Link Layer And LANs

Arnav Gupta

October 23, 2024

Contents

1 Introduction

Link layer is responsible to transfer a datagram from one node to a physically adjacent node over a link.

Two types of links: point-to-point and point-to-multipoint (shared medium)

Each link protocol provides different services:

- framing, link access (compulsory)
  - encapsulate datagram into frame, adding header, trailer
  - channel access if shared medium
  - MAC addresses in frame header identify source and destination
- error detection
  - errors caused by signal attentuation and noise
  - receiver detects errors, signals retransmission or drops frame
- error correction
  - receiver identifies and corrects bit errors without retransmission
- reliable delivery between adjacent nodes
  - seldom used on low bit-error links
  - always used on high-error rate wireless links
- flow control
  - pacing between sending and receiving nodes

Link layer is implemented in every host, switch, and router, in a **network interface card** (NIC) or on a chip that attaches into host system buses. Only one NIC per link.

Communication interfaces:

- sending side

  - encapsulates datagram in frame
  - adds error checking bits, reliable data transfer, flow control, and more
- receiving side
  - looks for errors, reliable data transfer, flow control, and more
  - extracts datagram from frame, passes to upper layer at receiving side

2 Framing, Error Detection, Correction

Frame contains datagram with header, footer, and flags. Header and footer carry link layer protocol info. Flags help to extract a frame from a bit stream.

Bit stuffing can be used:

- use special bit patterns for the flag and idle
- have one flag at the start of data and one at the end of data
- if the flag or idle pattern seem to show up in data, insert extra bits and remove at receiver

Noise and interference cause errors, which often come in bursts.

Bit Error Rate (BER) represents how often errors occur on average.

2.1 Error Detection

Performed at link, transport, and network layers.

Uses error detection bits (redundancy), preferably a large field of error detection for better detection. The protocol may miss some errors, though this is rare.

Can use parity checking (single bit for detection or 2D bit for detection and correction) since errors come in bursts.

Cyclic Redundancy Check (CRC) is more powerful and uses a bit pattern G called the generator with $r + 1$ bits (along with the data D):

- the sender computes r CRC bits R such that $\langle D, R\rangle = D \times 2^r \text{XOR} R$ is exactly divisible by $G \pmod 2$

  - R is the remainder of the division (mod 2) of $D \times 2^r$ by G
- the receiver knows G and divides $\langle D, R \rangle$ by G
  - if this has a non-zero remainder, errors are detected

CRC can detect all burst errors less than $r + 1$ bits (and any two errors), so it is widely used in practice.

Correction technique is to correct error at the receiver, but this is computationally expensive and has high overhead. This may be useful for real-time communication, long pipes, or noisy channels.

3 Retransmission Mechanisms

Reliable service is an abstraction where, to the network layer, the link layer is a reliable channel. Specifically, a link is a reliable channel.

The complexity of reliable data transfer depends on the characteristics of an unreliable channel, whether it loses, corrupts, or reorders data.

A retransmission mechanism should deliver data to the receiver without error, in the right order without any duplicate.

The sender and receiver do not know the other's state, unless this is communicated via a message on the same unreliable channel.

The unreliable channel may flip bits in the frame or lose frames, though frames have a CRC.

Only unidirectional data transfer will be considered, though control info will flow in both directions.

3.1 Stop and Wait

Sender sends one packet (frame) and waits for receiver response.

This needs:

- **acknowledgements (ACKs)**: receiver tells sender packet received
- **negative acknowledgements (NAKs)**: receiver tells sender that packet had errors (sender retransmits the packet on receipt of NAK)

- **sequence number (SN)**: sender tells receiver which packet this is in the sequence of packets to be sent and receiver tells sender which packet this ACK is for
- **timeout**: how long to wait before retransmission

A frame containing a packet or an ACK can be corrupted, lost, or delayed.

With this, the ACK is piggybacking, with separate SN fields for data and ACk in the header.

3.1.1 Performance

U_{sender} is **utilization**: the fraction of time the sender is busy sending

G is **goodput**: the number of successful data received per second (packets per second or bits per second)

If no error, the utilization is

$$\frac{L/R}{A/R + L/R + (2 \times \text{prop})}$$

where R is the link rate, L is the packet length, A is the ACK length, and prop is the propagation delay.

For stop and wait, the transmitter is mostly idle and goodput is low.

To tune the timeout perfectly, the timeout must be

$$\gamma = (2 \times \text{prop}) + \frac{A}{R}$$

A frame is received and acknowledged correctly after

$$t = \frac{L}{R} + \gamma$$

if the frame and its ACK are not corrupted. Otherwise, it takes kt time where $k - 1$ is the number of retransmissions.

Stop and wait limits the performance of underlying infrastructure (channel).

3.2 Pipelining

Sender allows multiple yet-to-be-acknowledged packets

Retransmission strategies for pipelining are:

- **go-back-N**: receiver only accepts the next expected packet, no extra buffering at the receiver
- **selective repeat**: receiver accepts and stores out of order packets, extra buffering

There is a tradeoff here between network traffic and buffer space and complexity. These both require a larger range of sequence numbers.

