COMPUTER NETWORKS

Subject Code: 21CSC302J

UNIT - 4

Presented by:

Mr. Parbhat Gupta

Assistant Professor

Computer Science and Engineering Department

TOPICS COVERED

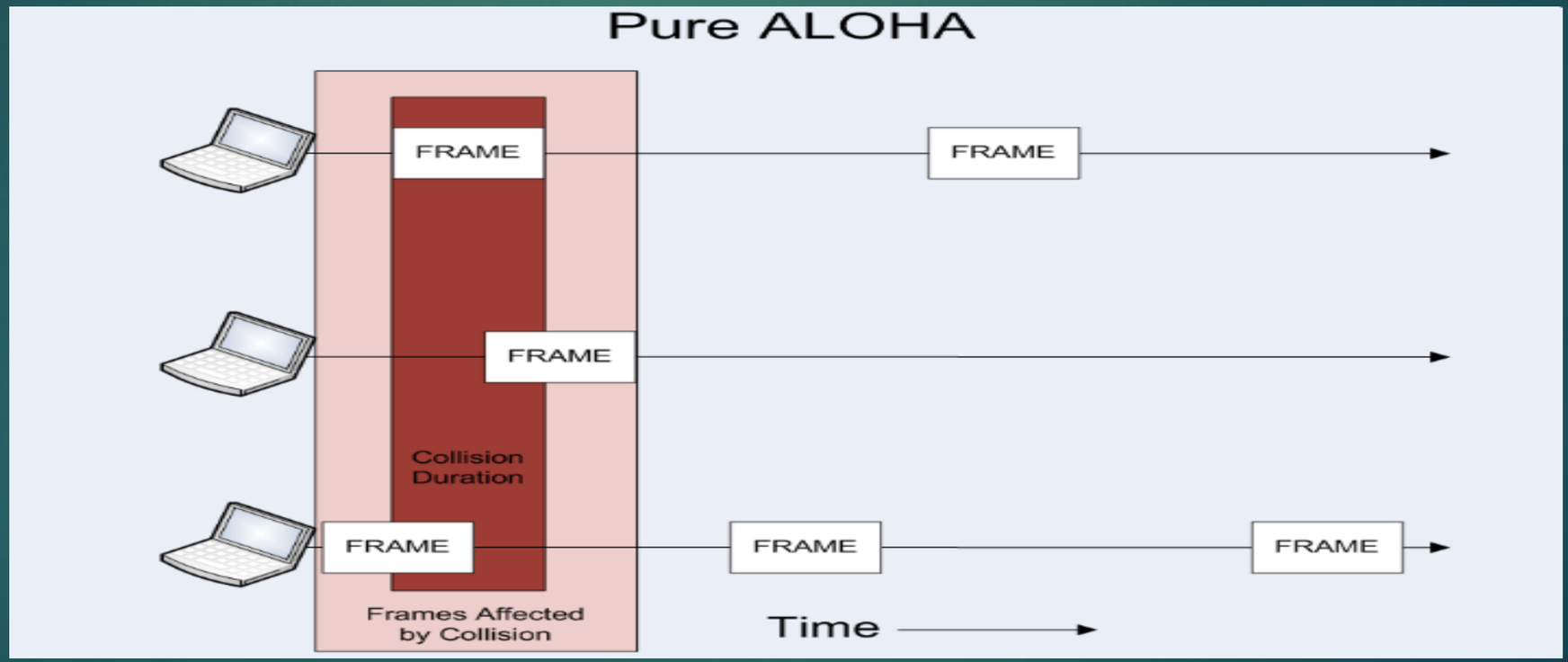
- Aloha
- CSMA/CD
- Ethernet
- Token Ring
- Flow Control : Stop And Wait
- Error Control
- Stop and Wait ARQ
- Sliding Window ARQ
- Error Detection
- Parity Check
- Checksum
- CRC
- Error Correction
- Hamming codes
- Data-Link Layer Protocols
- HDLC
- PPP

Aloha

- A multiple access protocol for transmission of data via a shared network channel.
- Operates in the medium access control sub layer of the OSI model.
- Each node or station transmits a frame without trying to detect whether the transmission channel is idle or busy.
- If the channel is idle, then the frames will be successfully transmitted. If two frames attempt to occupy the channel simultaneously, collision of frames will occur, and the frames will be discarded. These stations may choose to retransmit the corrupted frames repeatedly until successful transmission occurs.

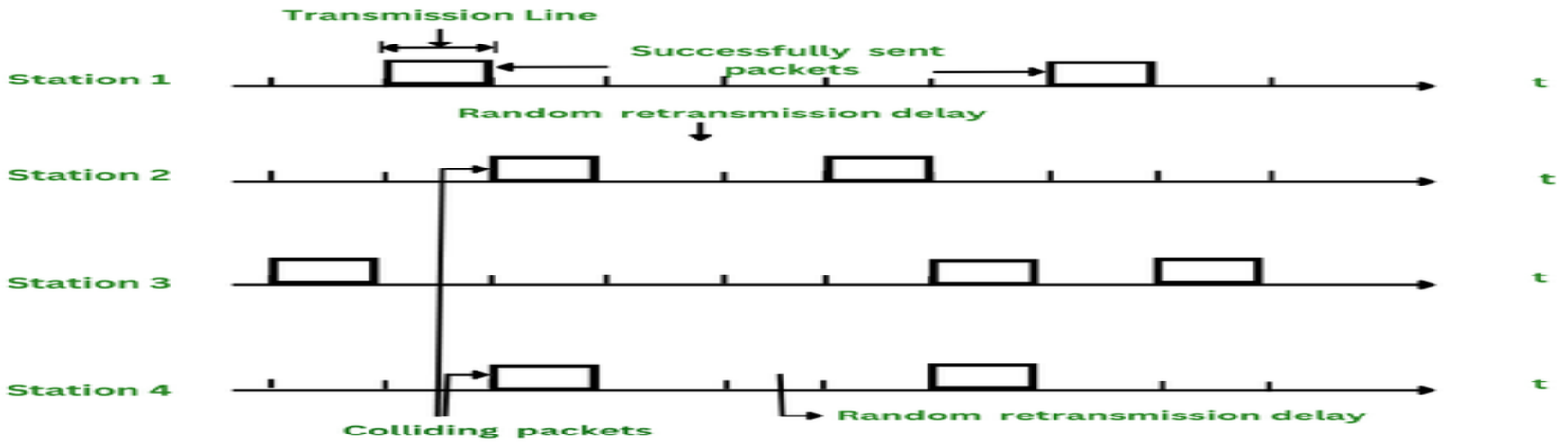
Pure Aloha

- In pure ALOHA, the time of transmission is continuous. Whenever a station has an available frame, it sends the frame. If there is collision and the frame is destroyed, the sender waits for a random amount of time before retransmitting it.



