ALLAMA IQBAL OPEN UNIVERSITY

USMAN NOTES
www.usmannotes.com

Name:

Registration No.:

Course Code: 3584

Program: Postgraduate

Semester: Autumn 2026

ALLAMA IQBAL OPEN UNIVERSITY, ISLAMABAD

(Department of Computer Science)

Course: Data Communication & Networks (3584) • Level: Postgraduate

Semester: Autumn 2026

Assignment No. 2

Total Marks: 100 • Pass Marks: 50

SOLVED ASSIGNMENT GUIDE — For Study & Reference

Note: This guide is prepared for study and reference purposes only. Students are advised to read the relevant parts of the course book, understand the concepts, redraw the diagrams and work through the examples themselves, and then write the assignment in their own words. Copying this guide word for word is not recommended. No units are printed in the question booklet for this course, so none are shown on this title page.

USMAN NOTES www.usmannotes.com

Question 1: Network Topologies — Types in Detail, with Examples (20 Marks)

Question as printed: Define network topology? Explain different types of topologies in detail with examples. (20)

Introduction

Before a network can be cabled, someone must decide its shape: which device connects to which, and along what paths messages will travel. That shape — more precisely, the arrangement of the nodes and links of a network — is its topology, and the choice has long

consequences for cost, reliability, performance and ease of growth. A topology that suits a small office will fail in a bank's data centre, and the tolerant, redundant shape a hospital needs is wasted money in a classroom. This answer defines topology, distinguishes its physical and logical forms, and then explains the bus, star, ring, mesh, tree and hybrid topologies in detail, each with its working, advantages, disadvantages and real examples.

What Is Network Topology?

Network topology is the geometric arrangement of the elements (nodes and links) of a communication network — the pattern in which stations are interconnected. Two aspects must be distinguished. Physical topology describes how devices are actually connected by cables or radio links: the literal layout an installer could draw on a floor plan. Logical topology describes how data actually travel through the network, which may differ from the physical picture: a network wired as a physical star through a hub can behave logically as a bus, because the hub repeats every signal to all stations, while a physical star built on a switch behaves logically as many separate point-to-point links. Topology is a design decision made together with the choice of medium, access-control method and devices, and its evaluation criteria recur throughout this answer: installation cost and cabling, what happens when one node or one link fails, how easily the network can grow, how much traffic it can carry, and how quickly a fault can be found and repaired.

USMAN NOTES www.usmannotes.com

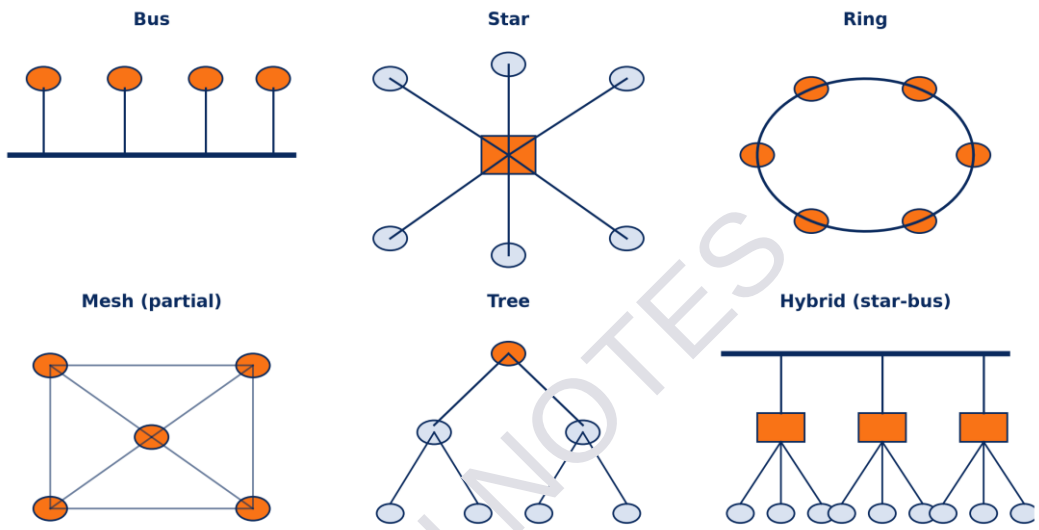


Figure 1: The standard topologies: bus, star, ring, mesh, tree and a hybrid star-bus arrangement.

Bus Topology

In a bus topology all stations attach, through short drop cables and taps, to a single long common cable — the bus or backbone — which is terminated at both ends so that signals are absorbed rather than reflected. A transmission from any station travels in both directions along the bus and is received by every station, though only the addressed station keeps it; an access-control method such as CSMA/CD (Question 4) is needed so that stations do not transmit simultaneously and corrupt each other. The bus is cheap and simple: it uses the least cable of any topology, is easy to extend by tapping the backbone, and was the shape of the original Ethernet (10BASE5 and 10BASE2) in offices and laboratories. Its weaknesses are structural: the backbone is a single point of failure whose break or faulty terminator can silence the whole network, faults are notoriously hard to localise, performance falls as stations are added because all share one medium, and the topology does not scale to large or busy networks. It

survives today mainly in legacy installations, in industrial control buses and conceptually inside the shared segments of other designs.

