
- The sending device keeps a copy of the sent frame transmitted until it receives an acknowledgment (ACK).
- The sender starts a timer when it sends a frame. If an ACK is not received within an allocated time period, the sender resends it.
- Both frames and acknowledgment (ACK) are numbered alternately 0 and 1(two sequence number only).
- This numbering allows for identification of frames in case of duplicate transmission.
- The acknowledgment number defines the number of next expected frame. (frame 0 received ACK 1 is sent).

Stop and Wait Protocol- Features

- A damage or lost frame treated by the same manner by the receiver.
- If the receiver detects an error in the received frame or receives a frame out of order, it simply discards the frame.
- The receiver send only positive ACK for frames received safe; it is silent about the frames damage or lost.
- The sender has a control variable holds the number of most recently sent frame S(0 or 1). The receiver has control variable R, that holds the number of the next frame expected (0 or 1).

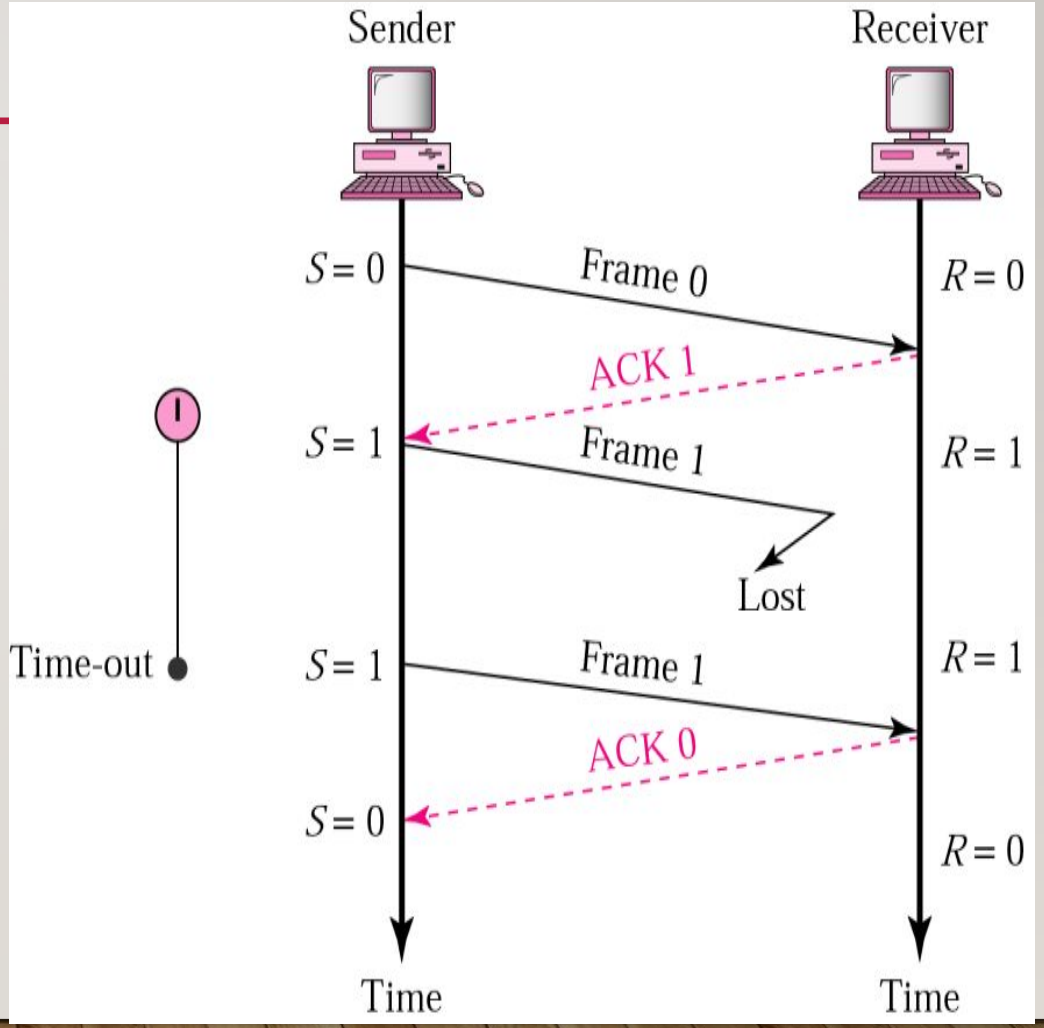
Stop and Wait Protocol- Normal operation

- The sender will not send the next frame until it is sure that the current one is received.
- Sequence number is necessary to check for duplicated frames .



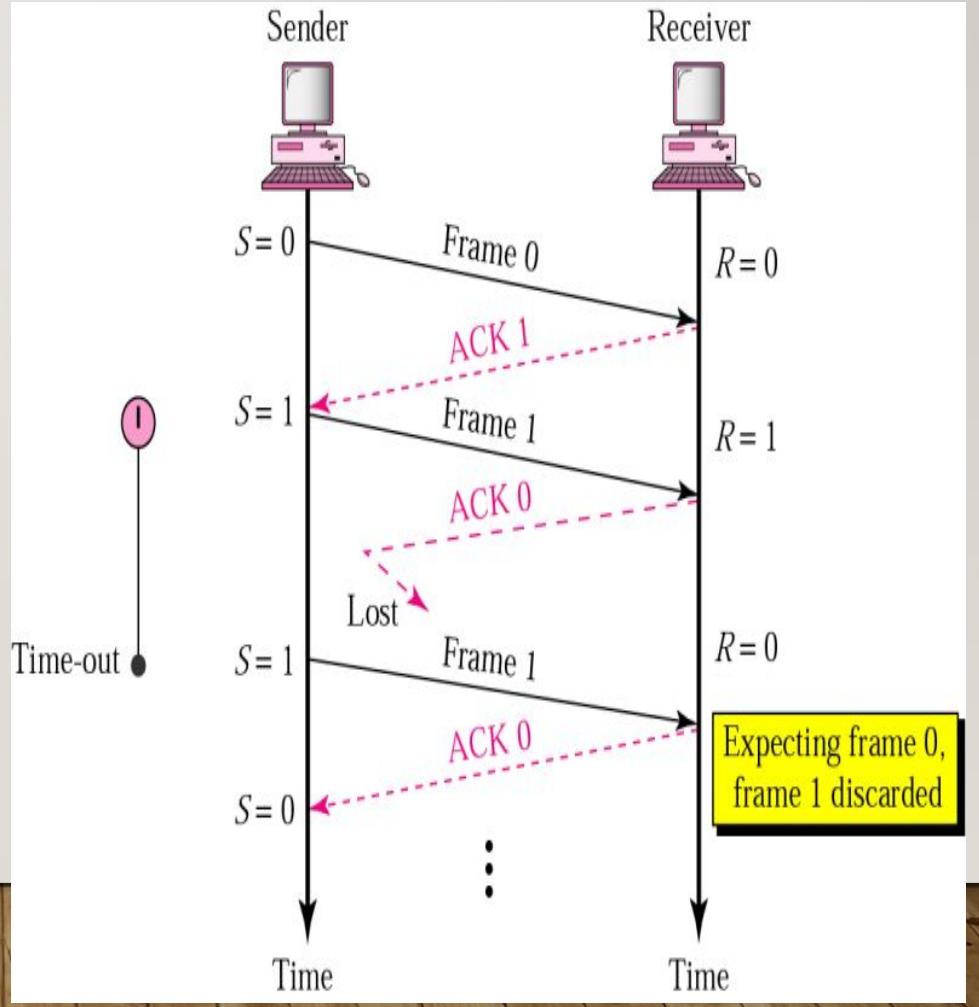
Stop and Wait Protocol- Lost or Damaged frame

- A damage or lost frame treated by the same manner by the receiver.
- No ACK when frame is corrupted or duplicate.



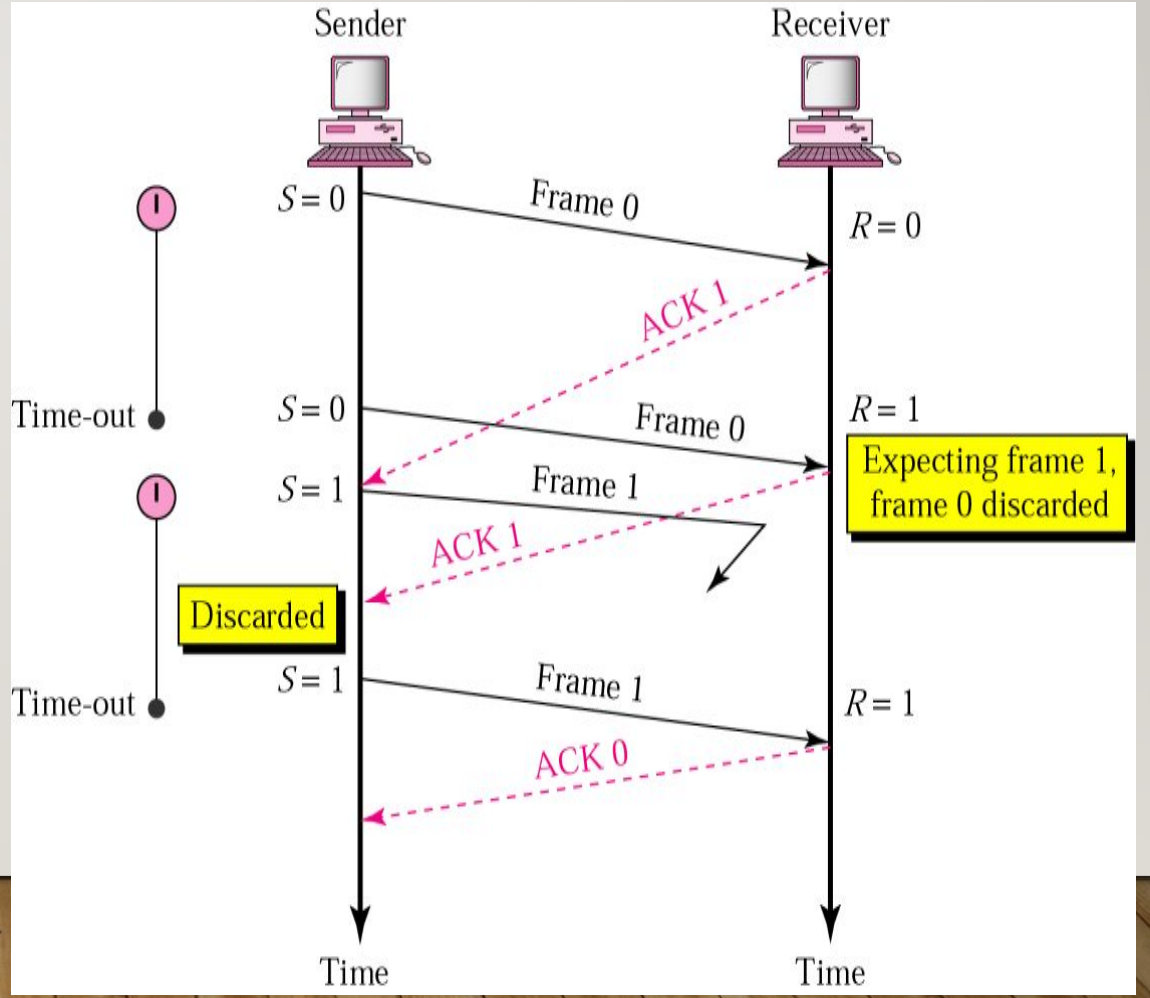
Stop and Wait Protocol- Lost ACK

- Importance of frame numbering.
- Prevents retaining duplicate frames.



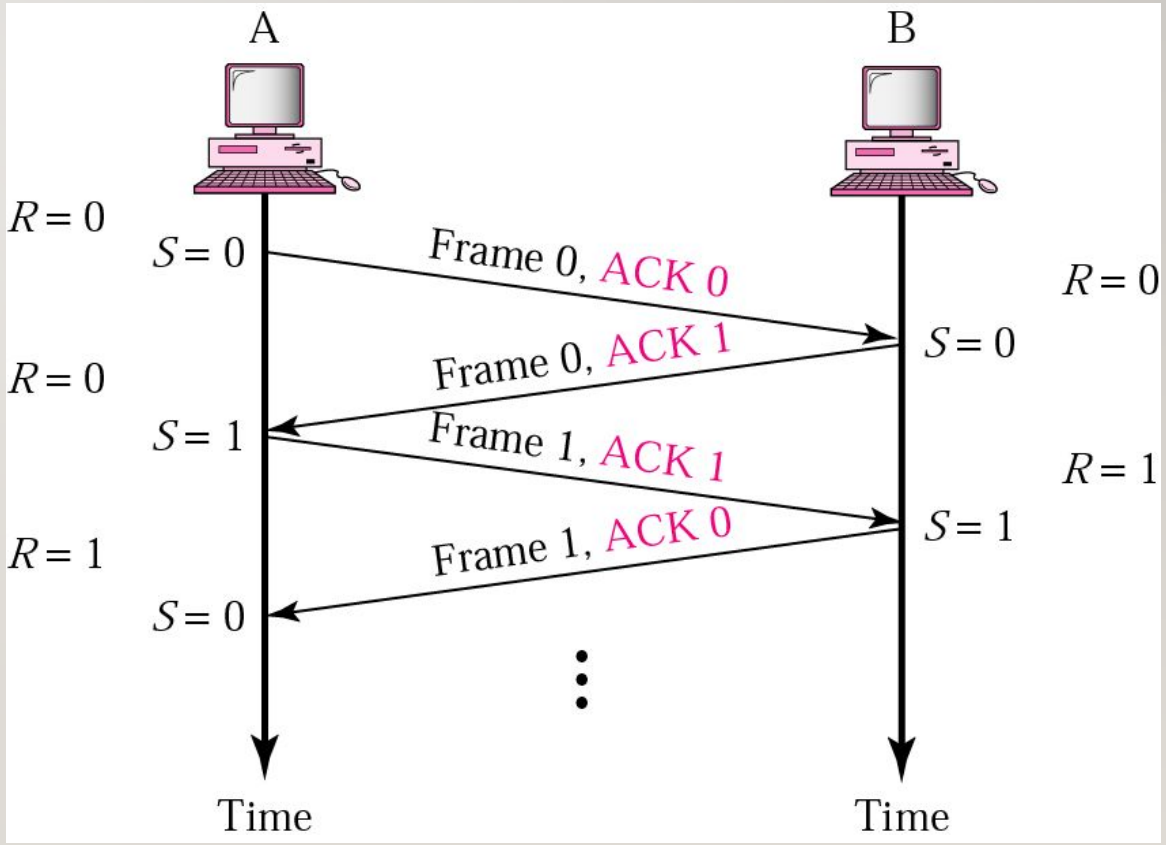
Stop and Wait Protocol- Delayed ACK & Lost frame

- Importance of frame numbering.
- Numbered acknowledgments are needed if an acknowledgment is delayed, and the next frame is lost.