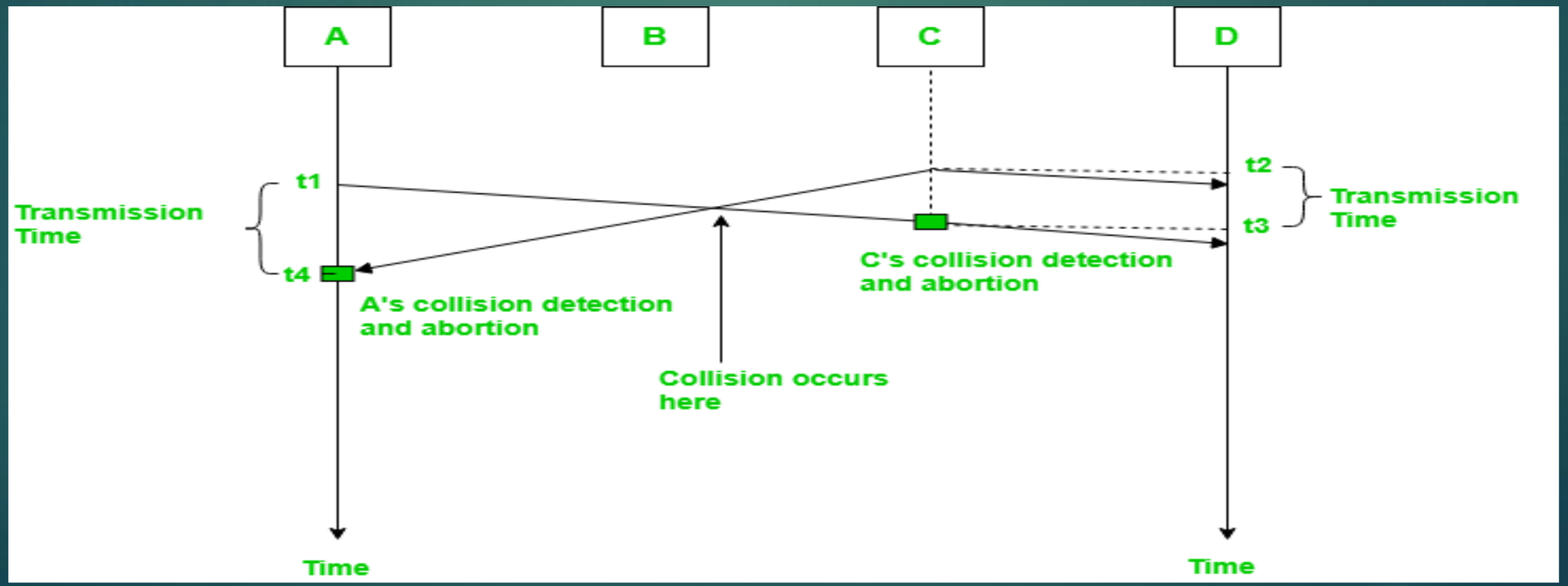
Slotted Aloha

- Slotted ALOHA reduces the number of collisions and doubles the capacity of pure ALOHA. The shared channel is divided into a number of discrete time intervals called slots. A station can transmit only at the beginning of each slot. However, there can still be collisions if more than one station tries to transmit at the beginning of the same time slot.



CSMA/CD

- Stands for Carrier Sense Multiple Access/ Collision Detection
- A media access control method that was widely used in Early Ethernet technology/LANs.
- Now a Days Ethernet is Full Duplex and Topology is either Star (connected via Switch or Router) or Point to Point (Direct Connection). Hence CSMA/CD is not used but they are still supported though.
- A technique where different stations that follow this protocol agree on some terms and collision detection measures for effective transmission. This protocol decides which station will transmit when so that data reaches the destination without corruption.



CSMA/CA

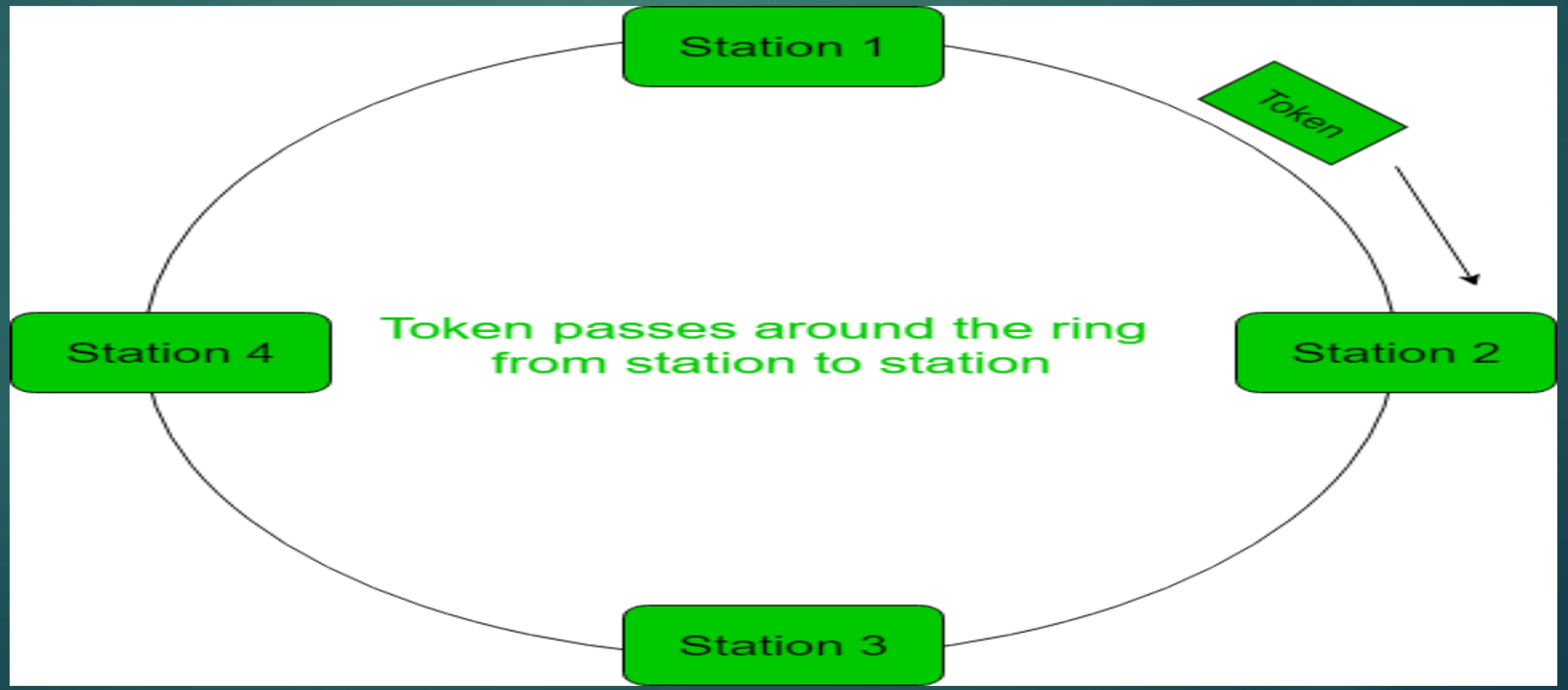
- Stands for Carrier Sense Multiple Access with Collision Avoidance.
- A network protocol that uses to avoid a collision rather than allowing it to occur, and it does not deal with the recovery of packets after a collision.
- Similar to the CSMA CD protocol that operates in the media access control layer. In CSMA CA, whenever a station sends a data frame to a channel, it checks whether it is in use. If the shared channel is busy, the station waits until the channel enters idle mode. Hence, we can say that it reduces the chances of collisions and makes better use of the medium to send data packets more efficiently.