Star Topology

In a star topology every station has its own dedicated cable to a central device — historically a hub, in modern networks a switch. All communication passes through the centre: a frame from one station is received by the switch and forwarded only out of the port leading to the destination. The star is the dominant LAN topology of the present day, as in a typical office or university computer lab where every PC cables back to a wiring-closet switch. Its advantages explain that dominance: a failed station cable isolates only that station, so the network is comparatively robust; faults are easy to find (the dead branch points at itself); adding or removing a station is trivial and does not disturb others; and with a switch, each station enjoys its own collision-free bandwidth. Its costs are the extra cabling — every station needs a home run — and dependence on the central device, whose failure stops the whole star and which must therefore be reliable and, in serious installations, duplicated.

Ring Topology

In a ring topology stations are connected in a closed loop, each station linked point-to-point to its two neighbours, and data circulate around the ring, usually in one direction, passing through each intermediate station until they reach their destination and are removed. Access is commonly controlled by a token: only the station holding the token may transmit, which guarantees fair, collision-free, predictable access even under heavy load. IBM's Token Ring LANs and the dual counter-rotating rings of FDDI (Fibre Distributed Data Interface) are the classic

examples; FDDI's second ring wraps to heal the loop when a link breaks, giving the ring family a genuine resilience option. Rings offer orderly performance and easy addition of stations around the loop, but a single failed station or link in a simple ring can break circulation, troubleshooting a dead loop is awkward, and reconfiguration disturbs the whole ring. Pure ring LANs are now rare, though ring structures persist in metropolitan fibre networks and in industrial and provider backbones, where the dual-ring self-healing idea remains valuable.

Mesh Topology

In a mesh topology stations are interconnected by multiple redundant paths. In a full mesh every node connects directly to every other node — needing n(n−1)/2 links for n nodes — while in a partial mesh, the practical form, nodes have several connections but not all possible ones. The virtue of the mesh is survivability and capacity: if any link or node fails, traffic simply reroutes along another path, and the many parallel paths give high aggregate throughput. This is why the core of the Internet and of telephone and provider backbones is a partial mesh of routers, and why military and banking networks, which must keep working under damage, are meshed. The cost is equally clear: cabling and interface counts explode combinatorially in a full mesh, installation and management are complex, and mesh designs require routing protocols to choose among the paths. Wireless mesh networks bring the same idea to radio, with each node relaying for its neighbours.

Tree and Hybrid Topologies

A tree (hierarchical) topology arranges stations as an inverted tree: a root device connects to intermediate devices, each of which serves a group of stations or a further level of devices — in

effect, stars joined by a backbone. A campus network is the everyday example: a core switch at the root, distribution switches for each building, and access switches for each floor or laboratory. Trees scale gracefully, localise traffic and faults within branches, and match the structure of organisations, but the root and upper links are critical points whose failure cuts off whole subtrees, and total cabling is substantial. A hybrid topology is any deliberate combination of two or more basic shapes, chosen so that each part of the network gets the shape that suits it: a star-bus hybrid links several departmental stars on a common backbone, a star-ring serves a large site, and most real enterprise networks are hybrids of a meshed core, a tree distribution layer and star access. Hybrids are not a failure of classification but its purpose: the basic topologies are building blocks and engineering consists of combining them under the constraints of budget, building layout and reliability targets.

USMAN NOTES www.usmannotes.com

Table 1: Topologies compared

Topology	Connection pattern	Single point of failure	Cost / cabling	Real example
Bus	All stations on one shared backbone	The backbone itself	Lowest	Legacy 10BASE2 Ethernet, industrial buses
Star	Every station to a central hub/switch	The central device	Moderate — one cable per station	Modern Ethernet LAN in an office or lab
Ring	Closed loop, station to station	A link/station (unless dual ring)	Moderate	Token Ring, FDDI, metro fibre rings

Mesh	Many redundant paths (full or partial)	None in a good partial mesh	Highest	Internet / provider backbone, wireless mesh
Tree	Root, branches, leaves (hierarchy)	Root and upper-level links	Moderate to high	Campus network: core, distribution, access
Hybrid	Combination of the above	Depends on the design	Depends on the design	Enterprise star-bus and star-mesh networks

USMAN NOTES www.usmannotes.com

Conclusion

To conclude, topology is the shape of a network, physical in its cabling and logical in the paths data actually take. The bus is cheapest and most fragile; the star trades extra cable for robustness and has become the default LAN shape; the ring offers orderly token-controlled access and, in dual form, self-healing; the mesh buys survivability with links and complexity and so guards network cores; the tree organises large sites hierarchically; and hybrids combine these blocks to fit real buildings and budgets. The designer's question is never which topology is best in the abstract, but which failure modes, costs and growth pattern the particular network can live with.

Question 2: Asynchronous versus Synchronous Transmission; Types of Multiplexing (20 Marks)

Question as printed: Differentiate asynchronous and synchronous transmission in detail? Also, explain different types of multiplexing. (20)

Introduction

Two timing problems dominate the design of a data link. The first is synchronisation: how does the receiver know where one unit of data ends and the next begins? Asynchronous and synchronous transmission are the two classical answers, differing in whether timing is supplied character by character or shared continuously across long blocks. The second problem is sharing: an expensive high-capacity link would be wasted on one user, so multiplexing lets many signals share it. This answer contrasts the two transmission modes in detail and then explains frequency-division, time-division (synchronous and statistical), wavelength-division and code-division multiplexing.