Piggybacking

- Method to combine a data frame with an acknowledgment.
- Used in bidirectional cases.
- It can save bandwidth because data frame and an ACK frame can be combined into just one frame.



Stop and Wait- Disadvantage

- After each frame sent the host must wait for an ACK.
  - Inefficient use of bandwidth.
- To improve efficiency ACK should be sent after multiple frames only.
- Alternatives: Sliding Window protocol.

Sliding Window Protocols

- For sending multiple frames at a time, thus improves efficiency of the transmission.
- No. of frames to be sent at a time is based on window size.
- Outstanding frames: frames sent but not acknowledged.
- Can send up to W frames and keep a copy of outstanding frames until the ACKs arrive.
- Each frame is numbered sequence number.
- Sequence number is stored in the header of the frame.
- If the header has ‘m’ bits for sequence number, it ranges from 0 to 2^m -1.
- If m = 3, sequence ranges from 0 to 7.

Sliding Window Protocols

- Sliding window is used to hold the unacknowledged outstanding frames.
- 2 bidirectional sliding window protocols with (max sending & receiving window size) are:
  - 1-bit sliding window (1,1)
  - Go back N ARQ(>1,1)
  - Selective Repeat ARQ(>1,>1)
- They differ in efficiency, complexity and buffer requirements.

1-bit Sliding Window Protocol

- Window size=1.
- Uses stop and wait since the sender transmits a frame and waits for its acknowledgement before sending the next one.

Go back N Sliding Window Protocol

- Uses the concept of protocol pipelining: multiple frames can be transmitted before receiving acknowledgement for the first frame.
- Finite number of frames and frames are numbered sequentially.
- The number of frames that can be send depends on sender window size.
- If acknowledgement of a frame is not received within the time interval, all frames in the current window are retransmitted.

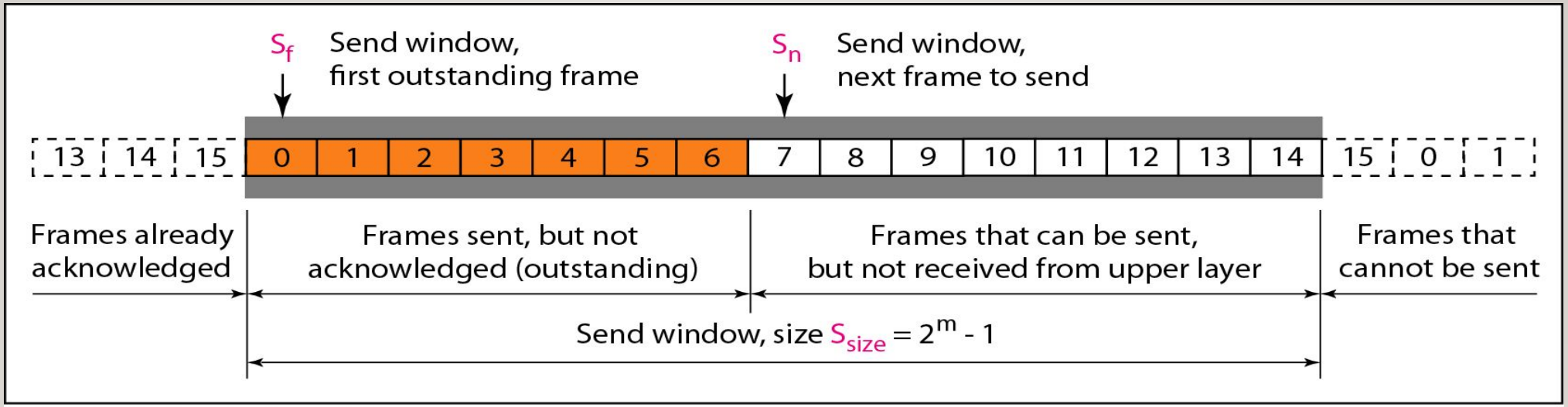
Go back N Sliding Window Protocol

- Uses cumulative acknowledgement technique
- 'N' → Sender's window size ($W_s = N$).
- 'N' is the number of frames that can be sent at a time before receiving acknowledgement.
- Receiver's window size is always 1 ($W_R = 1$).

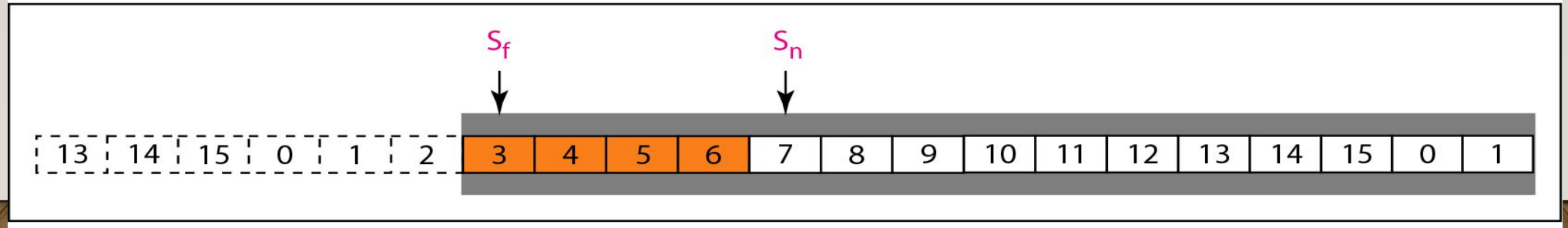
Go back N Sliding Window Protocol

- Size of the sending window determines the sequence numbers of the frames.
- E. g. if the sending window size=4, then the sequence numbers will be 0, 1, 2, 3, 0, 1, 2, 3 and so on.
- Let 'm' represents number of bits to represent sequence number, the range of sequence numbers assigned are 0 to $2^m - 1$.

Go back N Sliding Window Protocol



a. Send window before sliding



b. Send window after sliding

Go back N Sliding Window Protocol- Features

- One timer for the first outstanding frame.
- The receiver sends a positive ACK if a frame has arrived safe and in order.
- If a frame is damaged or out of order ,the receiver is silent and will discard all subsequent frames.
- When the timer of an unacknowledged frame at the sender site is expired , the sender goes back and resend all frames , beginning with the one with expired timer.

Selective Repeat Sliding Window Protocol

- Go back N protocol works well when errors are rare.
- Wastes a lot of bandwidth on retransmitted frames.
- Resent only the damaged frame while the correct frames are received and buffered.
- Receiver keeps track of the sequence numbers, buffers the frames in memory and send NACK for only frames which is missing or damaged.
- Both sender and receiver maintain a window of outstanding and acceptable sequence numbers, respectively.

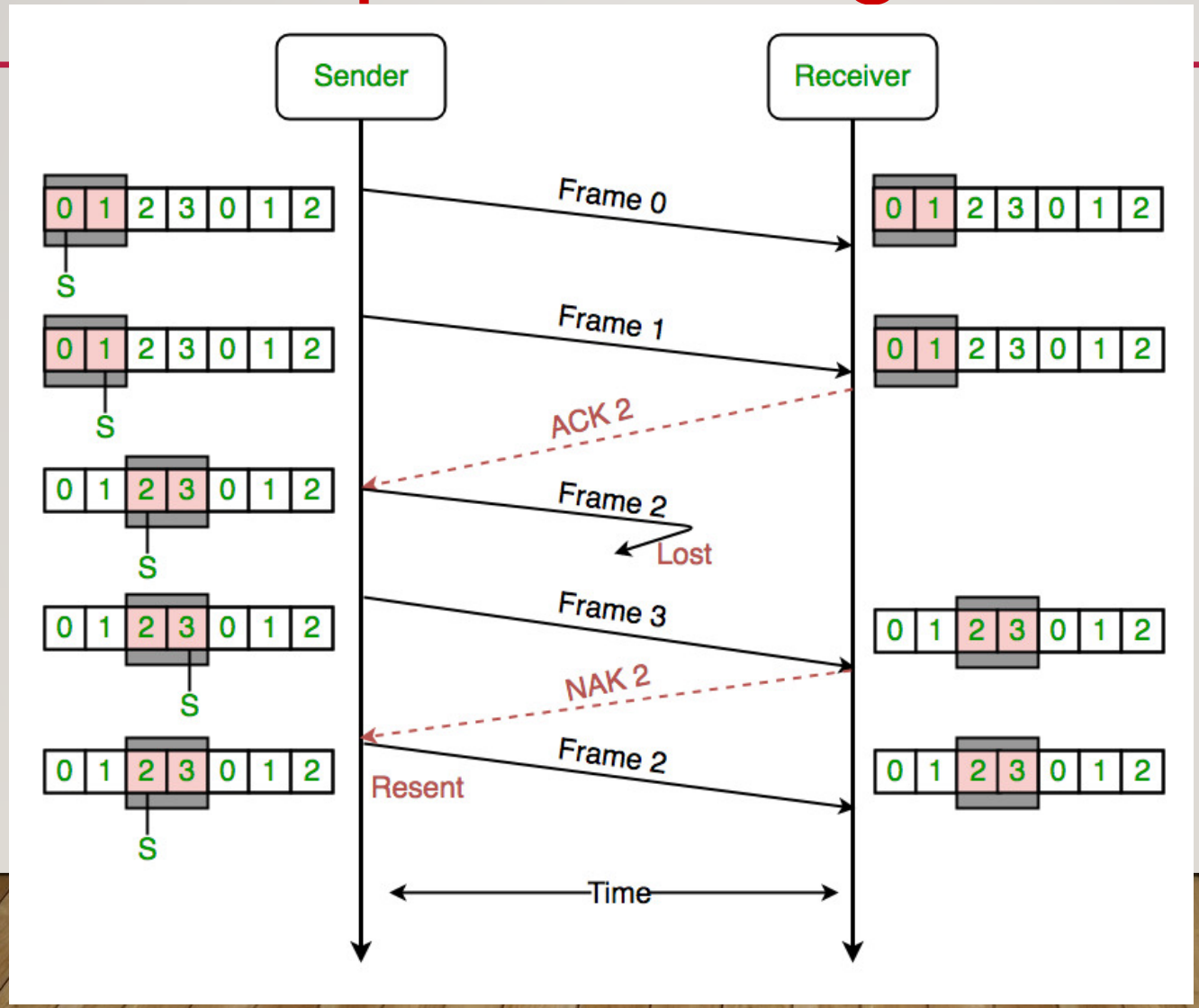
Selective Repeat Sliding Window Protocol

- Seq number $=2^m$, where 'm' is the number of bits to represent sequence number.
- E. g. : Let m=2, seq num=[0,1,2,3]
- Sender's Window size (Ws) = Receiver's Window size (Wr)= 2^{m-1}.
- Window size should be less than or equal to half the sequence number in Selective Repeat protocol ($W_s \leq \frac{2^m}{2} \leq 2^{m-1}$).
- Receiver must be able to accept packets out of order.
- Since receiver must release packets to higher layer in order, the receiver must be able to buffer some packets.