Pipelined retransmission mechanisms are complex but have high performance.

3.2.1 Go-Back-N

Sender can have up to N unacknowledged packets in the pipeline. The receiver only sends cumulative ACKs and does not keep packets out of order. The sender has a timer for the oldest unacknowledged packets, and if the timer expires, retransmit all unacknowledged packets.

Cumulative ACK: ACKs all packets up to and including the sequence number, and once received move window forward to begin at the next number

The timer is for the oldest in-flight packet, and so the timeout retransmits that packet and all higher sequence number packets in the window.

ACK-only: always send ACK for the correctly-received packet so far, with the highest in-order sequence number (no jumps or skips). This can generate duplicate ACKs.

If an out-of-order packet is received it is discarded and the highest in-order sequence number is re-ACKed.

Given a sender window size of $N = 2^n$, $m > N$ distinct sequence numbers are needed, which is at least $n + 1$ bits.

3.2.2 Selective Repeat

Sender can have up to N unacknowledged packets in the pipeline. The receiver keeps packets out of order and acknowledges individual frames. The sender maintains a timer for each unacknowledged packet, and if the timer expires only retransmit the single unacknowledged packet.

The sender window is N consecutive sequence numbers, and this limits the sequence numbers of sent, unACKed packets.

The sender checks if the next available SN is in the window and sends the packet accordingly. If a timeout occurs, only that packet is resent. Upon receiving an ACK, the packet is marked as received and if it is the smallest unACKed packet, the window is advanced one step forward.

The receiver sends an ACK for the SN when it receives a packet. If this is out of order, the packet remains in the buffer. If this is in order, the packet is then delivered (as well as any buffered in-order packets), and the window is advanced to the next not-yet-received packet. If the packet has already been received and is in the window, just return an ACK for that SN. Otherwise, ignore the packet.

Given a sender window size of $N = 2^n$, $m > 2N - 1$ distinct SNs are needed.

4 Multiple Access Protocols

Two types of links:

- **point-to-point**: between Ethernet switch and host or fiber between routers
- **broadcast (shared wire or medium)**
  - old-fashioned Ethernet
  - cable-based access networks
  - wireless LAN, cellular, satellite

Multiple Access Mechanism:

- distributed algorithm that determines how nodes share a channel
- communication about channel sharing must use the same channel as well

Multiple access mechanisms use a single shared broadcast channel, which can cause **collisions** if a node received 2+ signals at the same time.

The ideal (nonexistent) multiple access mechanism:

- given: a multipoint link of rate R bps
- desiderata:
  - when a single node wants to transmit, it can send at rate R

  - when M nodes want to transmit, each can send at average rate R/M
  - fully decentralized
    * no special node to coordinate transmissions
    * no synchronization of clocks, slots
  - simple

Two broad classes of MAC mechanisms:

- scheduling via a control node (**centralized**)
- **random access**: allow collisions and recover
  - when a node has a packet to send, it transmits at full channel data rate R, so no coordination
  - protocols specify:
    * degree of politeness
    * how to detect collisions
    * how to recover from collisions

4.1 Scheduling Using Time Division Multiple Access

Time is divided into time slots and there is a repeating cycle. The controller allocates time slots to stations. When a station's frame occurs, it can broadcast.

4.2 Scheduling Using Frequency Division Multiple Access

Channel spectrum is divided into frequency bands. Each station can broadcast in their allocated frequency band.

4.3 OFDMA

Orthogonal Frequency-Division Multiple Access: mix of TDMA and FDMA, each station has some frequency band(s) and time band(s).

4.4 Polling

Used before scheduling, but still used in Bluetooth.

Master invites nodes to transmit in turn.

Has polling overhead, latency, and single point of failure (master).

4.5 Slotted ALOHA

Assume the following:

- all L2 frames are the same size
- time is divided into equally sized slots, where each is the time to transmit a single L2 frame
- nodes start to transmit at slot beginning and are synchronized
- if 2+ nodes transmit in a slot, all nodes detect collision before the next slot

When a node obtains a new frame, it transmit it until success:

- *if no collision*: the node can move onto the next frame (if it exists)
- *if collision*: node retransmits the frame in each subsequent slot with probability p until success

Advantages:

- single active node continuously transmit at the full rate of the channel
- highly decentralized: only slots need to be synchronized
- simple

Disadvantages:

- collisions waste slots
- idle slots are likely
- nodes may be able to detect collision less than time to transmit packet
- must have clock synchronization

Efficiency: long-run fraction of successful slots

Max efficiency is $1/e = 0.37$, so at best, the channel is used for useful transmissions 37% of the time.

4.6 Pure ALOHA

No synchronization so no slots. When the frame arrives, transmit immediately.

Collision probability increases since the frame will collide with any frames sent before or after it within the time it takes to send a single frame.

Max efficiency is 18%.

4.7 Carrier Sense Multiple Access (CSMA)

Listen before transmit; if the channel is busy, defer transmission until it is free

Collisions can still occur since propagation delay means that two nodes may not hear transmissions from the other started around the same time.

If a collision occurs, the entire packet transmission time was wasted. This depends on the distance and propagation delay.