Asynchronous Transmission

In asynchronous transmission, data are sent one character (typically 5–8 bits) at a time, whenever the sender has a character ready, with no fixed time relationship between successive characters: gaps of random length may separate them, exactly as a typist's keystrokes are irregular. The receiver cannot assume timing, so each character carries its own synchronisation: a start bit (always in the opposite state to the idle line) warns the receiver that a character is beginning and lets it synchronise its clock, the data bits follow, an optional parity bit supports

error checking, and one or more stop bits return the line to the idle state and guarantee a minimum gap before the next start bit. The scheme is simple and cheap — no shared clock, tolerant of irregular sources such as keyboards and sensor readings — and a timing error affects at most one character. Its price is overhead and inefficiency: two or three extra bits surround every small character, wasting perhaps 20–30 per cent of the line's capacity, and the achievable bit rate is modest. It remains the mode of serial (COM/UART) ports, microcontroller links, modems' local interfaces and many low-speed device connections.

Synchronous Transmission

In synchronous transmission, data are sent as a continuous stream grouped into long blocks or frames, with no start and stop bits around individual characters. Sender and receiver clocks must stay synchronised for the whole block, achieved either by a separate clock line, by embedding timing in the signal encoding so the receiver recovers the clock from the transitions, or by short synchronising preambles (such as SYN characters or flag patterns) that open each block. Because synchronisation information is paid for once per block rather than once per character, the overhead percentage becomes very small for large blocks, and much higher bit rates and efficiency are possible — which is why synchronous working is used for mainframe and network links, HDLC/SDLC and the framed transmission inside modern data networks. The costs are complexity and risk: the framing must identify block boundaries exactly, the receiver needs buffering for whole blocks, clocks must be more precise, and a synchronisation slip can damage a whole block rather than one character, so synchronous protocols wrap their blocks in the error-control machinery of Question 3.

Table 1: Asynchronous versus synchronous transmission

Feature	Asynchronous	Synchronous
Unit sent	One character (5–8 bits) at a time	Long blocks / frames of many characters
Gaps between units	Irregular, of any length	None inside a block; continuous stream
Synchronisation	Start and stop bits around every character	Shared/embedded clock; preamble or flags per block
Overhead	High — roughly 2–3 bits per character	Low — small fixed overhead per large block
Data rate / efficiency	Low to moderate; suits bursty sources	High; suits continuous heavy traffic
Complexity and cost	Simple, cheap interfaces	More complex; buffering and precise clocks
Effect of a timing error	Usually one character lost	A whole block may be lost
Typical uses	Keyboards, serial ports, UARTs, sensors	Network links, HDLC, high-speed framed systems

USMAN NOTES www.usmannotes.com

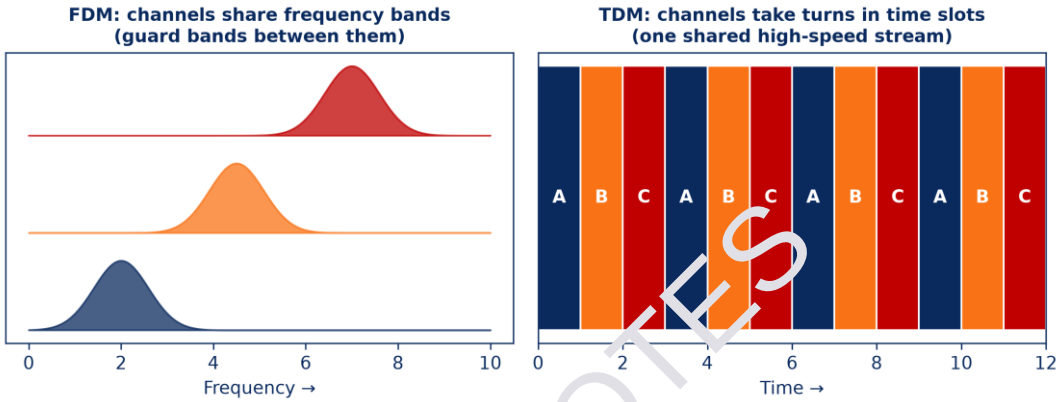


Figure 1: Multiplexing: FDM shares a link by frequency; TDM shares it by time. In both, one high-capacity stream carries many channels.

Multiplexing: The Idea

Multiplexing is the set of techniques by which a single transmission medium or link carries several signals simultaneously (or in rapid rotation), so that the expensive resource — a trunk cable, a fibre, a satellite channel — is shared. A multiplexer (MUX) at the sending end combines the incoming channels into one composite stream, and a demultiplexer (DEMUX) at the receiving end separates them again. The economic logic is compelling: links exhibit strong economies of scale, so one high-capacity circuit costs far less than many low-capacity ones of equal total rate. Multiplexing must be distinguished from simple line sharing or concentration: true multiplexing gives each channel a guaranteed portion of the shared resource, whether that portion is a band of frequencies, a recurring time slot, a wavelength or a code.

Frequency-Division Multiplexing (FDM)

In frequency-division multiplexing the total bandwidth of the link is divided into a set of non-overlapping frequency bands, one per channel, separated by unused guard bands that prevent

interference. Each source's signal is modulated onto a carrier within its assigned band, the modulated signals are combined for transmission, and at the receiver a bank of filters separates the bands and demodulates each channel. The channels operate simultaneously and continuously — each owns its slice of spectrum for the whole call or session, whether or not it has anything to send. FDM is an analogue technique at heart, though it can carry digital data modulated onto carriers. Examples surround us: broadcast radio and television (each station on its own frequency), the traditional analogue telephone trunk system, cable television, and ADSL, which places voice, uplink data and downlink data in separate bands on one telephone line. Its weakness is waste: an idle channel's band cannot be lent to a busy one, and guard bands consume spectrum.