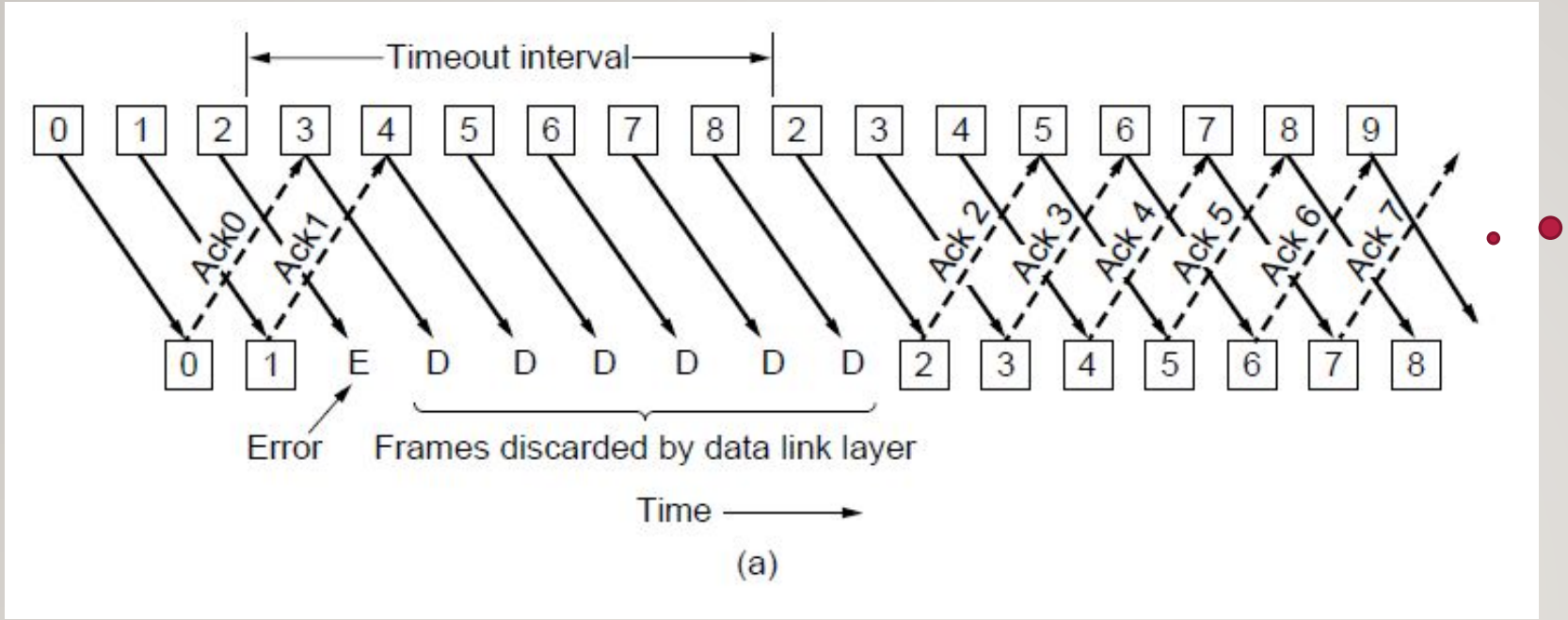
Selective Repeat Sliding Window Protocol

- When a frame arrives , its sequence number is checked to see if it falls within the window.
- If so, and if it has not already been received, it is accepted and stored.
- No. of retransmissions < that in Go back N ARQ
- Sender will retransmit only the packets for which NACK is received.
- It is more efficient for noisy link, but the processing at the receiver is more complex.

Selective Repeat Sliding Window Protocol

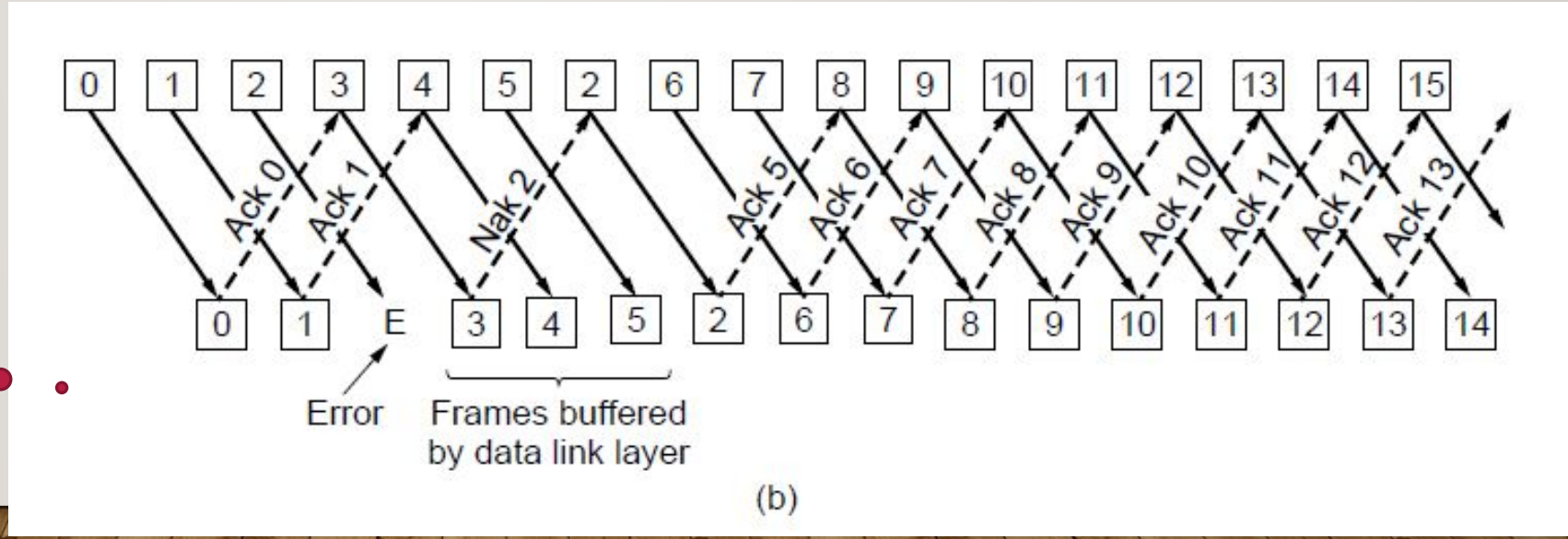


GO BACK N	SELECTIVE REPEAT
• Receiver in case of error discards all subsequent frames.	• Receiver in case of error discards bad frame, buffers all good frames.
• Sends no ACK for discarded frames.	• Sends a (NAK) when it detects an error.
• Receiver window size =1.	• Receiver window size >1.
• Data link refuses to accept any frame except the next one it must give to the network layer.	• Buffers remaining frames received.
• Eventually sender will time out and retransmit all unacknowledged frames in order starting with damaged or lost one.	• When sender times out, only the oldest unacknowledged frame is retransmitted as others are already buffered.
• Waste of bandwidth.	• Better use of bandwidth.



Go back N

Selective Repeat



Sliding Window Protocol

- Q1. Using 5 bit sequence numbers what is the maximum size of sender and receiver window for the following protocols: a) stop and wait ARQ (b) go back n ARQ (c) selective repeat ARQ
- (a) $W_s = 1 \quad W_R = 1$
- (b) The maximum sender window size in go-back n is (2^n)-1, n is the sequence number.

$$W_S = 31, \quad W_R = 1$$

- (c) The maximum window size is (2^n)/2

$$W_S = W_R = 16$$

Sliding Window Protocol

- Q2. A sender sends a series of packets to the same destination using 5-bit sequence numbers. If the sequence number starts with 0, what is the sequence number after sending 100 packets?

With 5 bits, no of sequence no. possible is 32 ie (0-31)

For frames 1-32 seq no 0-31

For frame 33-64 seq no 0-31

For frame 65-96 seq no 0-31

Now 97 98 99 100

0 1 2 3

So after sending 100 packets sequence no is 4

Sliding Window Protocol

- Q3. Station A needs to send a message consisting of 9 packets to station B using a sliding window (window size=3) and go back N strategy. All packets are ready and immediately available for transmission. If every 5^{th} packet that A transmits gets lost (but no ACKs from B ever get lost), then what is the number of packets that A transmit to B?

- Ans: 16

Sliding Window Protocol

- Q4. Host A wants to send 10 frames to host B. The hosts agreed to go with Go back 4. How many frames are transmitted by Host A if every 6^{th} frame that is transmitted by host A is either corrupted or lost?

- Ans: 17

Sliding Window Protocol

- Q5. In SR protocol, suppose frames through 0 to 4 have been transmitted. Now imagine that 0 times out, 5(a new frame) is transmitted, 1 times out, 2times out and 6 (another new frame) is transmitted. At this point, what will be the outstanding packets in sender's window?

- Ans: 3 4 0 5 1 2 6

Sliding Window Protocol

- Q4. Host A wants to send 10 frames to host B. The hosts agreed to go with SR protocol. How many frames are transmitted by Host A if every 6^{th} frame that is transmitted by host A is either corrupted or lost?

- Ans: 11

THANK YOU!!!

COMPUTER NETWORKS MODULE 2

MS. JINCY J FERNANDEZ

ASST. PROF, CSE

RSET

HDLC(High Level Data link Control) Protocol

- Bit – oriented protocol: views the frame as a collection of bits.
- IBM developed SDLC (Synchronous Data Link Control) protocol as a bit oriented protocol.
- ISO standardized SDLC to HDLC.
- Basis for many other DLL protocols.
- Used for communication over point – to –point and multipoint links.
- Implements Stop-and-Wait protocol.
- Supports error and flow control.

HDLC(High Level Data link Control) Protocol

- Types of stations:
  - Primary station: can send commands.
  - Secondary station: can only respond.
  - Combined: can both send and give response.

HDLC(High Level Data link Control) Protocol

	Flag	Header	Body	CRC	Flag
bits	8	16		16	8

- Flag: synchronization pattern
  - 01111110
  - start and end of the frame.
  - also transmitted during any times that the link is idle so that sender and receiver can keep their clocks synchronized.

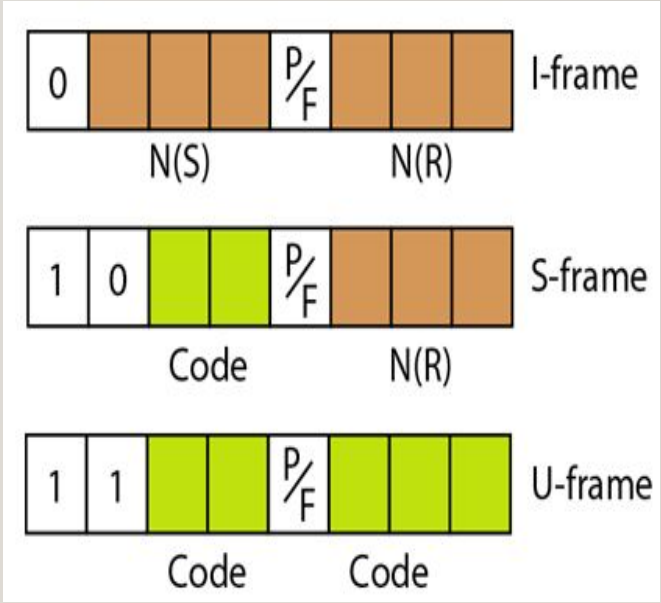
HDLC(High Level Data link Control) Protocol

	Flag	Header	Body	CRC	Flag
bits	8	16		16	8

- Header: address field and control field.
  - Address field: address of the secondary station.
  - Control field: to identify the types of HDLC frames; used for flow and error control.
- Body: Payload (varying size)
- CRC: Cyclic Redundancy Check: error detection

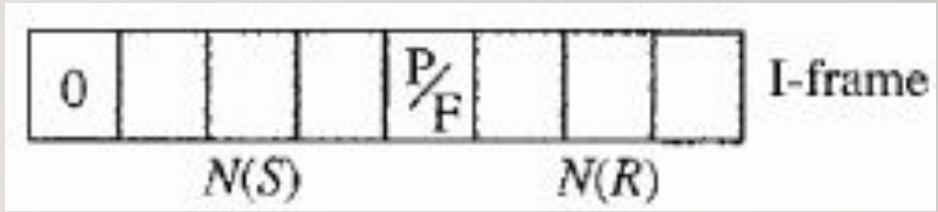
HDLC Protocol- Types of frames

1. I- Frames: Information Frame
   - carry user data from the network layer.
   - flow and error control information (piggybacking)
2. S-Frames: Supervisory Frame
   - flow and error control information (without piggybacking)
   - do not have information fields.
3. U- Frames: Un-numbered Frame
   - no user data
   - system management information



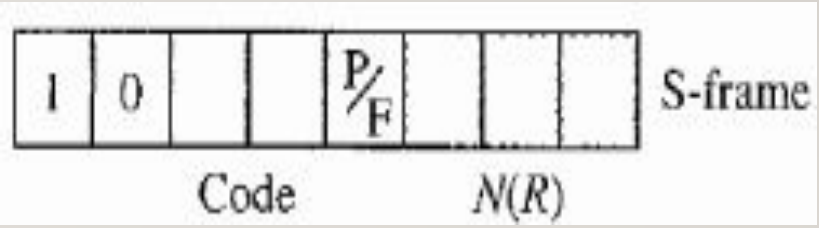
Control Field format for I- frames

- First bit – defines the type (0 for I-frame).
- Next 3 bits N(S) – sequence number of the frame.
- Last 3 bits N(R) – acknowledgement number when piggybacking is used.
- P/F bit – meaningful only when set to 1.
  - Poll/ final
  - Poll when the frame is sent by a primary station to a secondary station.
  - Final when the frame is sent by a secondary station to a primary station.