Ethernet

- Most widely used LAN technology
- Defined under IEEE standards 802.3.
- The reason behind its wide usability is that Ethernet is easy to understand, implement, and maintain, and allows low-cost network implementation.
- Offers flexibility in terms of the topologies that are allowed.
- Generally, uses a bus topology.
- Operates in two layers of the OSI model, the physical layer and the data link layer.
- For Ethernet, the protocol data unit is a frame since we mainly deal with DLLs. In order to handle collisions, the Access control mechanism used in Ethernet is CSMA/CD.

Token Ring

- A computer network topology and access method to connect devices in a physical ring or loop.
- In a token ring network, data can travel in a unidirectional or bidirectional manner around the ring, and devices are connected to the network in a sequential fashion.
- This topology contrasts other network topologies, such as Ethernet, which use a bus or star configuration.



Flow Control : Stop And Wait

- A simple protocol used for transmitting data between two devices over a communication channel. In this protocol, the sender sends a packet of data to the receiver and then waits for the receiver to acknowledge the packet before sending the next packet. The receiver sends an acknowledgement to the sender indicating that the packet has been received and is error-free.
- Features :
  - The sender transmits one packet at a time and waits for an acknowledgement before sending the next packet.
  - The receiver sends an acknowledgement for each packet received, indicating whether it is a duplicate or a new packet.
  - It is a simple and easy-to-implement protocol.
  - It has low efficiency compared to sliding window protocols as it requires a lot of time to wait for an acknowledgement for each packet.
  - It is ideal for situations where the transmission rate is low or the network is reliable.

Sliding Window

- A more efficient protocol for data transmission than the Stop-and-Wait protocol. It uses a window size to control the number of packets that can be transmitted without acknowledgement. The sender can transmit multiple packets within the window size before waiting for an acknowledgement from the receiver.
- Features:
  - The sender can transmit multiple packets without waiting for an acknowledgement for each packet.
  - The receiver sends a cumulative acknowledgement for a sequence of packets, indicating the last correctly received packet.
  - It uses a sliding window mechanism to allow the sender to transmit a group of packets at once before receiving an acknowledgement for the first packet.
  - It has a higher efficiency compared to stop and wait protocol as it allows for simultaneous transmission and acknowledgement of multiple packets.
  - It requires more processing power and memory to implement than stop and wait protocol.
  - There are two types of sliding window protocols – Go-Back-N and Selective Repeat – each with their own set of features and advantages.

Error Control

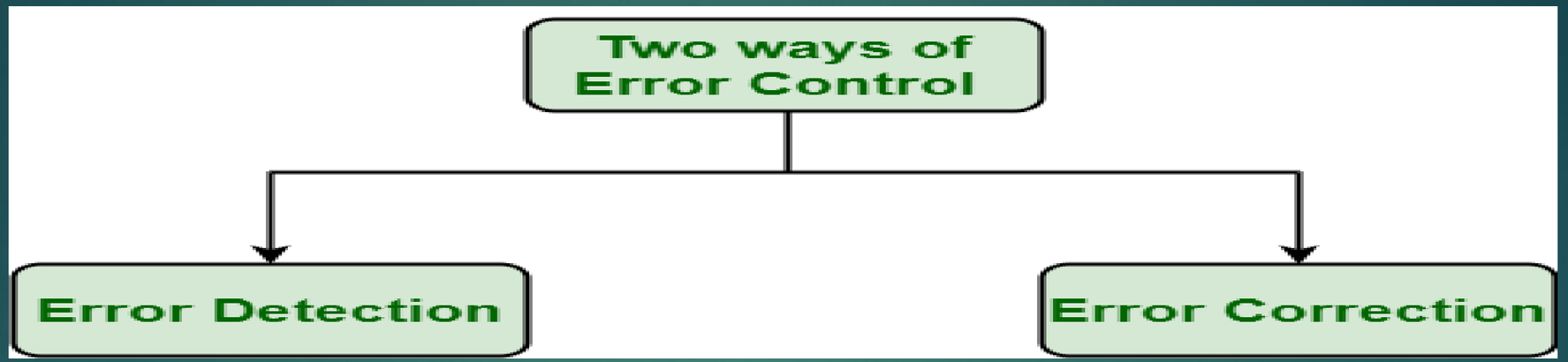
- How do we make sure that all frames are eventually delivered to the network layer at the destination and in the proper order?
- Provide sender with some acknowledgement about what is happening with the receiver
- Sender could wait for acknowledgement

Disadvantages

- If a frame vanishes, the receiver will not send an acknowledgement thus, sender will wait forever
- Dealt with by timers and sequence numbers – important part of DLL
- Sender transmits a frame, starts a timer.
- Timer set to expire after interval long enough for frame to reach destination, be processed, and have acknowledgement sent to sender
- Is a danger of frame being transmitted several times, however dealt with by assigning sequence numbers to outgoing frames, so that receiver can distinguish retransmissions from originals.

Ways of doing Error Control

There are basically two ways of doing Error control as given below :



- Error Detection : Error detection, as the name suggests, simply means detection or identification of errors. These errors may occur due to noise or any other impairments during transmission from transmitter to the receiver, in communication system. It is a class of techniques for detecting garbled i.e. unclear and distorted data or messages.
- Error Correction : Error correction, as the name suggests, simply means correction or solving or fixing of errors. It simply means reconstruction and rehabilitation of original data that is error-free. But error correction method is very costly and very hard.

Stop and Wait ARQ

Automatic Repeat Request (ARQ), an error control method, is incorporated with stop and wait flow control protocol

- If error is detected by receiver, it discards the frame and send a negative ACK (NAK), causing sender to re-send the frame.
- In case a frame never got to receiver, sender has a timer: each time a frame is sent, timer is set ! If no ACK or NAK is received during timeout period, it re-sends the frame.
- Timer introduces a problem: Suppose timeout and sender retransmits a frame but receiver actually received the previous transmission ! receiver has duplicated copies.
- To avoid receiving and accepting two copies of same frame, frames and ACKs are alternatively labeled 0 or 1: ACK0 for frame 1, ACK1 for frame 0.