Time-Division Multiplexing (TDM): Synchronous and Statistical

In time-division multiplexing the link's capacity is divided not in frequency but in time: the composite stream is organised into repeating frames, each frame contains time slots, and each input channel is assigned slots through which its data travel in rapid rotation — fast enough that each channel behaves as if it had a slower private link. In synchronous TDM the slots are fixed in advance: channel 3 receives the third slot of every frame whether it has data or not, so empty slots waste capacity but the receiver needs no labels — slot position identifies the channel. The digital telephone hierarchy (the T1/E1 systems) is built exactly this way to carry many digitised voice calls on one trunk. Statistical (asynchronous) TDM removes the waste: slots are allocated on demand only to channels that actually have data waiting, so the composite rate can be less than the sum of the input rates. The price is that slots must now carry address information identifying their owner, the multiplexer needs buffering and

intelligence, and the guarantee changes from fixed capacity to capacity on average — a trade that suits bursty computer traffic, which is why packet switching, the logical extreme of statistical sharing, dominates data networks. Wavelength-division multiplexing (WDM), the optical counterpart of FDM, sends each channel as light of a different wavelength (colour) down the same fibre; in its dense form (DWDM) dozens of wavelengths, each carrying many gigabits per second, multiply a single fibre's capacity enormously and underlie modern backbone capacity. Code-division multiplexing (CDM/CDMA) takes the opposite approach to separation: all channels transmit over the same band at the same time, but each is encoded with a unique spreading code; the receiver, knowing the code, extracts its channel while others appear as low-level noise. CDMA powered third-generation mobile networks and illustrates that multiplexing can divide a resource by code as well as by frequency, time or wavelength.

Conclusion

To conclude, asynchronous transmission synchronises afresh with start and stop bits around every character — simple, wasteful and perfect for irregular low-speed sources — while synchronous transmission shares a clock across long framed blocks, achieving high efficiency at the cost of complexity. Multiplexing then shares the resulting link: FDM by slices of spectrum, TDM by turns in time (fixed in synchronous TDM, on demand in statistical TDM), WDM by colours of light in a fibre, and CDM by spreading codes. Both halves of the answer express the same engineering instinct: spend a little structure — bits, slots or bands — to buy a great deal of efficiency.

Question 3: Flow Control Techniques; Error Detection and Error Control (20 Marks)

Question as printed: Explain different flow control techniques. Also, define error detection and error control techniques. (20)

Introduction

A data link has two natural enemies. One is imbalance: senders are often faster than receivers, and data poured into a device faster than it can process them will overflow its buffers and be lost. The other is corruption: noise, interference and distortion damage bits in transit, as Assignment 1 showed, and damaged data delivered silently can be worse than no data at all. Flow control answers the first enemy by pacing the sender; error detection and error control answer the second by recognising damaged frames and arranging their repair. This answer explains the principal flow-control techniques — stop-and-wait and sliding-window, with its common variants — and then defines and explains error detection (parity, checksum and CRC) and error control (the ARQ family and forward error correction).

Flow Control: Purpose and the Feedback Principle

Flow control is the set of techniques that regulate the amount of data a source may send before it must wait, so that the rate of transmission matches the receiver's capacity to accept and process it. Almost all practical flow control is feedback-based: the receiver informs the sender, explicitly or implicitly, how much more it can accept, and the sender obeys. A cruder open-loop alternative — simply designing the system so overload 'cannot happen' — rarely

survives contact with real traffic. Flow control must also not be confused with congestion control: flow control protects one receiver from its sender, while congestion control protects the shared network itself from total overload. The two techniques below operate at the data-link layer between two directly connected devices; the same sliding-window idea reappears end-to-end in TCP at the transport layer.

Stop-and-Wait Flow Control

Stop-and-wait is the simplest possible scheme. The source transmits one frame and then stops and waits. The destination, having received the frame and prepared to accept another, returns a short acknowledgement (ACK); only on receiving that acknowledgement does the source send the next frame. If the receiver is not ready, it simply withholds the acknowledgement (or sends a negative response), and the sender waits. The mechanism is easy to implement and guarantees the receiver can never be overrun, since at most one frame is ever unacknowledged. Its weakness is efficiency: the line sits idle during the whole round trip — propagation of the frame, processing, propagation of the ACK — so on links with long propagation delay or high bit rates the utilisation of the channel can fall to a few per cent, a penalty sometimes called the stop-and-wait inefficiency. It remains appropriate for short, slow or simple links and as the conceptual foundation on which better schemes are built. A timing example makes the point concrete: if a frame takes 1 ms to transmit but the round-trip delay is 20 ms, the sender spends most of every cycle waiting, and the expensive link is mostly silent.

Sliding-Window Flow Control

Sliding-window flow control removes the idle time by letting the sender transmit several frames before pausing. Each frame carries a sequence number. The receiver maintains a receive window — the set of sequence numbers it is prepared to accept — and the sender maintains a matching send window of frames it may transmit without yet having received acknowledgements. As acknowledgements arrive (often cumulatively, one ACK confirming all frames up to a given number), the sender's window slides forward over the sequence space, opening room for new frames; a piggybacking optimisation lets a station attach its acknowledgements to data frames travelling in the opposite direction, avoiding separate ACK traffic. The window size is set from the receiver's buffer capacity and the link's delay-bandwidth characteristics: a window at least large enough to keep the pipe full during one round trip restores efficiency to near 100 per cent on suitable links. Errors complicate the picture in two standard ways, which are properly error-control variants: in go-back-N, when a frame is damaged the receiver discards it and all following frames, and the sender retransmits from the damaged frame onward — simple for the receiver, wasteful on error-prone links; in selective repeat (selective reject), only the actually damaged frames are retransmitted and good out-of-order frames are buffered — efficient, but requiring more memory and logic at both ends. A related rate-based alternative, used in some technologies, dispenses with windows and instead grants the sender a credit or a permitted rate, but the window remains the classic and most examined technique.