Control Field format for S- frames

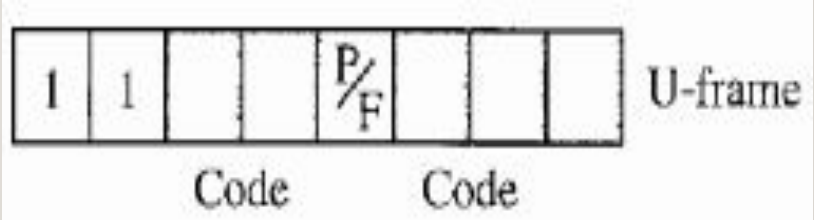
- Used for flow and error control when piggybacking is impossible or inappropriate.
- First 2 bits 10 for S-frame.
- Last 3 bits N(R) - acknowledgment number or negative acknowledgment
- 3rd and 4th bits (code) – type of S-frame
  - 4 types of frames:



Code Bits	Type	Purpose
00	Receive Ready (RR)	Acknowledges the receipt of a safe and sound frame or a group of frames N(R) field contains acknowledgment number
10	Receive not Ready (RNR)	Acknowledges the receipt of a safe and sound frame or a group of frames and announces that the receiver is busy and cannot receive more frames (acts as a congestion control mechanism) N(R) field contains acknowledgment number
01	Reject (REJ)	NAK frame used in Go-Back-N ARQ Informs the sender before the sender time expires that the last frame is lost or damaged N(R) field contains negative acknowledgment number
11	Selective Reject (SREJ)	NAK frame used in Selective Repeat ARQ N(R) field contains negative acknowledgment number

Control Field format for U- frames

- Used to exchange session management and control information between connected devices.
- Information field used for system management information.
- First 2 bits 11 for U-frame.
- U-frame codes divided into 2 sections
  - 2 bit prefix before the P/F bit
  - 3 bit prefix after the P/F bit
- 32 different types of U-frames.

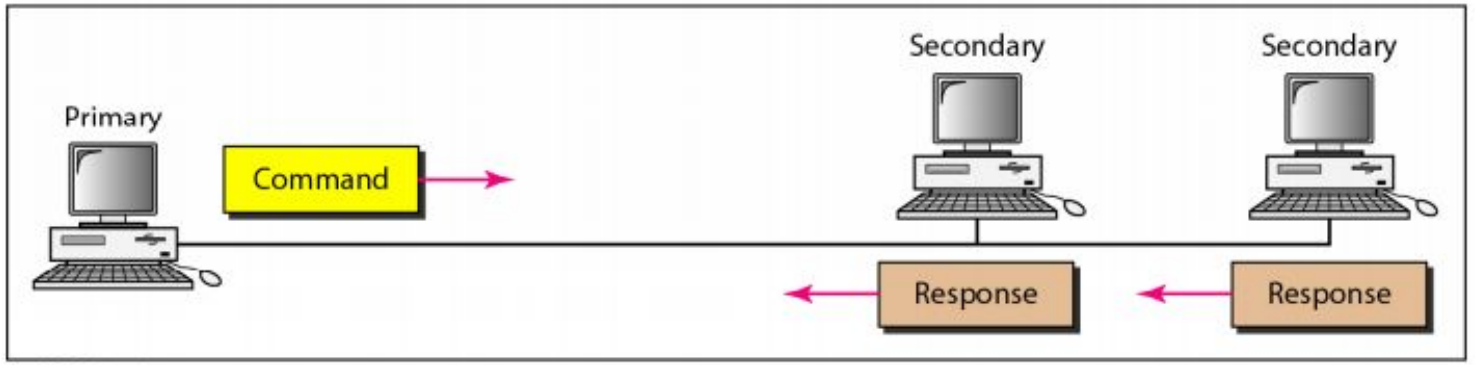


HDLC Protocol- Transfer modes

- Provides two transfer modes:
  - Normal Response Mode (NRM)
    - Unbalanced station configuration.
    - One primary station and multiple secondary stations.
    - Primary stations can send commands.
    - Secondary stations can only respond to commands.
    - Used for point-to-point and multipoint links.
  - Asynchronous Balanced Mode (ABM)

NRM

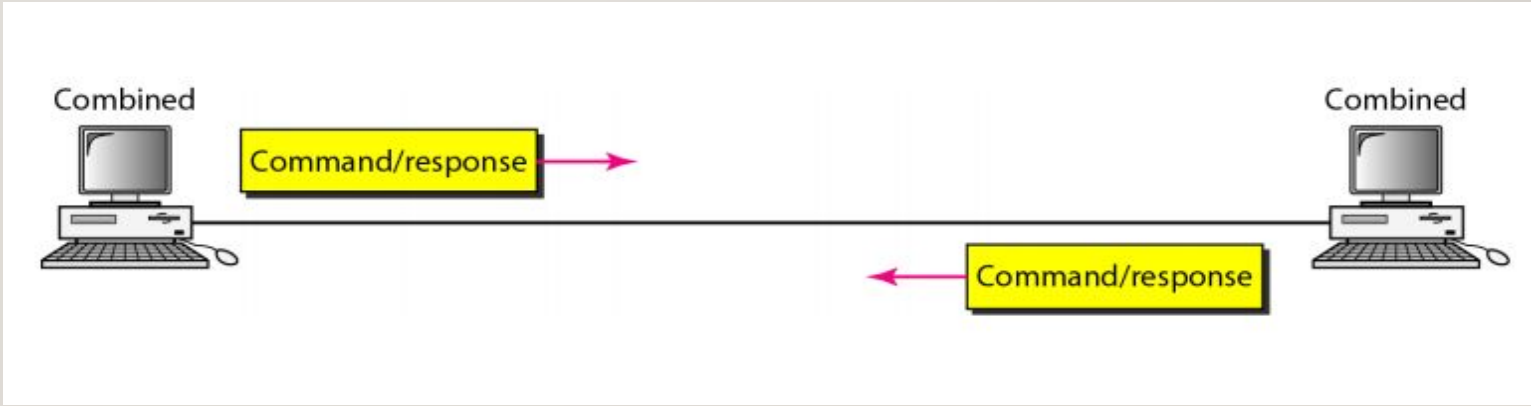
a. Point-to-point



b. Multipoint

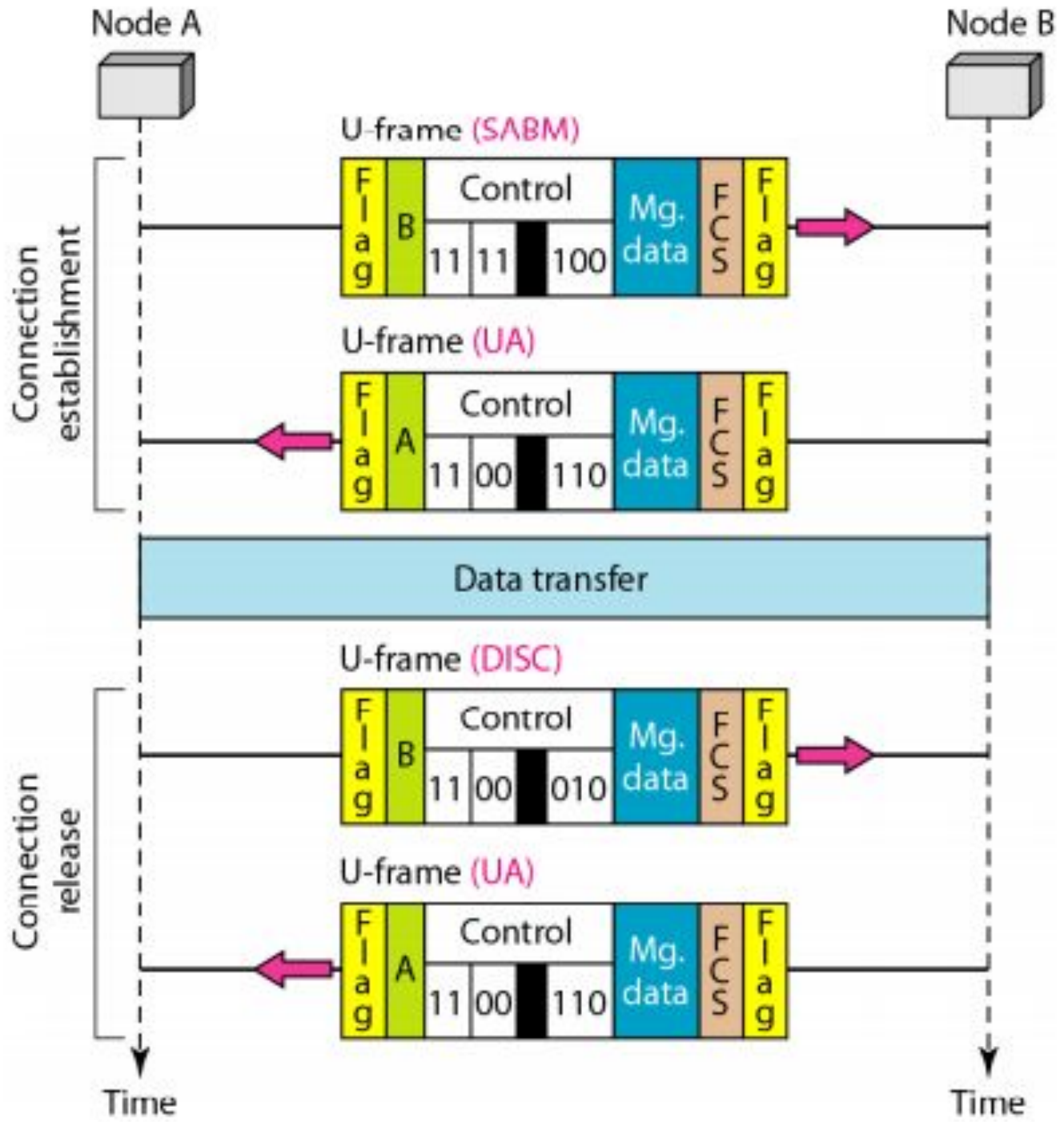
HDLC Protocol- Transfer modes

- Provides two transfer modes:
  - Asynchronous Balanced Mode (ABM)
    - Link is point-to-point in nature.
    - Each station can function as primary and secondary stations (peers).
    - Commonly used mode.

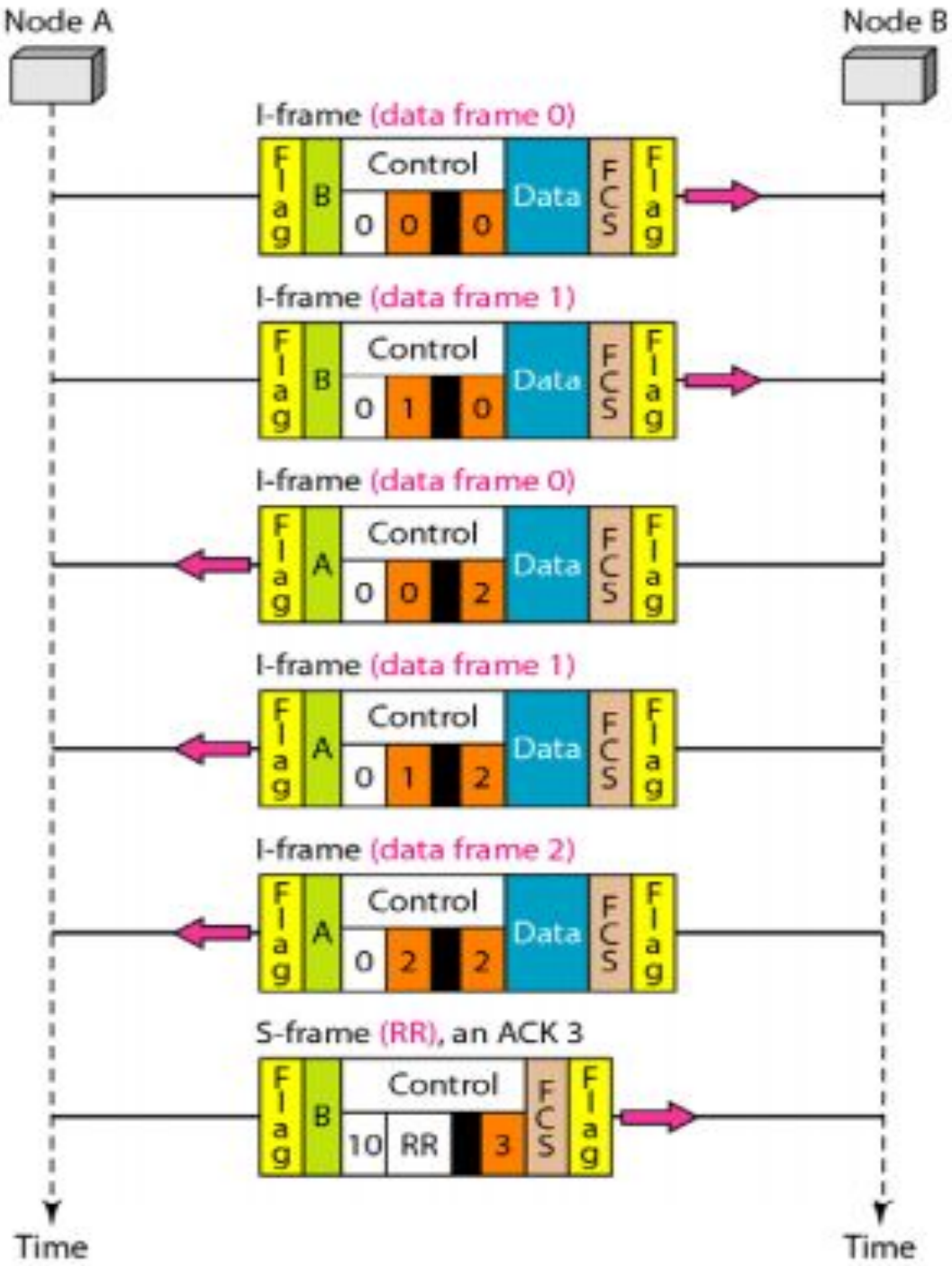


Code	*Command*	*Response*	*Meaning*
00 001	SNRM		Set normal response mode
11 011	SNRME		Set normal response mode, extended
11 100	SABM	**DM**	Set asynchronous balanced mode or **disconnect mode**
11 110	SABME		Set asynchronous balanced mode, extended
00 000	UI	**UI**	Unnumbered information
00 110		**UA**	**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