Design of the Stop-and-Wait ARQ Protocol

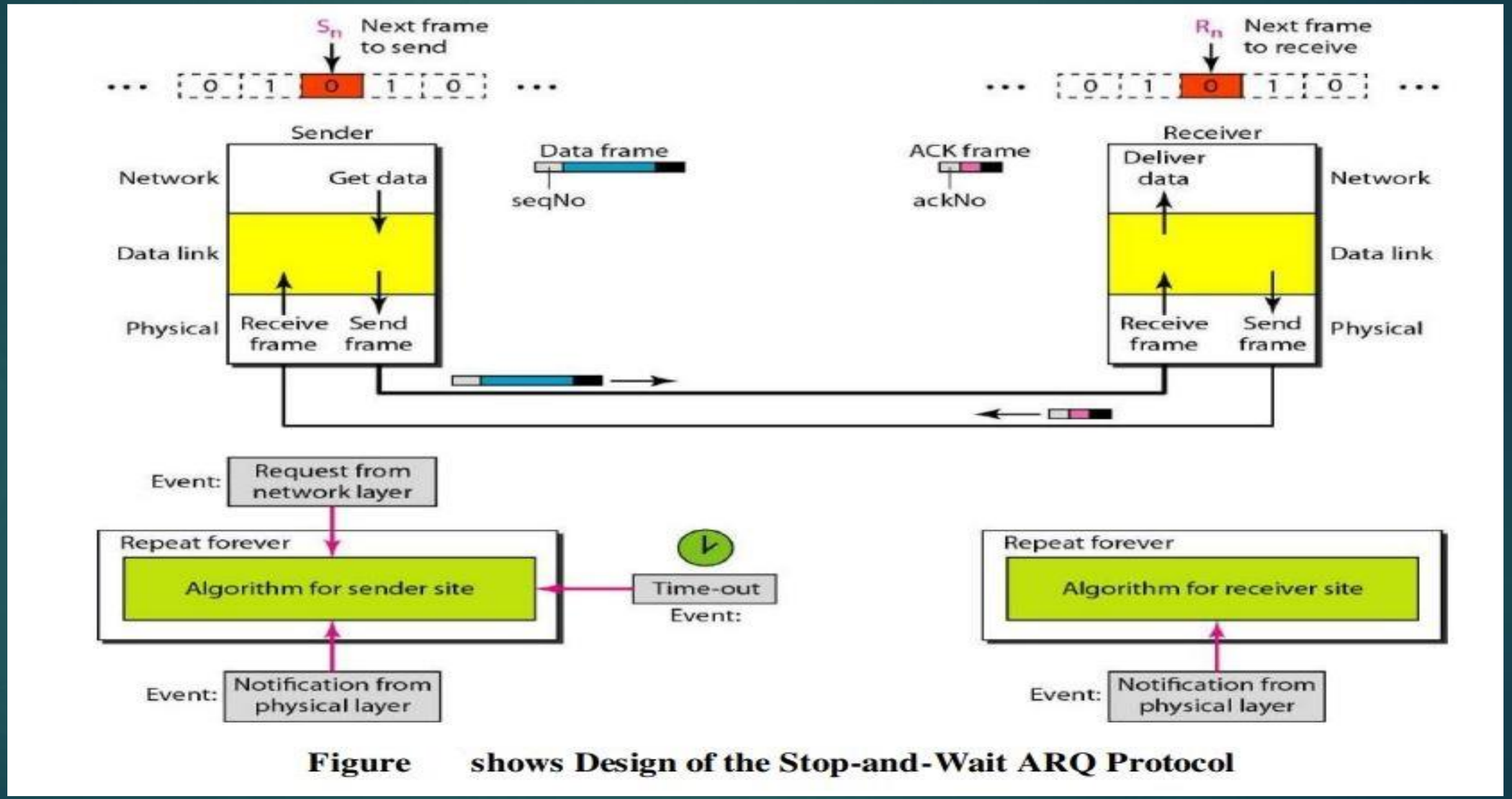


Figure shows Design of the Stop-and-Wait ARQ Protocol

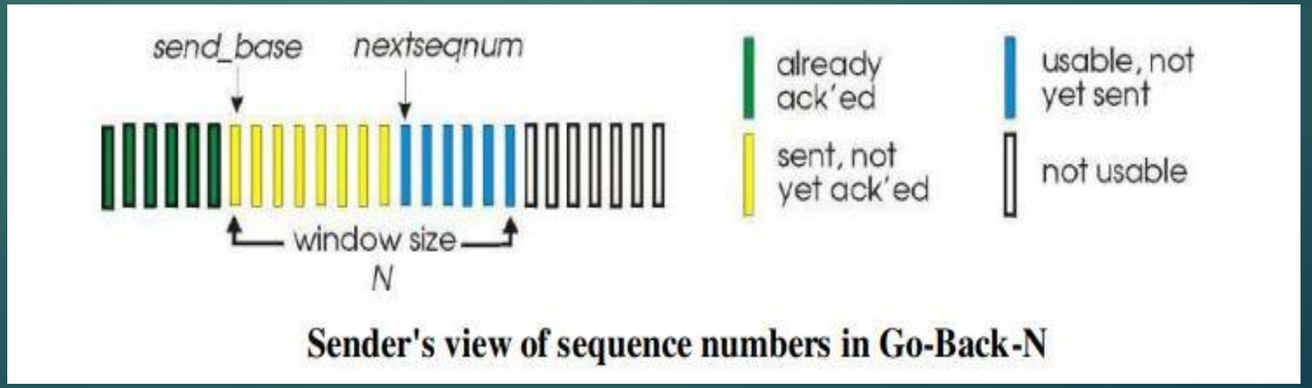
Sliding Window ARQ

- A One-Bit Sliding Window Protocol

The starting machine fetches the first packet from its network layer, builds a frame from it, and sends it. When this (or any) frame arrives, the receiving data link layer checks to see if it is a duplicate, just as in protocol 3. If the frame is the one expected, it is passed to the network layer and the receiver's window is slid up. The acknowledgement field contains the number of the last frame received without error.

- A Protocol Using Go-Back-N

In a Go-Back-N (GBN) protocol, the sender is allowed to transmit multiple packets (when available) without waiting for an acknowledgment but is constrained to have no more than some maximum allowable number, N, of unacknowledged packets in the pipeline.



Sender's view of sequence numbers in Go-Back-N

Cont.

The above figure shows the sender's view of the range of sequence numbers in a GBN protocol. If we define base to be the sequence number of the oldest unacknowledged packet and next seq num to be the smallest unused sequence number (i.e., the sequence number of the next packet to be sent), then four intervals in the range of sequence numbers can be identified. The range of permissible sequence numbers for transmitted but not-yet-acknowledged packets can be viewed as a ``window" of size N over the range of sequence numbers. N is often referred to as the window size and the GBN protocol itself as a sliding window protocol.