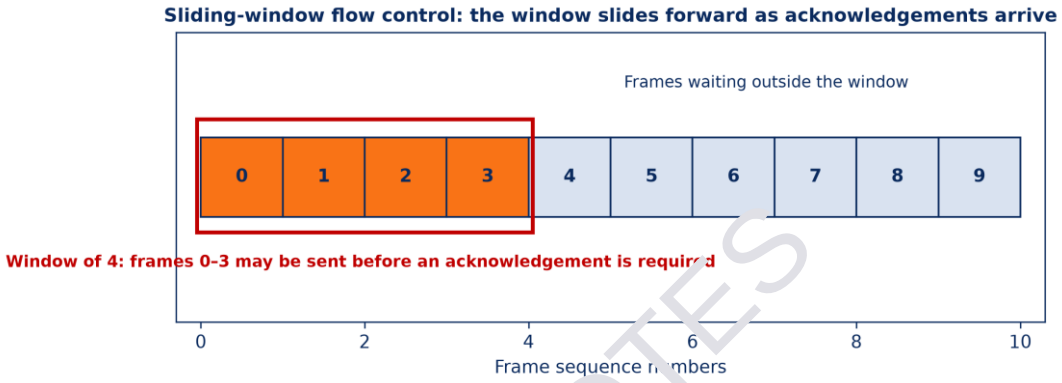


Figure 1: Sliding window: with a window of four, frames 0–3 may be outstanding; each acknowledgement slides the window forward.

Table 1: Stop-and-wait versus sliding window

Feature	Stop-and-wait	Sliding window
Frames outstanding	Exactly one	Up to the window size W
Acknowledgement effect	Releases the single next frame	Slides the window; may acknowledge cumulatively
Link utilisation	Very poor on long/fast links	Can approach full utilisation
Buffer needs	Minimal	Up to W frames at each end
Complexity	Trivial	Sequence numbers, timers, window management
Suitable for	Short, slow, simple links	Long-distance and high-speed links

Error Detection: Parity, Checksum and CRC

Error detection is the use of redundancy — extra bits calculated from the data and sent with it — so that the receiver can recognise when a frame has been damaged in transit. Detection

does not itself repair anything; it tells the receiver to discard the frame and triggers the error-control machinery described next. The simplest technique is the parity bit: one extra bit appended to each character so that the total number of 1 bits is even (even parity) or odd (odd parity); it catches any odd number of bit errors in the character but is blind to an even number of errors, and is used in asynchronous character transmission (Question 2). A two-dimensional extension arranges data in a table with row and column parity, which can even locate and correct a single-bit error. The checksum, used in IP and TCP, adds the data words (in ones' complement arithmetic) and sends the complemented sum; the receiver re-adds everything and checks for the expected result. The strongest and most widely used technique is the cyclic redundancy check (CRC): the sender treats the frame as a long binary number, divides it by an agreed generator polynomial, and appends the remainder (typically 16 or 32 bits) as the frame check sequence; the receiver repeats the division and declares an error if its remainder differs. CRC detects all single and double errors, all burst errors up to the length of the check, and the overwhelming majority of longer bursts — which is why Ethernet, Wi-Fi and HDLC all rely on it. The governing principle of all three is the same trade: a small, fixed percentage of extra bits buys an enormous reduction in the probability of undetected error.

Error Control: ARQ and Forward Error Correction

Error control is the complete mechanism — detection plus response — that ensures damaged or lost frames are nevertheless delivered correctly. Where a return channel exists, the standard response is automatic repeat request (ARQ): the receiver acknowledges good frames and the sender retransmits any frame that is not acknowledged within a timeout, or that is explicitly negatively acknowledged. ARQ comes in exactly the three forms introduced with the sliding

window: stop-and-wait ARQ (one frame at a time, with duplicate detection by alternating sequence numbers), go-back-N ARQ, and selective-repeat ARQ; the choice balances receiver simplicity against retransmission waste, given the link's error rate and delay. Where no return channel exists, or delay makes retransmission impractical — broadcasting, deep-space links, real-time media — the alternative is forward error correction (FEC): the sender adds enough structured redundancy (as in Hamming, Reed–Solomon or convolutional codes) for the receiver to reconstruct the original data despite a limited number of errors, without asking for anything to be resent. Real systems mix the strategies: a wireless LAN uses CRC detection with ARQ retransmission, a satellite television broadcast uses heavy FEC because viewers cannot talk back, and a stored file transfer uses both error-detecting codes and end-to-end checks. It should be noted honestly that no scheme makes error impossible — only arbitrarily improbable — and that the application ultimately decides how much residual risk is acceptable.