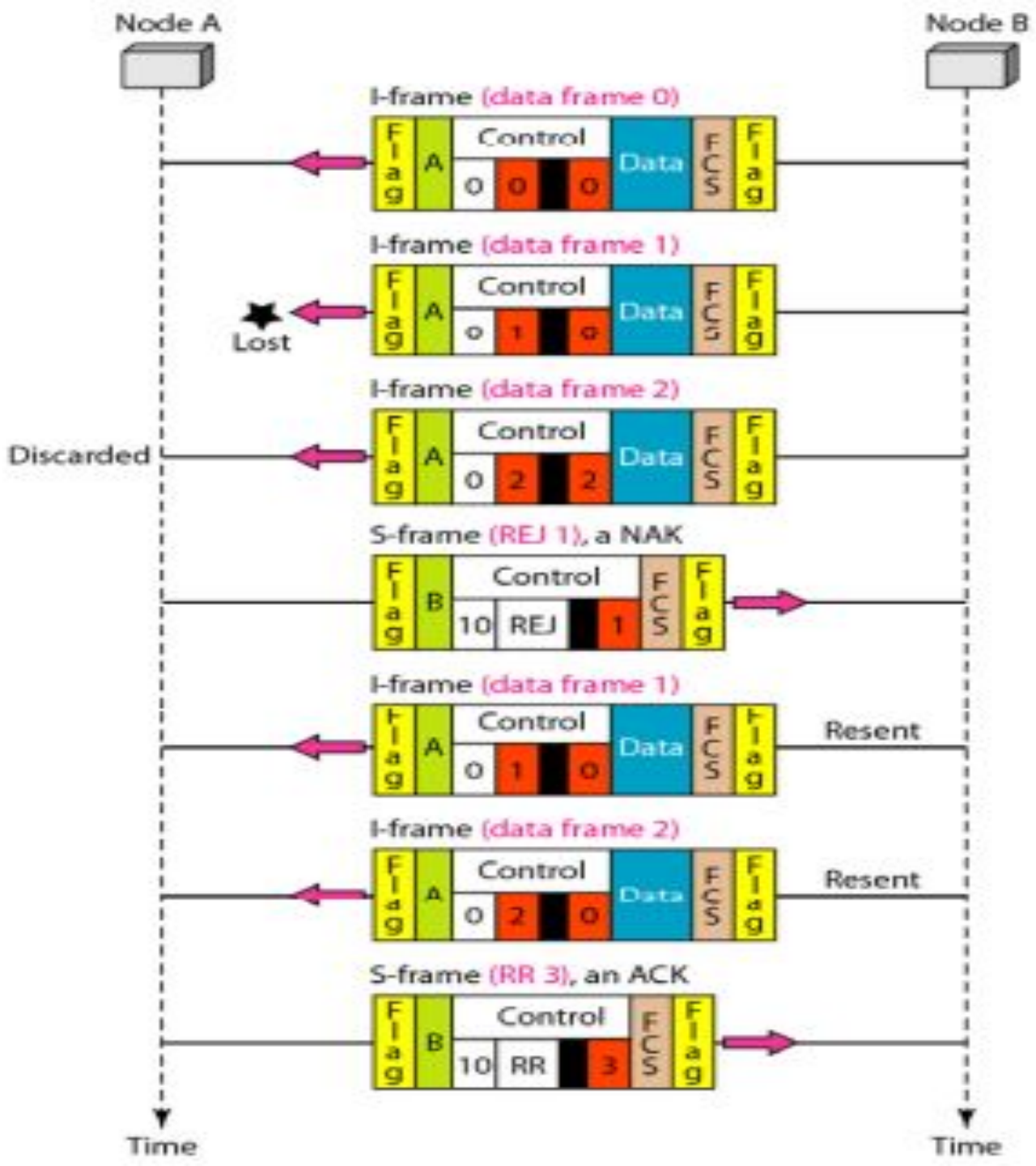
EXAMPLE 1


Node A
Node B
U-frame (SABM)
Flag
B
Control
11 11 100
Mg. data
FCS
Flag
Connection establishment
U-frame (UA)
Flag
A
Control
11 00 110
Mg. data
FCS
Flag
Data transfer
U-frame (DISC)
Flag
B
Control
11 00 010
Mg. data
FCS
Flag
Connection release
U-frame (UA)
Flag
A
Control
11 00 110
Mg. data
FCS
Flag
Time
Time

EXAMPLE 2


Node A
Node B
I-frame (data frame 0)
Flag
B
Control
0 0 0
Data
FCS
Flag
I-frame (data frame 1)
Flag
B
Control
0 1 0
Data
FCS
Flag
I-frame (data frame 0)
Flag
A
Control
0 0 2
Data
FCS
Flag
I-frame (data frame 1)
Flag
A
Control
0 1 2
Data
FCS
Flag
I-frame (data frame 2)
Flag
A
Control
0 2 2
Data
FCS
Flag
S-frame (RR), an ACK 3
Flag
B
Control
10 RR 3
FCS
Flag
Time
Time

EXAMPLE 3


Node A
Node B
I-frame (data frame 0)
Flag
A
Control
0
0
0
Data
FCS
Flag
I-frame (data frame 1)
Lost
Control
0
1
0
I-frame (data frame 2)
Discarded
Control
0
2
2
S-frame (REJ 1), a NAK
B
Control
10
REJ
1
I-frame (data frame 1)
Resent
Control
0
1
0
I-frame (data frame 2)
Resent
Control
0
2
0
S-frame (RR 3), an ACK
Control
10
RR
3
Time
Time

THANK YOU!!!

COMPUTER NETWORKS MODULE 2.4

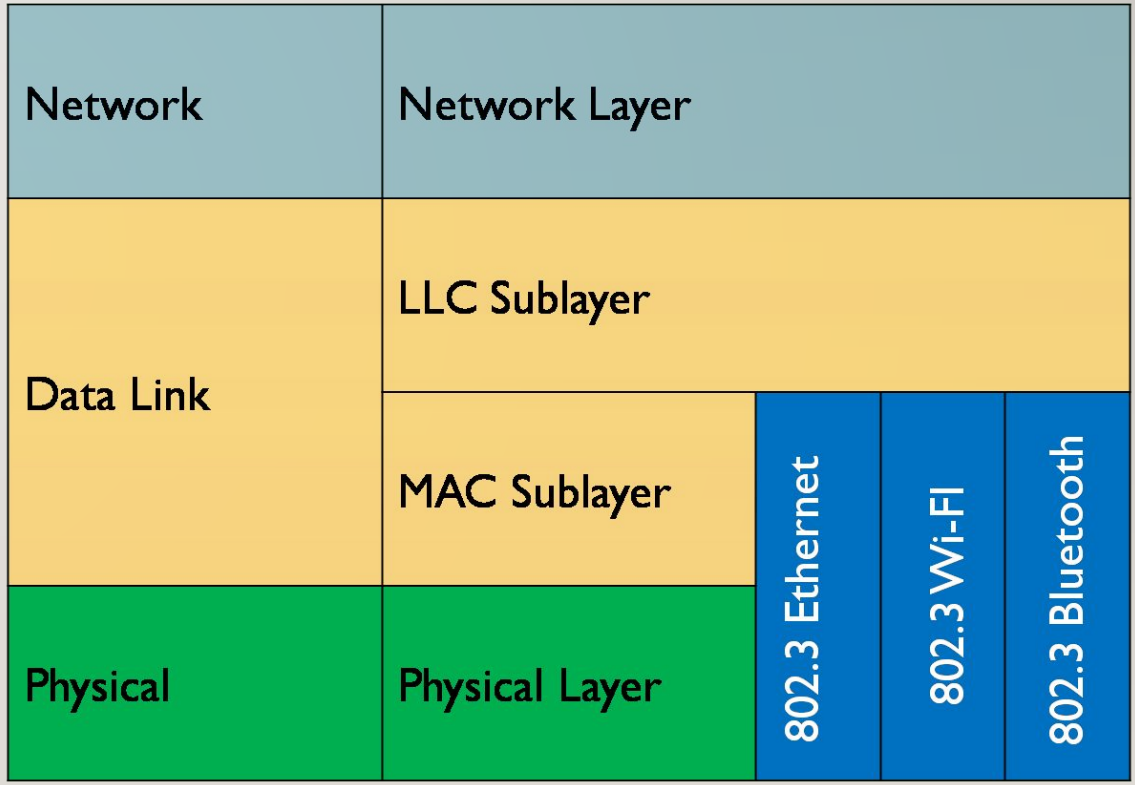
MS. JINCY J FERNANDEZ

ASST. PROF, CSE

RSET

Sublayers of Data link layer

- Logical Link Control sublayer (LLC)
  - Takes data from the network layer
  - Add control information to the data packet to ensure flow control.
- Medium Access Control sublayer (MAC)
  - Lowest sublayer of Data Link layer
  - Two responsibilities
    - Data encapsulation
    - Medium Access Control



Sublayers of Data link layer

- Medium Access Control sublayer (MAC)
  - Data encapsulation
    - Frame assembly before transmission and frame disassembly upon reception of a frame.
    - Adds the header and trailer information to the packet to be transmitted to network layer
  - Medium Access Control
    - Handles access to a shared medium.

Medium Access Control Sublayer

- Network Links can be divided into:
  - Point-to-point connections
  - Broadcast channels(Multi-access channels or random-access channels)
- In a broadcast network, the key issue is how to determine who gets to use the channel when there is competition.
- The protocols used to determine who goes next on a multicast channel belongs to a sublayer of the DLL – Medium Access Control sublayer.
- LANs use multi-access channels.

Channel Allocation Problem

- How to allocate a single broadcast channel among competing users.
- Channel portion of the wireless spectrum or a single wire or optical fiber to which multiple nodes are connected.
- Two types:
  - Static Channel Allocation
  - Dynamic Channel Allocation

Static Channel Allocation

- Capacity of the channel is split among multiple competing users:
  - Frequency Division Multiplexing (FDM).
  - Time Division Multiplexing (TDM).
- FDM
  - With N users, bandwidth is divided into N equal sized portions, with each user being assigned one portion no interference among users.
  - Simple and efficient if there is only small and constant number of users.
  - E. g. FM radio station.
- TDM
  - The time domain is divided into several recurrent *time slots* of fixed length, one for each sub-channel.

Static Channel Allocation

- Problem with FDM and TDM if number of users is too large.
- If the spectrum is cut up into N regions:
  - Fewer than N users are currently interested in communicating then,
    - A part or large piece of valuable spectrum will be wasted.
  - More than N users want to communicate then,
    - Some of them will be denied permission due to lack of bandwidth.

Dynamic Channel Allocation

- Channel allocation is done dynamically.
- Allocation is done based on the current demands of the users.
- Protocols that solve the channel allocation problem dynamically are called Multiple Access Protocols.

Assumptions

- **Independent Traffic.** The model consists of *N* independent **stations/terminals**. Once a frame is generated, the station is blocked and does nothing until the frame has been successfully transmitted.
- **Single Channel Assumption.** A single channel is available for all communication. All stations can transmit on it, and all can receive from it.
- **Observable Collisions.** If two frames are transmitted simultaneously, they overlap in time and the resulting signal is garbled. This event is called a **collision**. All stations can detect collisions. A collided frame must be transmitted again later.

Assumptions

- **Continuous or Slotted Time:**
  - **Continuous Time:** Frame transmission can begin at any instant. There is no master clock dividing time into discrete intervals.
  - **Slotted Time.** Time is divided into discrete intervals (slots). Frame transmissions always begin at the start of a slot. A slot may contain 0, 1, or more frames, corresponding to an idle slot, a successful transmission, or a collision, respectively.

Assumptions

- **Carrier Sense or No Carrier Sense:**
  - **Carrier Sense** Stations can tell if the channel is in use before trying to use it. If the channel is sensed as busy, no station will attempt to use it until it goes idle.
  - **No Carrier Sense.** Stations cannot sense the channel before trying to use it. They just go ahead and transmit. Only later can they determine whether the transmission was successful.