- A Protocol Using Selective Repeat

Selective Repeat (SR) protocols avoid unnecessary retransmissions by having the sender retransmit only those packets that it suspects were received in error (i.e., were lost or corrupted) at the receiver. This individual, as-needed, retransmission will require that the receiver individually acknowledge correctly received packets. A window size of N will again be used to limit the number of outstanding, unacknowledged packets in the pipeline. The SR receiver will acknowledge a correctly received packet whether or not it is in-order. Out-of order packets are buffered until any missing packets (i.e., packets with lower sequence numbers) are received, at which point a batch of packets can be delivered in-order to the upper layer. Figure receiver itemizes the various actions taken by the SR receiver.

Error Detection

- Network designers have developed two basics strategies for dealing with errors. One way is to include enough redundant information along with each block of data sent, to enable the receiver to deduce what the transmitted data must have been .The other way is to include only enough redundancy to allow the receiver to deduce that an error occurred, but not which error, and have it request a retransmission. The former strategy uses Error – correcting codes and the latter uses Error-detecting codes. Error-correcting codes are widely used on wireless links, which are notoriously noisy and error prone when compared to copper wire or optical fibers. Without error-correcting codes, it would be hard to get anything through. However, over copper wire or fiber, the error rate is much lower, so error detection and retransmission is usually more efficient there for dealing with the occasional error. As a simple example, consider a channel on which errors are isolated and the error rate is 10-6 per bit. Let the block size be 1000 bits. To provide error correction for 1000-bit blocks, 10 check bits are needed; a megabit of data would require 10,000 check bits. To merely detect a block with a single 1-bit error, one parity bit per block will suffice. Once every 1000 blocks, an extra block (1001 bits) will have to be transmitted. The total overhead for the error detection + retransmission method is only 2001 bits per megabit of data, versus 10,000 bits for a Hamming code. If a single parity bit is added to a block and the block is badly garbled by a long burst error, the probability that the error will be detected is only 0.5, which is hardly acceptable.

Cont.

- The Error – correcting and Error- detecting methods are
  - PARITY CHECK METHOD
  - CHECK SUM METHOD
  - CRC METHOD
  - HAMMING CODE METHOD

Parity Check Method

- Appends a parity bit to the end of each word in the frame
- Even parity is used for asynchronous Transmission
- Odd parity is used for synchronous Transmission

Ex 1.	Character code	even parity	odd parity
	1100100	1100100 1	1100100 0
2.	0011000	0011000 0	0011000 1

IF one bit or any odd no bits is erroneously inverted during Transmission, the Receiver will detect an error. How ever if two or even no of bits are inverted an undetected error occurs.

Examples

Ex 3.

The Transmitted data is 10011010. The received data is 11011010.

- Let both the transmitter and receiver are agreed on EVEN parity.
- Now an error will be detected, since the no of ones received are ODD.

Ex 4.

The Transmitted data is 10011010. The received data is 01011010.

- The received data is wrong even though the no of ones are EVEN.
- Science two bits are inverted error can‘t be detected.

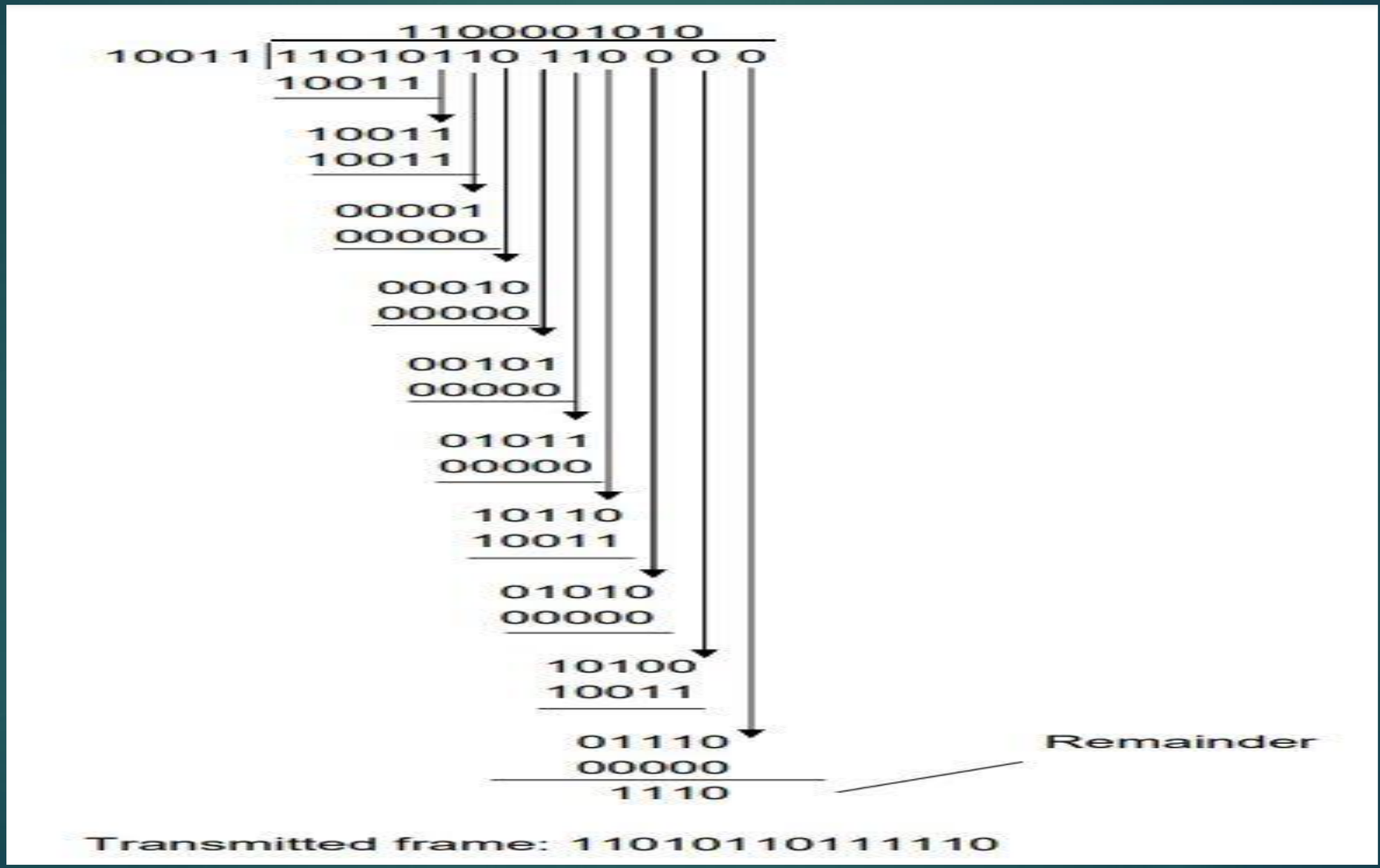
CRC

- The frame is expressed in the form of a Polynomial F(x) 0 1 1 1 1 1 1 0
- Both the sender and receiver will agree upon a generator polynomial G(x) in advance.
- Let ‘r’ be the degree of G(x).Append ‘r’ zero bits to the lower – order end of frame now it contains m+r bits.
- Divide the bit string by G(x) using Mod 2 operation.
- Transmitted frame [T(x)] = frame + remainder
- Divide T(x) by G(x) at the receiver end. If the result is a zero, then the frame is transmitted correctly.