Conclusion

To conclude, flow control protects the receiver from the sender: stop-and-wait paces transmission one frame per acknowledgement, perfectly safe and perfectly wasteful, while the sliding window keeps a controlled number of frames in flight and slides forward on acknowledgements, restoring full link efficiency. Error handling protects the data from the channel: detection codes — parity for characters, checksums for packets, CRC for frames — recognise damage, and error control then repairs it, by ARQ retransmission where a return path exists (stop-and-wait, go-back-N or selective repeat) or by forward error correction where it does not. Together these mechanisms convert a noisy, imperfect physical link into the dependable service that every higher layer assumes.

Question 4: LAN Architecture; CSMA/CD versus Gigabit LANs (20 Marks)

Question as printed: Define LAN architecture. Differentiate CSMA/CD and Gigabit LANs. (20)

Introduction

The local area network is where most users meet data communication: the network of the office, campus or home. To discuss such a network sensibly one needs its architecture — the layered set of choices about media, access control and protocols that defines it — and its history is largely the history of Ethernet and its access method, CSMA/CD. Gigabit LANs then mark the point at which Ethernet grew a thousand times faster than its original form and, in growing, had to change some of the very rules that defined it. This answer defines LAN architecture, outlines its layers and reference model, and then differentiates CSMA/CD operation from Gigabit (Ethernet) LANs in detail.

LAN Architecture Defined

A local area network (LAN) is a privately owned communication network that interconnects devices within a limited area — a room, building or campus — at high data rates, using media owned by the organisation rather than a public carrier. LAN architecture is the structured set of design choices that together specify how such a network works: its topology and transmission media, its medium access control method, and the layered protocol structure through which stations communicate. In the IEEE 802 reference model, the architecture refines the two lowest OSI layers. The physical layer specifies the encoding, signalling rate and medium (twisted pair,

fibre, radio). Above it, the data-link function is split into two sublayers: the Medium Access Control (MAC) sublayer, which governs who may transmit on the shared medium and assembles frames with MAC addresses and error checks, and the Logical Link Control (LLC) sublayer (IEEE 802.2), which provides a common interface to the network layer, with connectionless or connection-oriented service, independently of which MAC technology lies beneath. Ethernet (802.3), Token Ring (802.5), and wireless LANs (802.11) are alternative MAC/physical combinations under the same LLC roof. The architectural point is modularity: upper layers and applications survive unchanged when the physical technology beneath them is replaced — which is exactly what happened, repeatedly, as Ethernet evolved from 10 Mbps to gigabit speeds and beyond. A complete LAN architecture also embraces the interconnecting devices of Question 5 — hubs, bridges and switches — and the structured cabling that ties the star-wired building together.

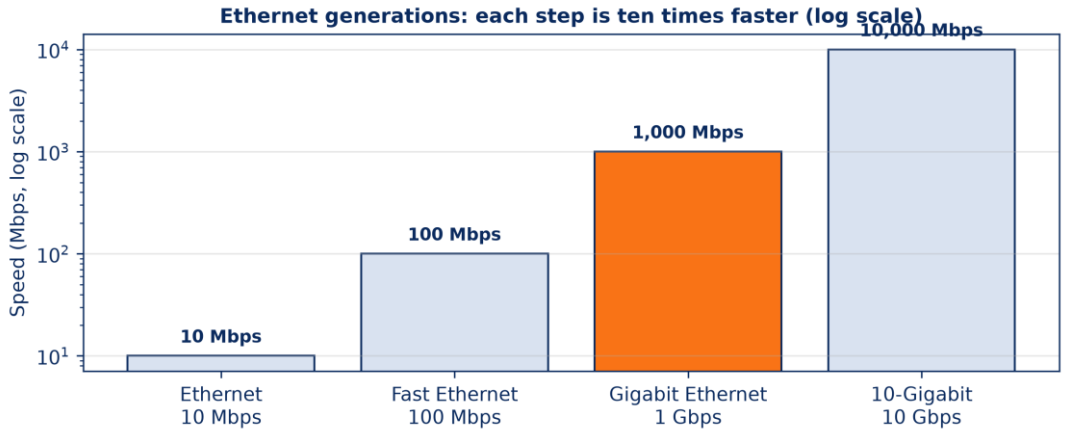


Figure 1: Ethernet generations: from 10 Mbps to 10 Gbps, each generation ten times faster than the last (logarithmic scale).

CSMA/CD: How Classic Ethernet Shares a Medium

CSMA/CD — Carrier Sense Multiple Access with Collision Detection — is the access-control protocol of classic shared-medium Ethernet. Multiple access means many stations share one channel; carrier sense means a station listens before transmitting and defers if the medium is busy; collision detection means that if two stations nevertheless begin together (because both sensed an idle medium within the propagation window), each detects the resulting collision, transmits a short jamming signal so all parties know, stops, and retries after a random backoff interval chosen by the binary exponential backoff algorithm, which lengthens the range of possible delays after repeated collisions. CSMA/CD defines the whole character of classic Ethernet: half-duplex operation, a minimum frame size (64 bytes) guaranteeing that a collision is still detectable when the frame ends, a maximum network diameter limiting how far signals may travel, and an efficiency that falls as load and network size grow. In its original form it ran at 10 Mbps over coaxial bus and later over twisted-pair stars built on hubs, which repeated signals to all ports and so preserved one shared collision domain.