Multiple Access Protocols

- Contention Protocols
  - Resolves collision after it occurs.
  - Executes a collision resolution protocol after each collision.
    - ALOHA
    - Carrier Sense Multiple Access (CSMA)
    - Carrier Sense Multiple Access with Collision Detection (CSMA/CD)
- Collision Free Protocols
  - Ensures that a collision never occurs.
    - Bit –Map Protocol
    - Token passing
    - Binary Countdown

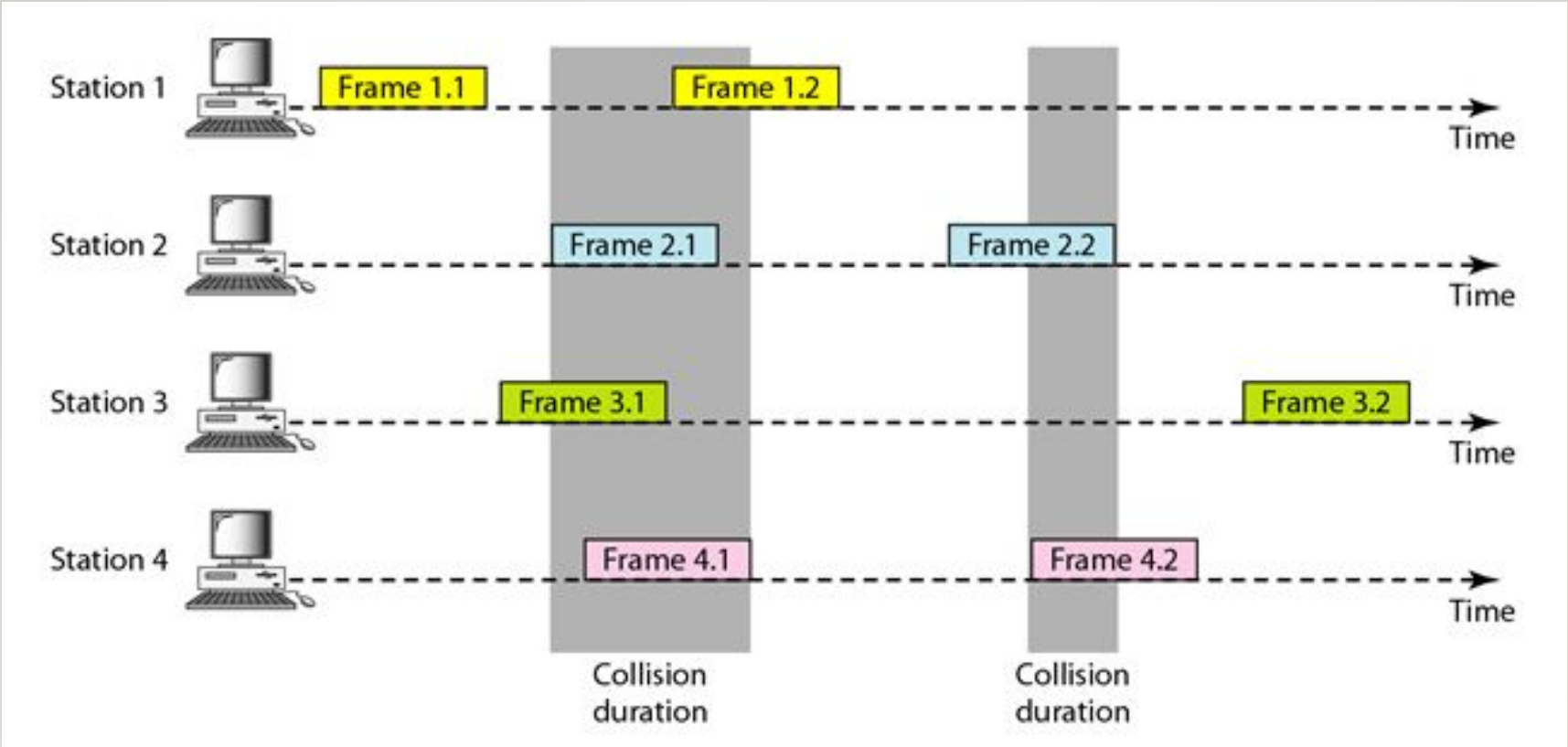
Aloha

- Developed by Norman Abramson in 1970.
- Earliest random-access protocol : any station can send data at any time.
- It was designed for a radio (wireless) LAN, but it can be used on any shared medium.
- Two versions of the protocol:
  - Pure ALOHA
  - Slotted ALOHA

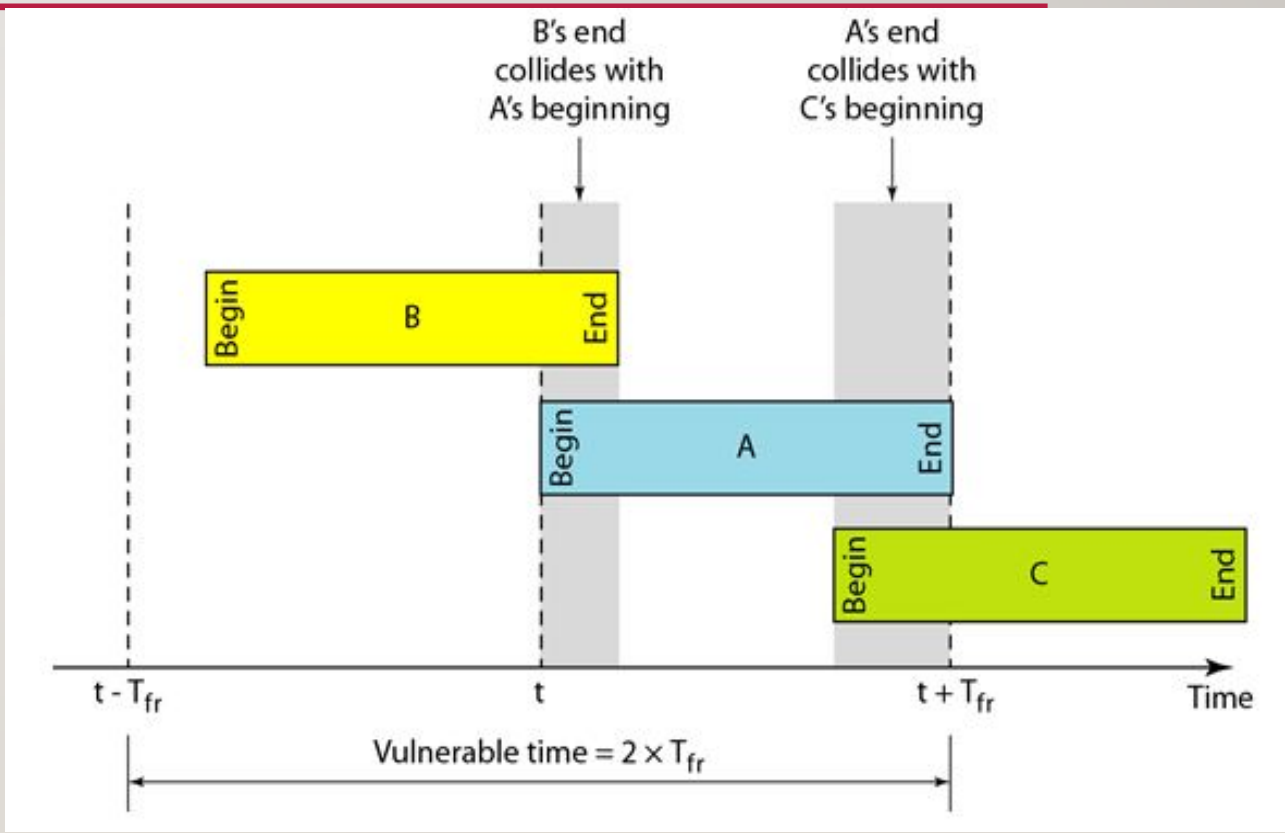
Pure Aloha

- Each user is free to transmit whenever they have data to be sent.
- There are possibilities of collision and colliding frames are destroyed.
- Sender finds whether transmission was successful or has experienced a collision by listening to the channel. (feedback system).
- If the frame is destroyed, the sender waits a random amount of time and sends it again.
- Waiting time must be random.
- Such systems are called contention systems.

Example


Station 1
Frame 1.1
Frame 1.2
Time
Station 2
Frame 2.1
Frame 2.2
Time
Station 3
Frame 3.1
Frame 3.2
Time
Station 4
Frame 4.1
Frame 4.2
Time
Collision duration
Collision duration

Vulnerable Time for a frame



Maximum Throughput Of Pure Aloha

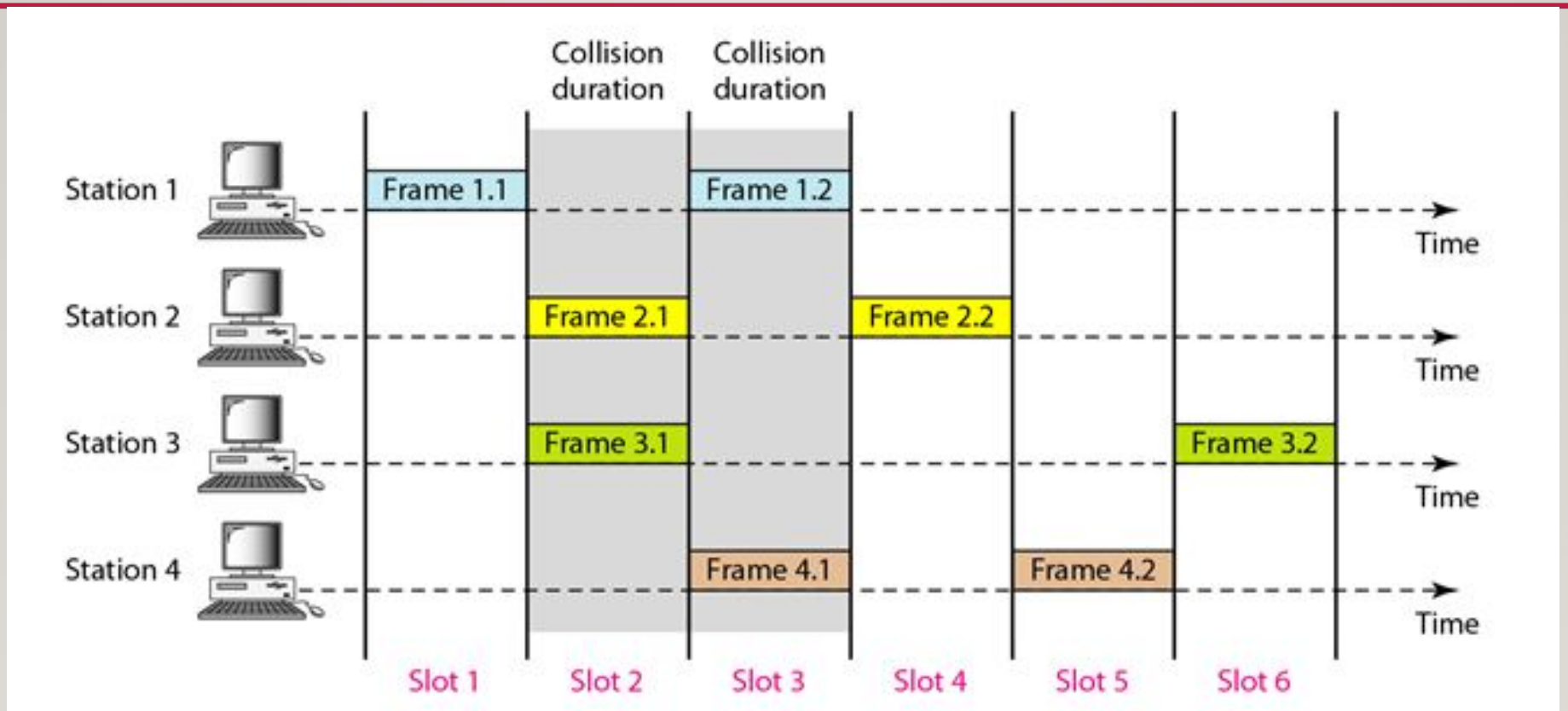
- Let G→ average number of frames generated during one frame transmission time.

- Throughput, S = $G \times e^{-2G}$

Slotted Aloha

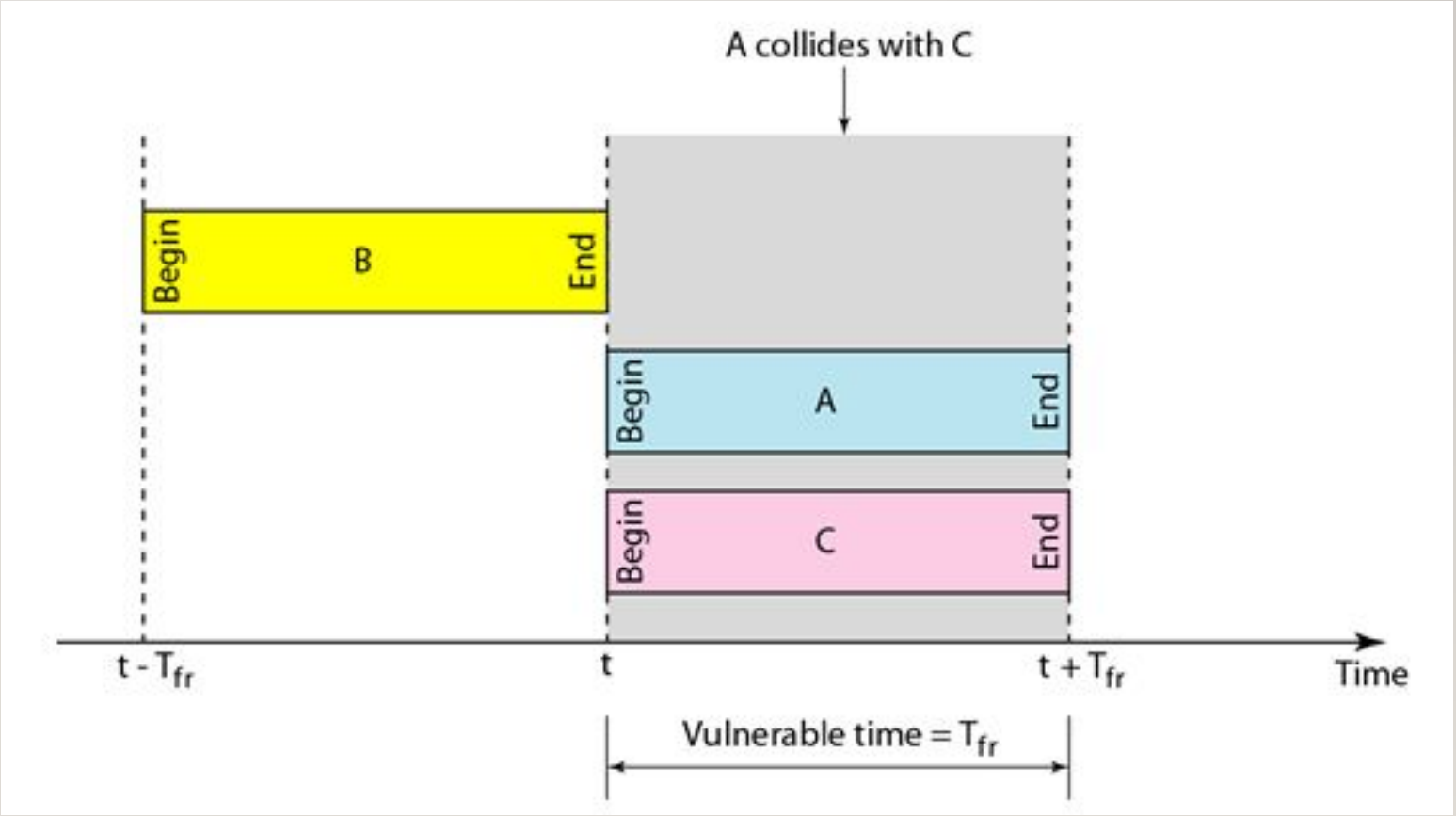
- Capacity of ALOHA doubled - Roberts in 1972.
- Time divided into discrete intervals called slots; each interval corresponds to one frame.
- Users agree on slot boundaries – synchronization needed using clocks.
- A station is not permitted to send whenever it is ready with the data.
- Have to wait for the beginning of the next slot.
- So, this is a discrete ALOHA, previous one was continuous ALOHA.
- Still the possibility of collision if two stations try to send at the beginning of the same slot.