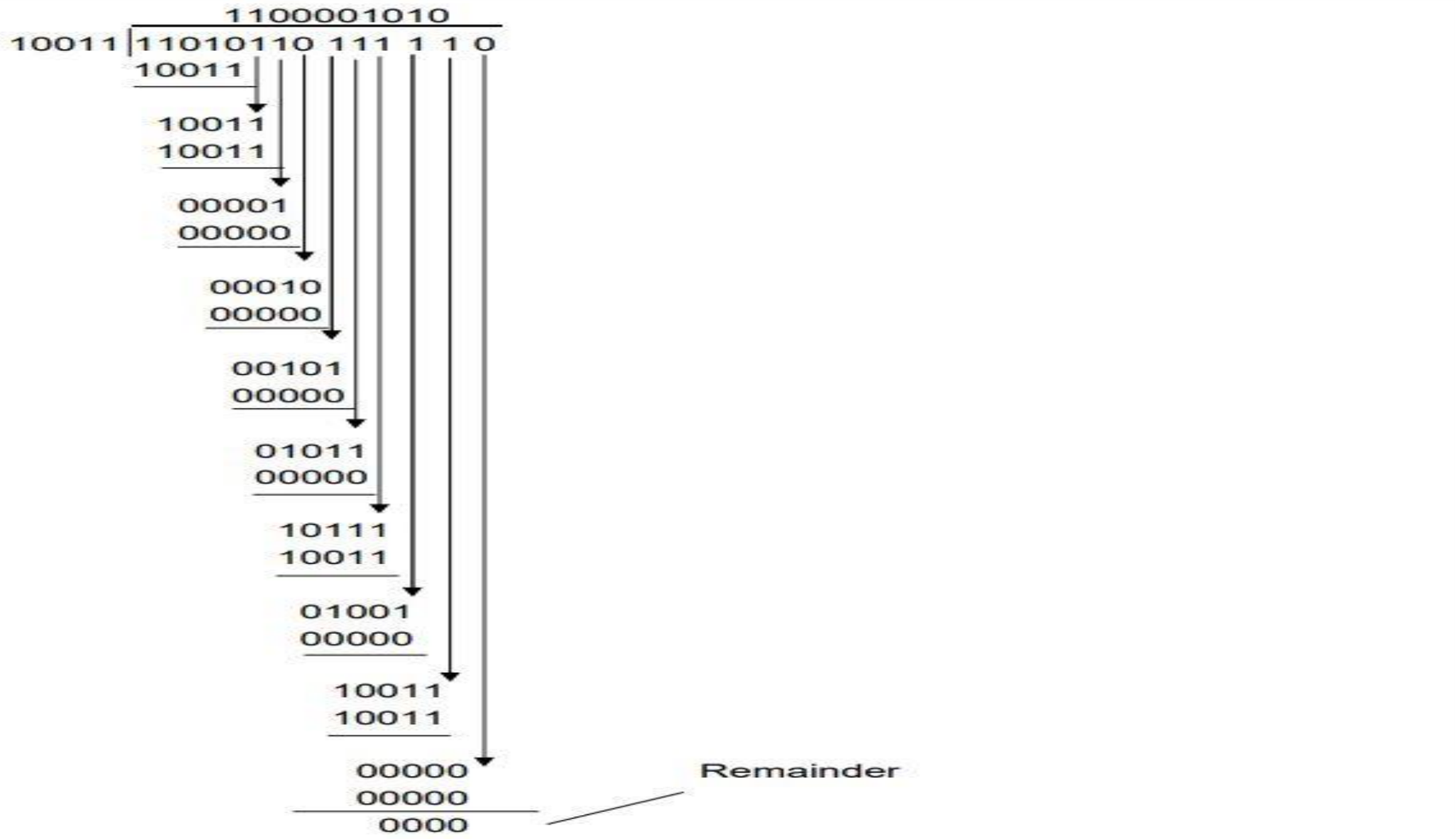
Ex.

- Frame: 1101011011
- Generator: 10011
- Message after appending 4 zero bits: 11010110000

Cont.


1100001010
10011 | 11010110 110 0 0 0
10011
10011
10011
00001
00000
00010
00000
00101
00000
01011
00000
10110
10011
01010
00000
10100
10011
01110
00000
1110
Remainder
Transmitted frame: 11010110111110

Cont.



Since the remainder is zero there is no error in the transmitted frame.

Error Correction

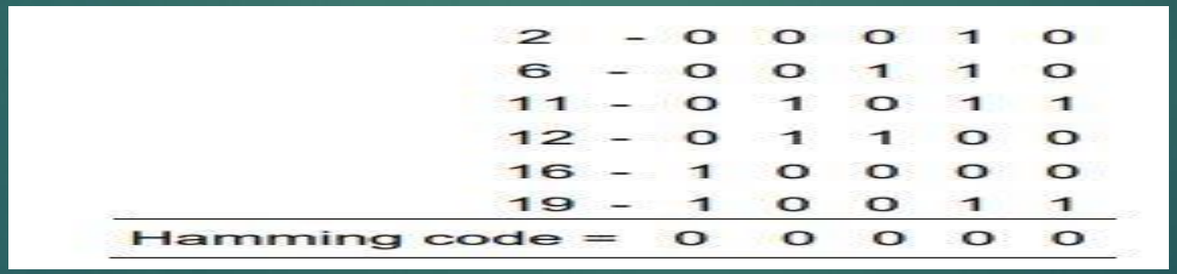
- Network designers have developed two basic strategies for dealing with errors. One way is to include enough redundant information along with each block of data sent, to enable the receiver to deduce what the transmitted data must have been. The other way is to include only enough redundancy to allow the receiver to deduce that an error occurred, but not which error, and have it request a retransmission. The former strategy uses error-correcting codes, and the latter uses error-detecting codes. The use of error-correcting codes is often referred to as forward error correction. Each of these techniques occupies a different ecological niche. On channels that are highly reliable, such as fiber, it is cheaper to use an error detecting code and just retransmit the occasional block found to be faulty. However, on channels such as wireless links that make many errors, it is better to add enough redundancy to each block for the receiver to be able to figure out what the original block was, rather than relying on a retransmission, which itself may be in error. To understand how errors can be handled, it is necessary to look closely at what an error really is. Normally, a frame consists of m data (i.e., message) bits and r redundant, or check, bits. Let the total length be n (i.e., n = m + r). An n-bit unit containing data and check bits is often referred to as an n-bit codeword.