Gigabit LANs: Ethernet at 1 Gbps

A Gigabit LAN — in practice, Gigabit Ethernet (IEEE 802.3z over fibre and 802.3ab over twisted pair) — is a LAN operating at 1,000 Mbps, one hundred times the original Ethernet rate, while retaining the Ethernet frame format, addressing and management model so that existing networks can upgrade incrementally. Two physical families matter: 1000BASE-X, over optical fibre (and short copper jumper), used for backbones and switch uplinks, and 1000BASE-T, over four pairs of Category 5e/6 unshielded twisted pair, used to the desktop and throughout

modern buildings. The difficulty gigabit speed creates is timing: at 1 Gbps a 64-byte frame occupies the medium for only about half a microsecond, too briefly for collision detection to work over normal cable distances. The standard therefore extended the minimum slot time by carrier extension (padding short frames' carrier on the wire) and added frame bursting, letting a station send a sequence of frames in one access. More importantly, the way gigabit LANs are actually deployed bypasses the problem: they are built with switches in full-duplex mode, each port a private point-to-point link on which collisions cannot occur and CSMA/CD is switched off entirely; traffic is then limited by the switch fabric, not by contention. CSMA/CD survives in gigabit standards only for the rarely used half-duplex shared mode. The result is that a modern 'Gigabit LAN' is not merely a faster version of the 1980s network; it is a switched, full-duplex, structured-cabling system in which the old bus and its contention protocol have quietly retired, while their frame format and addressing live on.

Table 1: CSMA/CD (classic shared Ethernet) versus Gigabit LANs (Gigabit Ethernet)

Feature	CSMA/CD — classic Ethernet	Gigabit LAN (Gigabit Ethernet)
Role	A medium access control protocol	A complete LAN technology / generation (which may use CSMA/CD or not)
Nominal speed	10 Mbps (classic); 100 Mbps in Fast Ethernet form	1,000 Mbps (1 Gbps)
Mode	Half-duplex: one transmission at a time on the shared medium	Predominantly full-duplex switched: simultaneous send and receive

Medium and topology	Coaxial bus, later UTP star through hubs	Fibre (1000BASE-X) and Cat 5e/6 UTP (1000BASE-T), star through switches
Collision domain	The whole shared segment is one collision domain	Each switch port is its own domain; none in full-duplex
Collision handling	Detect, jam, binary exponential backoff, retransmit	Unnecessary in switched full-duplex; carrier extension + frame bursting defined for half-duplex mode
Minimum frame / timing	64-byte minimum frame matched to network diameter	Same 64-byte frame, but slot time extended on the wire
Efficiency under load	Falls with contention as load rises	Determined by switch capacity; scales with ports
Typical setting	Legacy shared LANs of the 1980s–90s	Modern building backbones, server and desktop connections

One conceptual caution completes the comparison: CSMA/CD and 'Gigabit LAN' are not objects of the same kind. CSMA/CD is a rule for sharing; a Gigabit LAN is a whole operating network at a stated speed, defined by its physical standards, devices and architecture, which in its switched full-duplex form has no sharing left to arbitrate. Examination answers lose marks by comparing them as if they were rival products; the correct picture is historical and architectural — CSMA/CD carried Ethernet's first generations, and gigabit speeds forced Ethernet to outgrow it.

Conclusion

To conclude, LAN architecture is the layered set of choices — physical medium and signalling, MAC and LLC sublayers, topology and devices — that specifies a privately owned, high-speed local network, standardised in the IEEE 802 family. CSMA/CD is the contention protocol of the shared-medium era: listen, transmit, detect collisions, back off and retry, at the price of half-duplex working and load-sensitive efficiency. Gigabit LANs deliver a hundred times the classic speed by changing the physical layers, extending the timing rules where half-duplex survives, and above all by switching to full-duplex switched stars in which collisions, and therefore CSMA/CD itself, disappear — while Ethernet's frames, addresses and architecture continue unchanged on top.

Question 5: Short Notes — Switch, Router, Bridge and Proxy Server (20 Marks)

Question as printed: Write short notes on the following with respect to their functionality: (20) (a) Switch (b) Router (c) Bridge (d) Proxy Server

Introduction

Networks are joined together, and made useful, by intermediate devices that move traffic intelligently. The four devices in this question are often confused because several of them sit in similar boxes in a wiring closet — yet each works at a different layer, reads a different kind of address, and makes a fundamentally different forwarding decision. The switch and the bridge are close relatives inside the LAN; the router joins different networks together; and the proxy

server, unlike the other three, does not forward traffic transparently at all but terminates it and speaks on the client's behalf. Each note below describes function first, then working, advantages and limits.

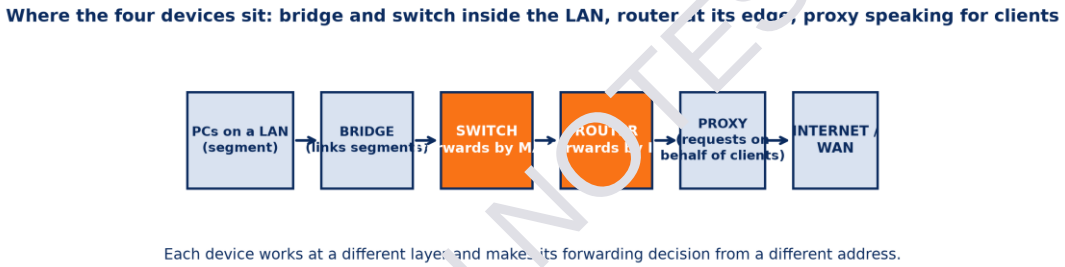


Figure 1: The path of traffic: bridge and switch operate inside the LAN, the router joins the LAN to other networks, and the proxy server acts for its clients.