Example


Collision duration
Collision duration
Station 1
Station 2
Station 3
Station 4
Frame 1.1
Frame 1.2
Frame 2.1
Frame 2.2
Frame 3.1
Frame 3.2
Frame 4.1
Frame 4.2
Time
Time
Time
Time
Slot 1
Slot 2
Slot 3
Slot 4
Slot 5
Slot 6

Maximum Throughput Of Slotted Aloha

- Throughput, $S = G \times e^{-G}$

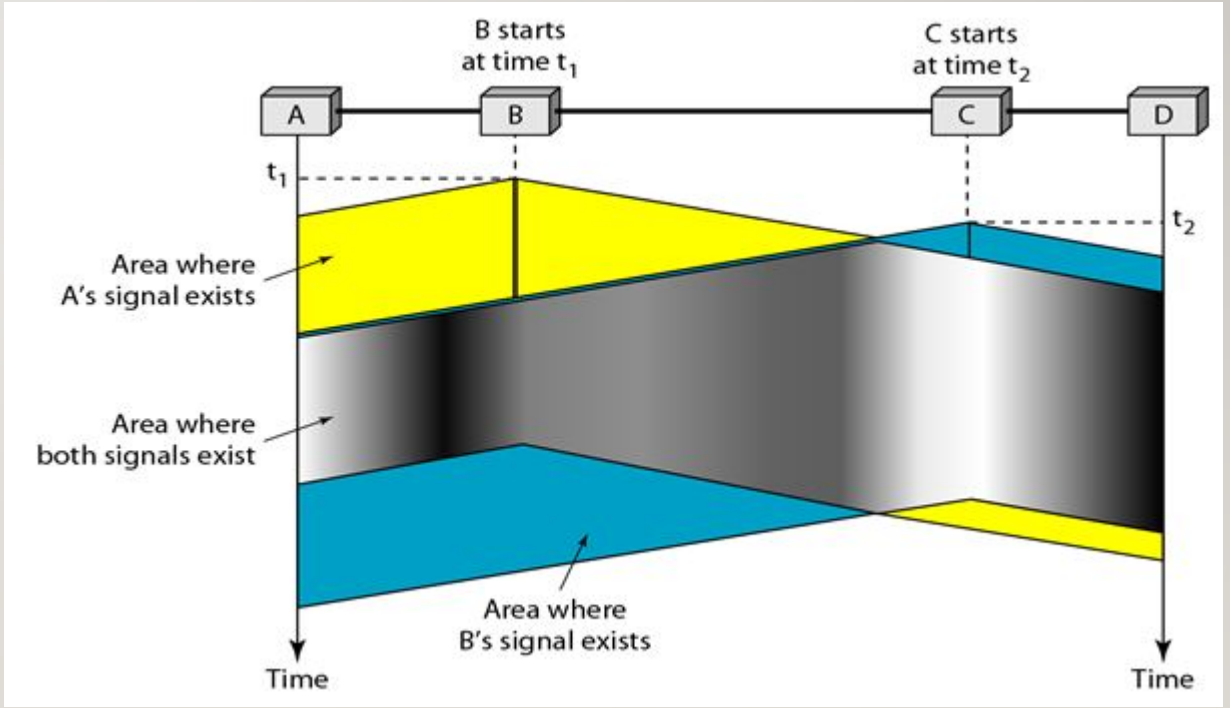


Carrier Sense Multiple Access Protocols

- Protocols in which stations listen for a carrier (transmission) and act accordingly are called Carrier Sense Multiple Access (CSMA) protocols.
- Minimize the chance of collision and increase the performance.
- CSMA is based on the principle “sense before transmit” or “listen before talk”.
- Possibility of collision still exists because of propagation delay: - when a station sends a frame, it still takes time for the first bit to reach every destination and for every station to sense it.

Carrier Sense Multiple Access Protocols

- At time 't1', station B senses the medium and finds it idle, so it sends frame.
- At time 't2' (t2>t1), station C senses the medium and finds it idle because first bit of B is not reached, so C also sends frame.
- Two signals collided and both frames are destroyed.



Carrier Sense Multiple Access Protocols

- Two versions:
  - Persistent and Non-persistent Protocols
    - What should a station do if the channel is busy?
    - What should a station do if the channel is idle?
      - 1-persistent CSMA
      - p-persistent CSMA
      - Non-persistent CSMA
  - CSMA with Collision Detection (CSMA/CD)

1-persistent CSMA

- When a station has data to send, it first listens to the channel.
- If the channel is busy, the station waits until it becomes idle. (continuously checking).
- If the channel is idle, the stations immediately transmits data with a probability of 1.
- If a collision occurs, station waits a random amount of time and starts all over again.
- Propagation delay is an issue here. Longer the delay worse the performance of the protocol.
- Better performance than ALOHA.

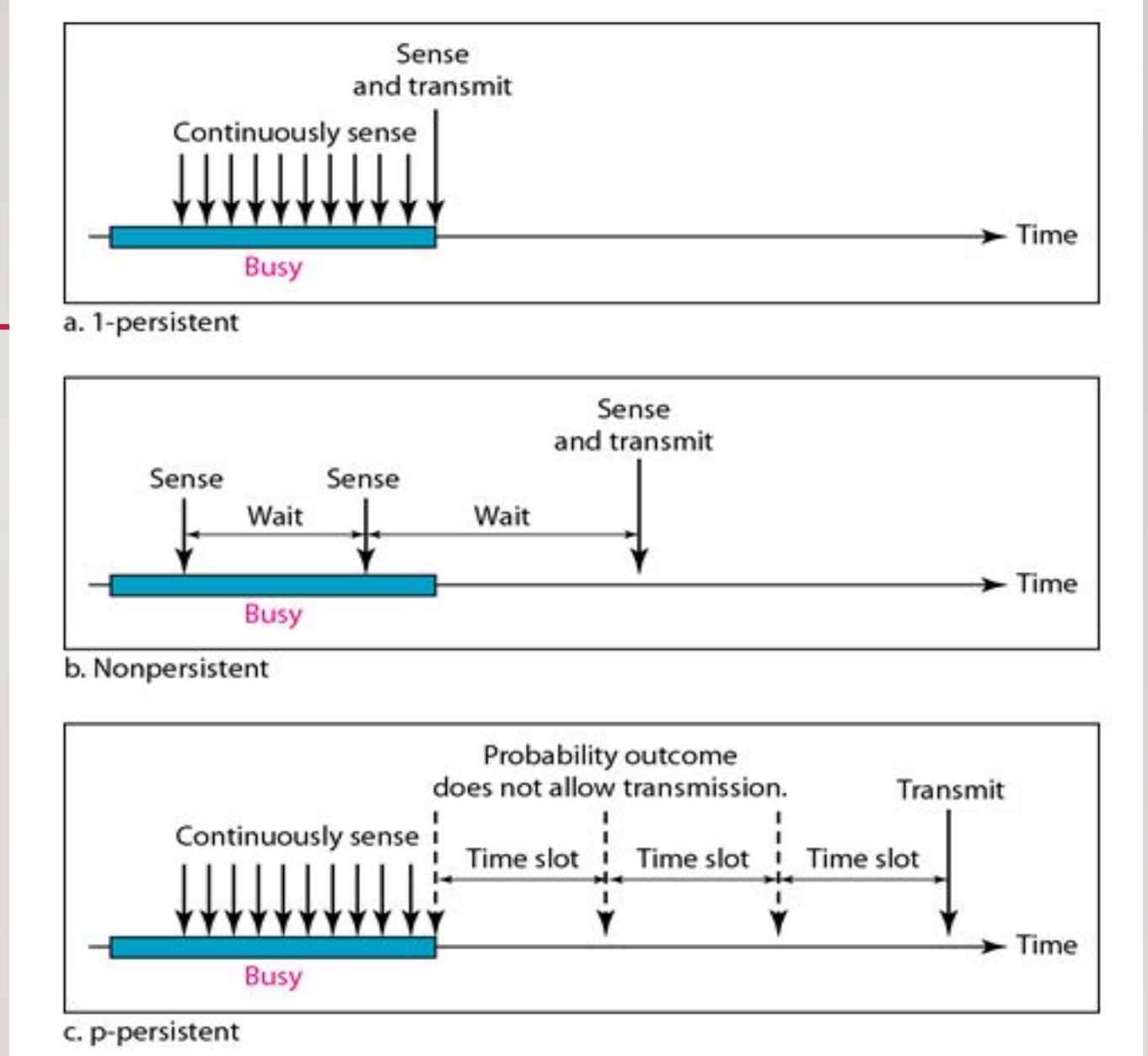
Non-persistent CSMA

- A station senses the channel when it wants to send a frame. If idle, send the frame. If the channel is already in use, the station does not continually sense it.
- Instead, it waits a random period of time and then repeats the algorithm.
- This algorithm leads to better channel utilization but longer delays than 1-persistent CSMA.

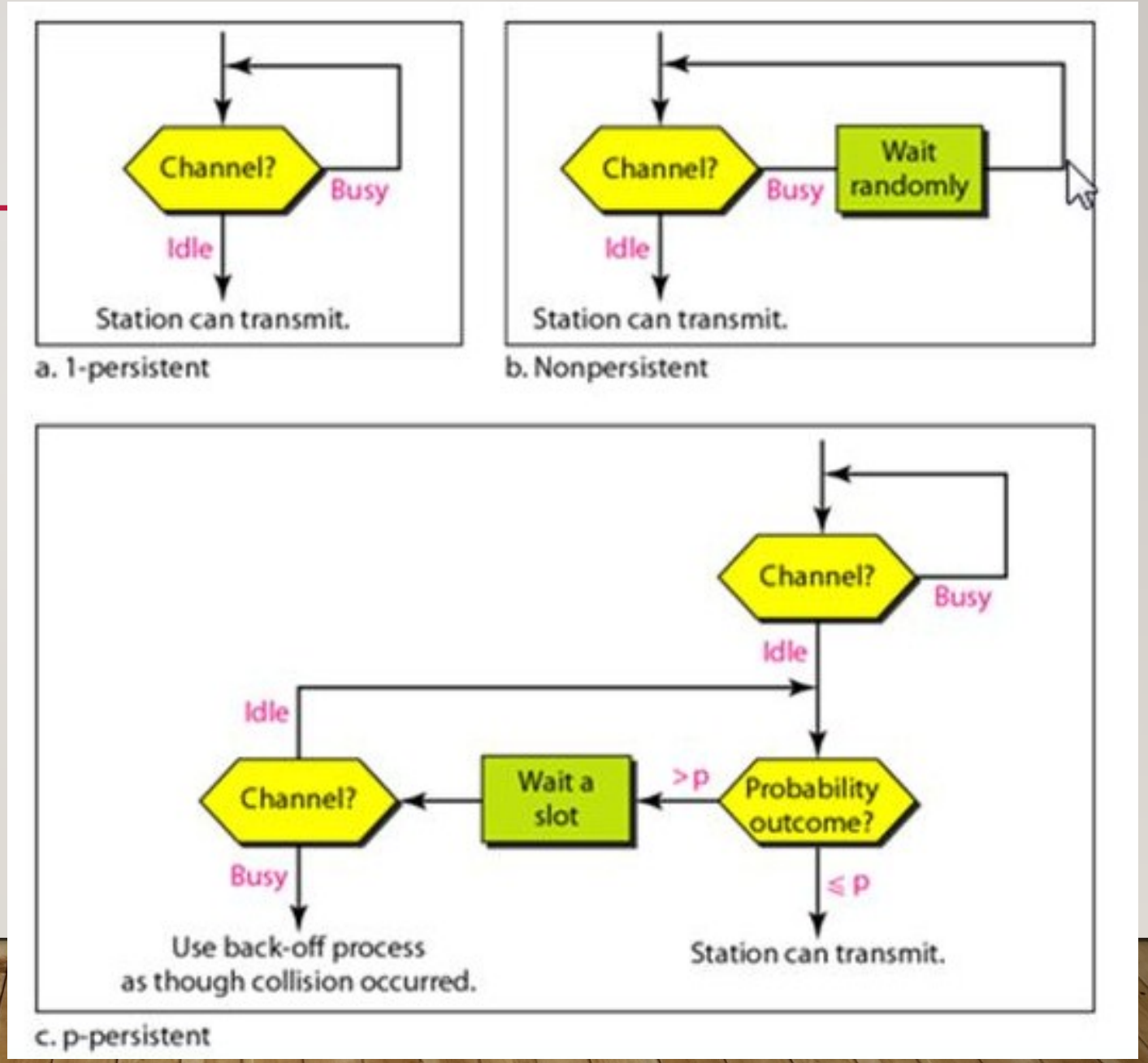
P-persistent CSMA

- 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. (only in its available slot).*
- It defers with *a* probability $q = 1 - p$ 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.
- So, station has to check for channel and time slot.