Hamming codes

- Hamming codes provide another method for error correction. Error bits, called Hamming bits, are inserted into message bits at random locations. It is believed that the randomness of their locations reduces the odds that these Hamming bits themselves would be in error. This is based on a mathematical assumption that because there are so many more message bits compared with Hamming bits, there is a greater chance for a message bit to be in error than for a Hamming bit to be wrong. Determining the placement and binary value of the Hamming bits can be implemented using hardware, but it is often more practical to implement them using software. The number of bits in a message (M) are counted and used to solve the following equation to determine the number of Hamming bits (H) to be used: $2^H \geq M + H + 1$
- Once the number of Hamming bits is determined, the actual placement of the bits into the message is performed. It is important to note that despite the random nature of the Hamming bit placements, the exact sample placements must be known and used by both the transmitter and receiver. Once the Hamming bits are inserted into their positions, the numerical values of the bit positions of the logic 1 bits in the original message are listed. The equivalent binary numbers of these values are added in the same manner as used in previous error methods by discarding all carry results. The sum produced is used as the states of the Hamming bits in the message. The numerical difference between the Hamming values transmitted and that produced at the receiver indicates the bit position that contains a bad bit, which is then inverted to correct it.

Example

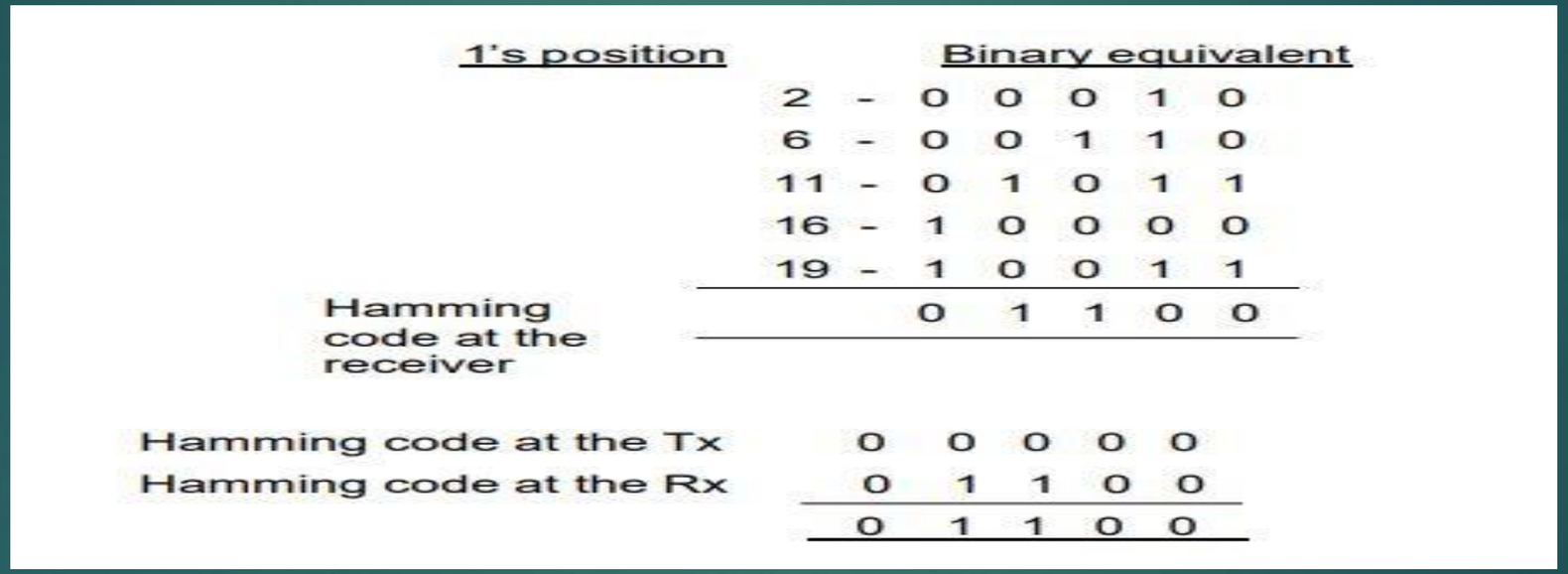
- The given data 100100011100101(14- bits)
- The number of hamming codes $2H \geq M + H + 1$
- H = ? M = 14 to satisfy this equation H should be 5 i.e., 5 hamming code bits should be incorporated in the data bits.
- 1 0 0 1 0 0 0 1 1 0 H 0 H 1 H 0 H 1 H
- Now count the positions where binary 1‘s are present. Add using mod 2 operation (Ex-OR). The result will give the Hamming code at the transmitter end.



- This Hamming code will be incorporated at the places of ‘H’ in the data bits and the data will be transmitted.

How to find out there is an error in the data?

- Let the receiver received the 12th bit as zero. The receiver also finds out the Hamming code in the same way as transmitter.



1's position		Binary equivalent
2	-	0 0 0 1 0
6	-	0 0 1 1 0
11	-	0 1 0 1 1
16	-	1 0 0 0 0
19	-	1 0 0 1 1
Hamming code at the receiver		0 1 1 0 0

Hamming code at the Tx	0 0 0 0 0
Hamming code at the Rx	0 1 1 0 0
	0 1 1 0 0

- The decimal equivalent for the binary is 12 so error is occurred at 12th place.

Data-Link Layer Protocols

AUTOPIAN SIMPLEX PROTOCOL

- The following assumption has been made for developing the (algorithm) simplex protocol. The channel is a perfect noiseless channel. Hence an ideal channel in which no frames are lost, duplicated, or corrupted. No flow control and error control used. It is a unidirectional protocol in which data frames are traveling in only one direction- from the sender to receiver. Both transmitting and receiving network layer are always ready. Processing time that is small enough to be negligible. Infinite buffer space is available.