4.7.1 CSMA/CD

CSMA with collision detection:

- collisions are detected quickly
- upon collision, abort transmission

Collision detection is easier in wired LANs since signal strengths can be measured so that transmitted and received signals can be compared.

Let T_p be the maximum propagation delay and T_C be time after which everyone is aware of the collision. The minimum frame size so that the first sender can detect that there is a collision is $F_{min}/R > T_C$, where R is the rate.

After aborting, the NIC enters **binary (exponential) backoff**, so it creates time-slots of 512 bit times. After the collision m for $m < 16$, the NIC chooses a random number between 0 and $2^m - 1$ and waits 512 times this number before retrying. After collision 16, the NIC aborts sending the frame.

Consider a maximum propagation delay between 2 nodes of T_p and transmission time for the max-size frame $\$t_{\text{trans}}$. The efficiency is

$$\frac{1}{1 + 5T_p/t_{trans}}$$

As T_p goes to 0 and as t_{trans} goes to ∞, the efficiency goes to 1.

Much better performance than ALOHA, and simple, cheap, and decentralized.

4.8 Cable Access Network

Can use FDM, TDM, and random access.

Uses multiple downstream (broadcast) FDM channels (up to 1.6 Gbps/channel). A single cable modem termination system transmits into channels.

Uses multiple upstream channels (up to 1 Gbps/channel) that use multiple access: all users content for certain upstream mini time slots to send requests, others use time slots that are assigned through a map.

5 LANs

LANs are cheap, easy to install, and can be large.

Ancient LANs were a shared link. Modern LANs are many shared links connected by L2 switches.

5.1 Addressing, ARP

5.1.1 MAC Address

Each interface on a LAN has a unique 48-bit MAC address. This is usually burned into NIC ROM, but can also be set with software sometimes.

Allocation is administered by IEEE. MAC flat address can move from one LAN to another, so it is portable.

The broadcast address is FF-FF-FF-FF-FF-FF.

MAC addresses are used locally to get frames from interface to another physically-connected interface.

5.1.2 Address Resolution Protocol (ARP)

Each node on a LAN has an ARP table that has:

- IP/MAC address mappings for some LAN nodes
- **TTL (time to live)** for each node, which is the time after which the address mapping will be forgotten

To get a MAC address, a node will broadcast an ARP query containing the related IP address and another node will reply with the relevant MAC address.

5.1.3 Routing to Another Subnet

To send a datagram from A to B via router R, assuming:

- A knows the IP address of B
- A knows the IP address of R
- A knows the MAC address of R

A creates an IP datagram with IP source A and destination B. A then creates a link-layer frame containing this datagram, with the MAC address of R as the frame's destination.

Upon receipt at R, R repeats the same process to B.

5.2 Ethernet

Standards define:

- network topology
- endpoint addressing scheme
- frame (packet) format
- media access mechanism
- physical layer aspects and wiring

Standards issued by IEEE, IETF, ITU, ISO, and W3C.

IEEE Project 802 focuses on L1 and L2, dividing L2 into *logical link control* (LLC) and *media access control* (MAC).

Ethernet is the dominant wired LAN technology. It is simple, cheap, high speed and uses a single chip to produce multiple speeds.

Bus: all nodes in the same collision domain

Switched: active link-layer 2 switch in center, with each spoke running a separate Ethernet protocol (no collision)

The Ethernet frame structure involves encapsulating IP datagram in Ethernet frame with a preamble (used to synchronize receiver and sender), destination address, source address, type (higher layer protocol), data, and CRC (4 bytes).

Addresses are destination MAC addresses.

The minimum frame size is 64 bytes and maximum datagram size is 1500 bytes.

Connectionless: no handshaking between sending and receiving NICs

Unreliable: receiving NIC doesn't send ACKs or NAKs, so data in dropped frames can only be recovered at higher levels

MAC protocol is unslotted CSMA/CD with binary backoff.

5.3 Switches

Link-layer device that takes an active role by storing and forwarding Ethernet frames.

Switches examine incoming frame's MAC address, and selectively forwards frame to 1+ outgoing links using CSMA/CD to access each link.

Transparent: hosts unaware of presence of switches

Plug-and-play: switches do not need to be configured

Hosts have dedicated, direct connection to switch which switches buffer packets.

The Ethernet protocol is used on each incoming link so there are no collisions (full duplex) and each link is in its own collision domain.

Transmissions can occur simultaneously without collisions as long as there is no overlap in the interfaces used.

Each switch has a **switch table** where each entry has:

- MAC address of host
- interface to reach host
- timestamp

The switch learns which hosts can be reached through which interfaces. It learns the location of the sender from the incoming LAN segment and records the sender/location pair in the switch table.

To learn the location of the destination, it sends a flood.

Self-learning switches can be connected together.

5.3.1 Switches vs Routers

Both are store and forward and have forwarding tables.

Router: network-layer device (examine network-layer headers)

Switch: link-layer device (examine link-layer headers)

Routers compute tables using routing algorithms and IP addresses. Switches learn forwarding table through flooding, learning, and MAC addresses.