Example


Sense
and transmit
Continuously sense
Busy
Time
a. 1-persistent
Sense
and transmit
Sense
Sense
Wait
Wait
Busy
Time
b. Nonpersistent
Probability outcome
does not allow transmission.
Transmit
Continuously sense
Time slot
Time slot
Time slot
Busy
Time
c. p-persistent

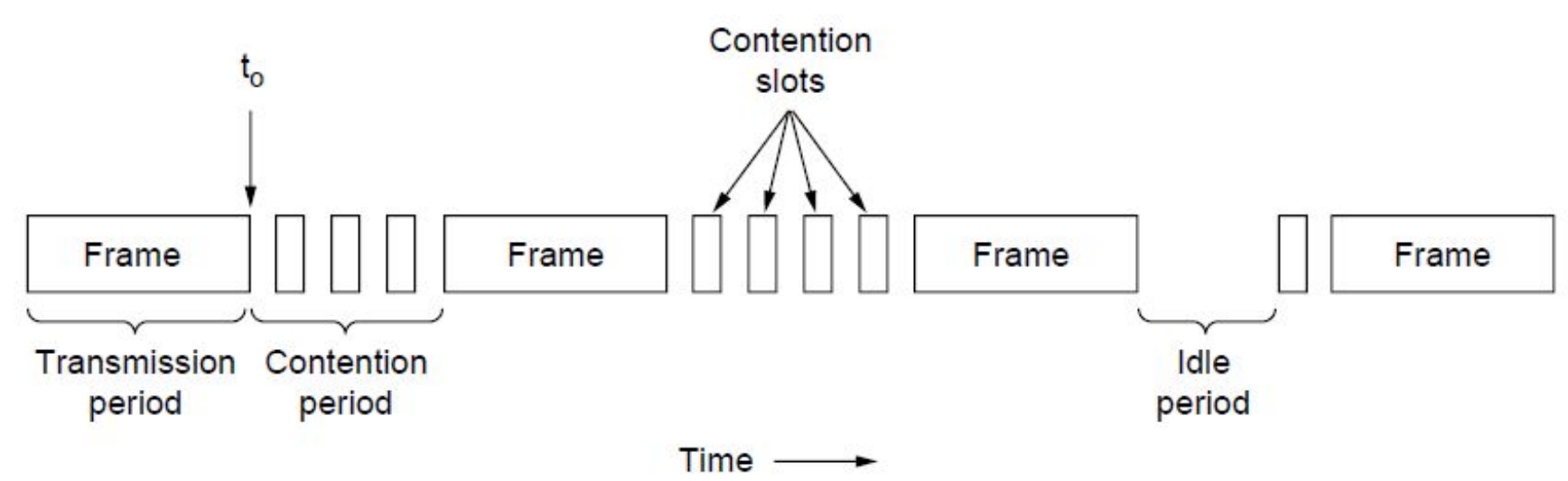
Flow Diagram



CSMA With Collision Detection

- Protocols that abort transmissions as soon as they detect collisions CSMA with Collision Detection (CSMA/CD).
- Quickly terminating damaged frames save time and bandwidth.
- This protocol is a basis of classical Ethernet LAN.
- Uses power or pulse width of the received signals to detect collision: power or pulse width of the transmitted is better than that of received signal.
- No ACK used; It checks for the successful and unsuccessful transmissions through collision signals.

CSMA With Collision Detection



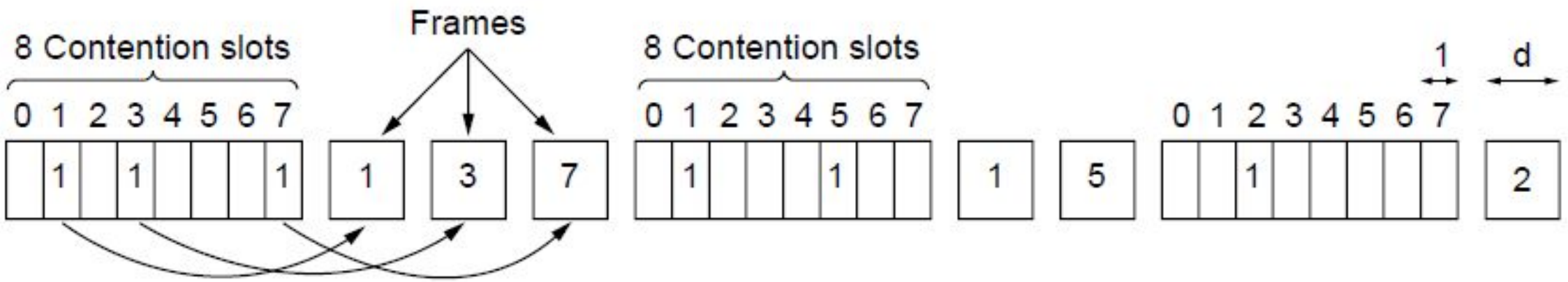
- The model for CSMA/CD consists of alternating contention and transmission periods, with idle periods occurring when all the stations are quiet.
- Contention period is the minimum time a host must wait to make sure that no other host is transmitting.

Collision – Free Protocols

- Collisions may occur during contention period in CSMA/CD.
- Also, it is not universally applicable.
- So go for protocols that ensure collisions don't occur at all.
- Assumptions:
  - N – Stations each with a unique address 0- (N-1).
  - Propagation delay is assumed to be negligible.
  - Some stations may be inactive throughout.
- Basic question-Which station gets the channel (e.g., the right to transmit) after a successful transmission?

Bit-map Protocol

- Called reservation protocol.
- Each contention period (reservation frame) consists of N slots.
- If station 0 has a frame to send, it transmits a 1 bit during the zeroth slot.
- No other station transmits during this slot.
- Station j may announce that it has a frame to send by inserting a 1 bit into slot j.
- After all N slots have passed by, each station has complete knowledge of which stations wish to transmit.
- The stations then begin transmitting frames in numerical order.



The basic bit-map protocol

Since everyone agrees on who goes next, there will never be any collisions.

Token Passing

- A small message called token is passed from one station to the next in a predefined order.
- Token represents permission to send.
- If a station has a frame queued for transmission when it gets the token, it can send that frame before it passes the token to the next station.

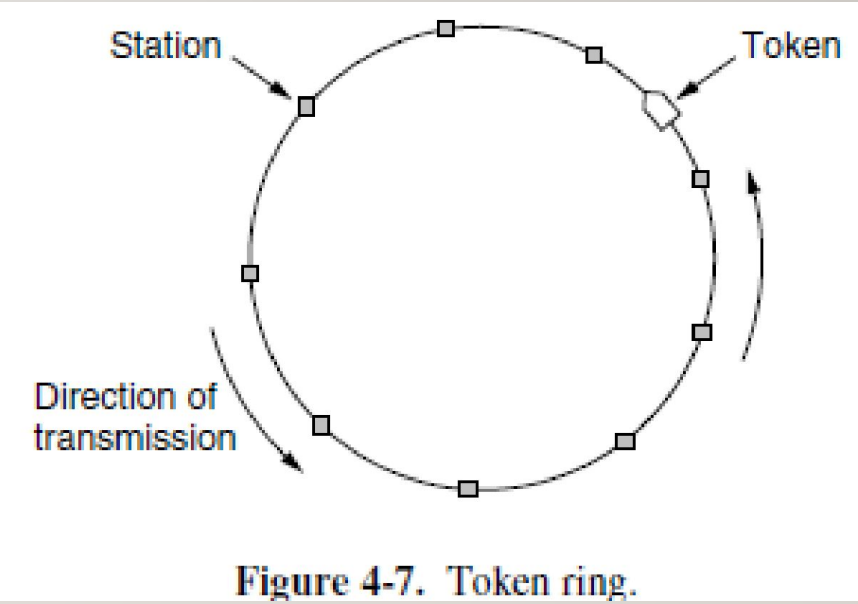


Figure 4-7. Token ring.

- If it has no queued frame, it simply passes the token..

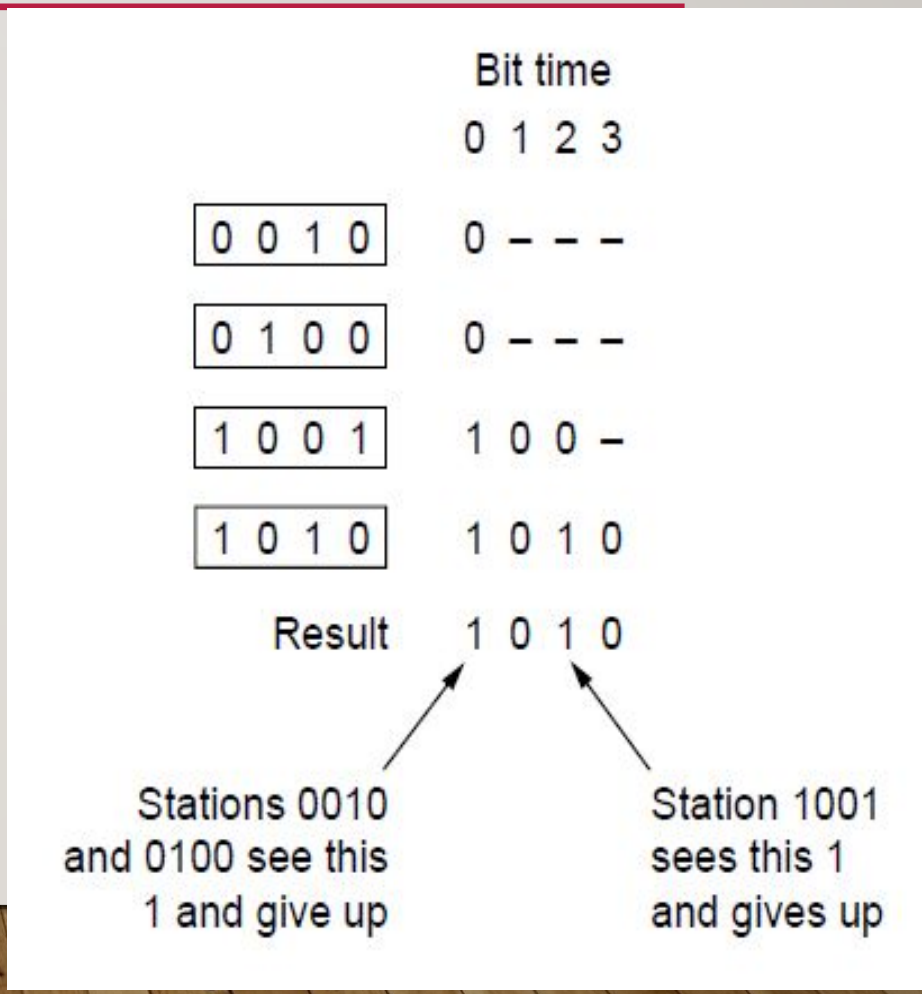
After sending a frame a station must wait for all N stations to pass the token to its neighbor and all N-1 stations to send a frame, if they have one, before it gets the next chance.

Binary Countdown Protocol

- Issue with bitmap protocol is its overhead. Does not scale well for large networks.
- Issue with token passing is starvation.
- Better solution- go for binary station addresses and priorities.
- A station that wants to use the channel broadcasts its address (assumed to be the same length) as a binary bit string starting with the high-order bit.
- The bits in each address position from different stations are BOOLEAN OR-ed together by the channel when they are sent at the same time.
- Arbitration rule: As soon as a station sees that a high-ordered bit position that is 0 in its address has been overwritten with 1 it gives up.

Binary Countdown Protocol

- If stations 0010,0100,1001,1010 are trying to get the channel, in the first bit time, stations transmits 0, 0, 1, 1. Boolean OR results in 1.
- So, stations 0010, 0100 gives up and wait.
- Stations 1001 and 1010 continue and check next bit and so on.
- It has the property that higher numbered stations have a higher priority.



Performance Measures

- Two main measures:
  - Delay
  - Channel efficiency
- Under conditions of light load:
  - Contention protocols preferable – low delay only.
- Under conditions of high load:
  - Collision free protocols preferable – better channel efficiency.
- Combine these two protocols into a hybrid one- limited contention protocols.

Wireless LAN Protocols

- Each radio transmitter has some fixed range.
- Its range is represented by an ideal circular coverage region.
- Within that region station can sense and receive the station's transmission.
- Using CSMA for channel access is possible.
- Carrier Sense with Collision Avoidance
- Listen for other transmissions and transmit only if no one else is doing so.

THANK YOU!!!

Pure Aloha

- Q1. A pure ALOHA network transmits 200-bit frames on a shared channel of 200Kbps. What is the throughput if the system produces 1000 frames per second?

  - Frame transmission time =200/200Kbps =1ms
  - 1000 frames per second= 1 frame per milliseconds, G=1.
  - S=

Ans: 135