A SIMPLEX STOP-AND-WAIT PROTOCOL FOR AN ERROR-FREE CHANNEL

- The following assumption has been made for developing the Stop-and-Wait Protocol. The channel is a perfect noiseless channel. Flow control used. It is a bidirectional protocol in which frames are traveling in both direction. Both transmitting and receiving network layer are always not ready. Processing time considerable. Finite buffer space is available. The receiver may not be always ready to receive the next frame (finite buffer storage). Receiver sends a positive acknowledgment frame to sender to transmit the next data frame which showed in the below. Error-free communication channel assumed. No retransmissions used

Design of Stop-and-Wait Protocol

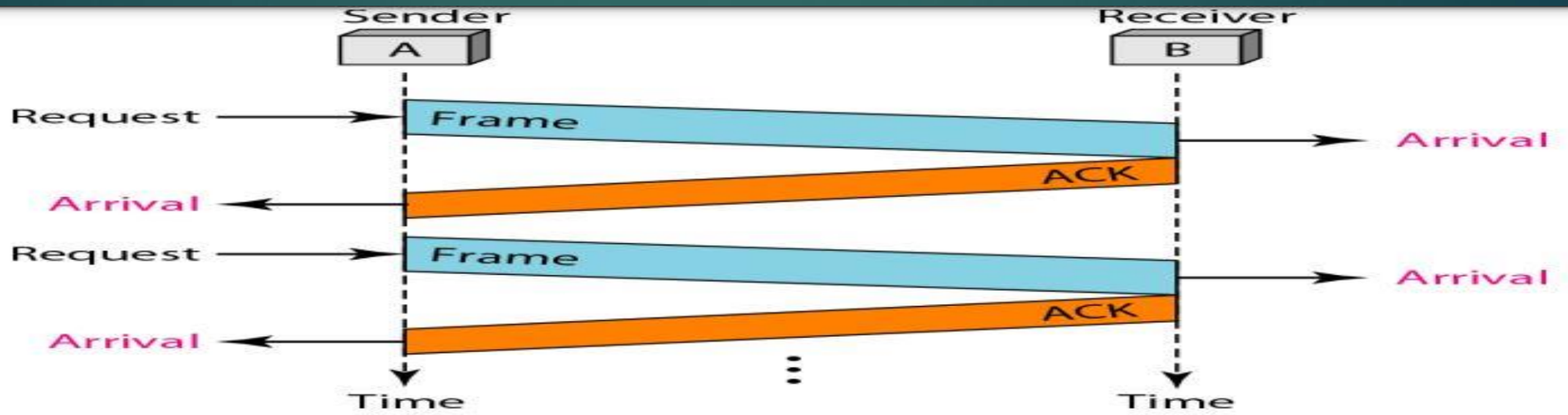


Figure 3.4. Stop-and-Wait protocol flow diagram

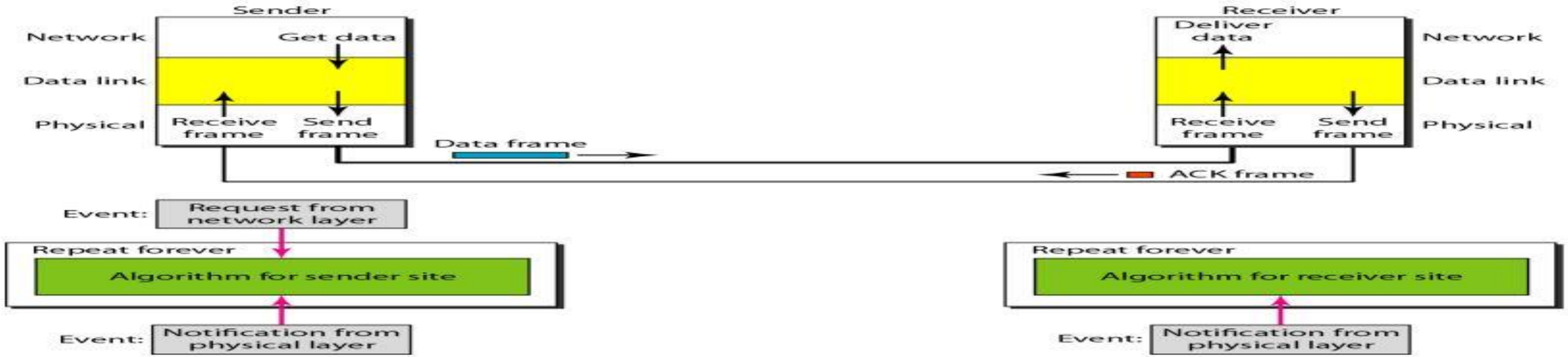


Figure 3.5 Design of Stop-and-Wait Protocol

HDLC

- High-Level Data Link Control (HDLC) is a communication protocol used for transmitting data between devices in telecommunication and networking.
- HDLC stands for High-Level Data Link Control, which is a group of protocols that allow data to be transmitted between network points. It's a bit-oriented, synchronous data link layer protocol created by the International Organization for Standardization (ISO). HDLC is part of the data link layer protocol in the OSI Model.
- HDLC is used for point-to-point and multipoint link structures. It provides both connection-oriented and connectionless service. HDLC frames include a beginning and ending flag, an address field, a control field, a data field, and a frame check sequence (FCS) for error checking. HDLC also uses the Automatic Repeat Request (ARQ) for full-duplex communication between network channels.

Types of HDLC Frames

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

- I-frame − I-frames or Information frames carry user data from the network layer. They also include flow and error control information that is piggybacked on user data. The first bit of control field of I-frame is 0.
- S-frame − S-frames or Supervisory frames do not contain information field. They are used for flow and error control when piggybacking is not required. The first two bits of control field of S-frame is 10.
- U-frame − U-frames or Un-numbered frames are used for myriad miscellaneous functions, like link management. It may contain an information field, if required. The first two bits of control field of U-frame is 11.

PPP

- Point-to-Point Protocol (PPP) is a TCP/IP protocol that is used to connect one computer system to another. Computers use PPP to communicate over the telephone network or the Internet. A PPP connection exists when two systems physically connect through a telephone line.
- Point-to-Point Protocol (PPP) is a suite of computer communication protocols that enable two computers to communicate over the internet or telephone network. PPP is a peer-to-peer protocol, which means that either side of the link can establish a connection. It's a common OSI data-link protocol that can encapsulate and transmit data to higher layer protocols, such as IP, over a physical serial transmission medium.
- PPP is typically used for internet connections and connecting remote networks via a Wide Area Network (WAN) link. It can provide loop detection, authentication, transmission encryption, and data compression.
- A PPP server typically sits idle until a PPP client attempts to establish a connection. If configured for authentication, the server responds to the client with an authentication request. The server then establishes the Network Control Protocol (NCP) used between systems.

Components of PPP

Point - to - Point Protocol is a layered protocol having three components −

- Encapsulation Component − It encapsulates the datagram so that it can be transmitted over the specified physical layer.
- Link Control Protocol (LCP) − It is responsible for establishing, configuring, testing, maintaining and terminating links for transmission. It also imparts negotiation for set up of options and use of features by the two endpoints of the links.
- Authentication Protocols (AP) − These protocols authenticate endpoints for use of services. The two authentication protocols of PPP are Password Authentication Protocol (PAP), Challenge Handshake Authentication Protocol (CHAP)
- Network Control Protocols (NCPs) − These protocols are used for negotiating the parameters and facilities for the network layer. For every higher-layer protocol supported by PPP, one NCP is there. Some of the NCPs of PPP are − Internet Protocol Control Protocol (IPCP), OSI Network Layer Control Protocol (OSINLCP), Internetwork Packet Exchange Control Protocol (IPXCP), DEC net Phase IV Control Protocol (DNCP), NetBIOS Frames Control Protocol (NBFCP), IPv6 Control Protocol (IPV6CP)

Thank You