(a) Switch

A switch is a multiport data-link layer (Layer 2) device that interconnects the stations of a LAN and forwards each frame only to the port leading to its destination. To do this the switch maintains a MAC address table (also called a CAM table) that it builds by learning: when a frame arrives on a port, the switch records the frame's source MAC address against that port. A frame whose destination is known is then forwarded out of that single port (filtering/forwarding); a frame for an unknown destination is flooded to all ports; broadcast frames are likewise flooded. Because each port is its own collision domain and, in full-duplex mode, a station and the switch can transmit simultaneously, a switched LAN abolishes the contention of CSMA/CD (Question 4) and multiplies the LAN's aggregate capacity by the number of ports. Modern switches work in hardware at wire speed, support VLANs that partition one physical switch into several logical LANs, may implement spanning-tree protection

against loops, and are sold from small desktop units to chassis switches with hundreds of ports. A Layer 3 ('multilayer') switch adds routing functions in hardware and blurs into the router category. The switch's limitation is its horizon: it knows MAC addresses within its broadcast domain and nothing of the wider network — for that, a router is required.

(b) Router

A router is a network layer (Layer 3) device whose function is to interconnect different networks and forward packets between them along the best available path. Each router interface belongs to a different network and has its own IP (logical) address. When a packet arrives, the router reads its destination IP address, consults its routing table — built manually from static entries or automatically by routing protocols such as RIP, OSPF and BGP, which exchange information about reachable networks and their costs — selects the outgoing interface and next hop, repackages the packet into the frame format of the outgoing link, and forwards it. Hop by hop, packet by packet, this process carries data across interconnected networks of completely different technologies — from an office Ethernet, over a provider's fibre backbone, to a home Wi-Fi network. Routers also separate broadcast domains, since broadcasts do not pass through them, provide the natural checkpoint for access-control lists, NAT and firewall functions, and choose among redundant paths, making the meshed topology of Question 1 usable. Their cost for this intelligence is processing delay and price above those of switches, and a need for configuration. Every Internet connection, from a home broadband router to the great backbone routers of providers, rests on this single functionality.

(c) Bridge

A bridge is the original two-port (or few-port) Layer 2 device for interconnecting LAN segments into a single larger LAN. Its function is traffic separation: it learns, from source addresses, which stations lie on each side of it, and it forwards a frame to the other segment only if the destination station actually lives there; local traffic stays local. In this way a bridge relieves congestion on each segment, extends a LAN's physical reach beyond the limits of a single segment, and can join segments of different media — historically even different LAN technologies through translational bridging. Bridges also forward or filter using the spanning tree protocol (IEEE 802.1D), which disables selected bridge ports to prevent the loops in which broadcast frames would otherwise circulate forever. The modern verdict is that the bridge's functionality has been absorbed: a switch is precisely a multiport bridge implemented in fast hardware, and in current catalogues 'bridge' survives mainly as the name of the function, in the spanning-tree standards switches still run, and in wireless bridging, where two access points link buildings across a gap no cable may cross. Functionally, then: bridge = segment-level forwarding by MAC address, in its minimal two-port form.

(d) Proxy Server

A proxy server is an application-layer (Layer 7) intermediary that acts on behalf of clients: instead of connecting directly to a destination server, the client sends its request to the proxy, the proxy makes the request to the destination in its own name, receives the reply, and returns it to the client. The destination server therefore sees and talks to the proxy, not to the individual client. From this single arrangement flow several valuable functions. Caching is the

most famous: a web proxy stores copies of popular pages and objects, so repeated requests are served locally, saving bandwidth and time. Filtering and access control allow an organisation to permit only authorised users, block forbidden sites and log usage. Anonymity and address conservation follow because many internal clients appear to the Internet as the proxy's single public address (a function related to NAT), and security improves because internal machines are never directly exposed to outside contact. Proxies exist for many protocols — HTTP/HTTPS web proxies are the standard example, with mail and FTP proxies also common — and a reverse proxy performs the same trick in the opposite direction, standing in front of servers to balance load and shield them. The limitations are the mirror of the mechanism: the proxy is a single point of failure and a potential bottleneck, it must understand each application protocol it serves, end-to-end encryption limits what it may inspect or cache, and its logs concentrate sensitive records of user activity that must be protected.

Table 1: The four devices compared by functionality

Device	OSI layer	Address used	Core function	Scope
Bridge	2 — Data Link	MAC address	Forwards frames between two LAN segments only when the destination is there	One LAN, extended
Switch	2 — Data Link	MAC address	Forwards frames to the single	One LAN / VLAN set

			correct port; one collision domain per port	
Router	3 — Network	IP address	Chooses the best path and forwards packets between different networks	Between networks / the Internet
Proxy Server	7 — Application	Application request (e.g. URL)	Terminates client requests and re-issues them in its own name; cache, filter, log	Between clients and application servers

USMAN NOTES www.usmannotes.com

Conclusion

In summary, the four devices form a ladder of intelligence. The bridge joins two segments and keeps local traffic local; the switch generalises the bridge to many fast ports and becomes the centre of the modern LAN; the router climbs a layer to read IP addresses and route packets between entirely different networks, providing the Internet's connective tissue; and the proxy server climbs to the top layer, ending connections rather than forwarding them so that it can cache, filter and represent its clients. Choosing among them is choosing the layer at which a problem lives: segment congestion belongs to bridges and switches, inter-network reachability to routers, and control over application access to the proxy.