Chapter 5: The Data Link Layer

Our goals:
- understand principles behind data link layer services.
  - error detection, correction
  - framing, link access
  - channel access if sharedmedium
  - link layer addressing
  - reliable data transfer, flow control, store
- instantiation and implementation of various link layer technologies

Link Layer: Introduction

Some terminology:
- Hubs and routers are nodes
- Communication channels that connect adjacent nodes along communication path are links
- Wired links
- Wireless links
- Links
- Layer-2 packet is a frame, encapsulated datagram

data-link layer has responsibility of transferring datagram from one node to adjacent node over a link

Link Layer Services

- Framing, link access:
  - encapsulate datagram into frame, adding header, trailer
  - channel access if shared medium
  - MAC addresses used in frame headers to identify source, dest
    - different from IP address?
  - Reliable delivery between adjacent nodes
    - we learned how to do this already (chapter 3)
    - Modem used on low bit error link (fiber, some twisted pairs)
    - Wireless links, high error rates
    - Q: why both link-level and end-end reliability?
Link Layer Services (more)

- Flow Control
  - spacing between adjacent sending and receiving nodes
- Error Detection
  - errors caused by signal attenuation, noise
  - receiver detects presence of errors
- Error Correction
  - receiver identifies and corrects bit error(s) without resending to retransmit
- Half-duplex and full-duplex
  - in half-duplex, nodes at both ends of link can transmit, but not at same time

Adapters Communicating

- Link layer implemented in "adapter" (also NIC)
  - Ethernet card, PCMCIA card, 802.11 card
- Reception side
  - looks for errors, not, flow control, etc.
- Sending side
  - encapsulates diagram, passes to moving node
- half-duplex/semi-autonomous
- Link & physical layers

Link Layer

- 5.1 Introduction and services
- 5.2 Error detection and correction
- 5.3 Multiple access protocols
- 5.4 Link Layer Addressing
- 5.5 Ethernet

Error Detection

EDC: Error Detection and Correction bits (redundancy)
- Data protected by error checking may include header fields
- Error detection not 100% reliable
- Physical layer may also use same errors, but mostly
- Longer EDC yield better detection and correction

Parity Checking

Two Dimensional Bit Parity:
- Detect and correct single bit errors

Internet Checksum

Goal: Detect "errors" (e.g., flipped bits) in transmitted segment (note used at transport layer only)

Sender
- treat segment contents as sequence of 16-bit integers
- checksum addition (1's complement sum) of segment contents
- sender puts checksum value into UDP checksum field

Receiver
- compute checksum of received segment
- check if computed checksum equals checksum field value:
  - NO = error detected
  - YES: no error detected, but maybe errors elsewhere!
  - More later...
Checksumming: Cyclic Redundancy Check

- View data bits, D, as a binary number
- Choose a 1-bit pattern (generator), G
- Use G to XOR CRC bits, R, such that:
  1. R is exactly divisible by R (modulo-2)
  2. Receiver knows G, divides by R, checks for zero remainder (error detected)
- Can detect all burst errors less than 11 bits
- Widely used in practice (ATM, Ethernet)

\[ D \cdot \overline{G} \XOR R \]

CRC Example

\[ \text{Want:} \]
\[ D \cdot \overline{G} \XOR R = 0 \]
\[ \text{equivalently:} \]
\[ D \cdot \overline{G} = R \]

Link Layer

- 5.1 Introduction and services
- 5.2 Error detection and correction
- 5.3 Multiple access protocols
- 5.4 Link Layer Addressing
- 5.5 Ethernet

Multiple Access Links and Protocols

Two types of "links":
- Point-to-point
- Broadcast (shared wire or medium)

\[ \text{Ideal Multiple Access Protocol} \]

- Broadcast channel of rate \( R \) bps
  1. When one node wants to transmit, it can send at rate \( R \).
  2. When \( M \) nodes want to transmit, each can send at average rate \( R/M \).
- Fully decentralized:
  1. No special node to coordinate transmissions
  2. No synchronization of clocks, slates
- Simple
MAC Protocols: a taxonomy

Three broad classes:

- Channel Partitioning
  - divide channel into smaller "pieces" (time slots, frequency, codes)
  - allocate pieces to nodes for exclusive use
- Random Access
  - channel not divided, allow collisions
  - "recovery" from collisions
- "Taking turns"
  - Nodes take turns, but nodes with more to send can take longer turns

Channel Partitioning MAC protocols: TDMA

TDMA: time division multiple access
- access to channel in "rounds"
- each station gets fixed length slot (length = pkt trans time) in each round
- unused slots go idle
- example: 6-station LAN, 1,3,4 have pkt, slots 2,5,6 idle

Channel Partitioning MAC protocols: FDMA

FDMA: frequency division multiple access
- channel spectrum divided into frequency bands
- each station assigned fixed frequency band
- unused transmission time in frequency bands go idle
- example: 6-station LAN, 1,3,4 have pkt, frequency bands 2,5,6 idle

Random Access Protocols

- When node has packet to send
  - transmit at full channel data rate R
  - no a priori coordination among nodes
- two or more transmitting nodes "collision"
- random access MAC protocol specifies:
  - how to detect collisions
  - how to recover from collisions (e.g., via delayed retransmissions)
- Examples of random access MAC protocols:
  - slotted ALOHA
  - ALOHA
  - CSMA, CSMA/CD, CSMA/CA

Slotted ALOHA

Assumptions
- all frames same size
- time is divided into equal size slots, time to transmit 1 frame
- nodes start to transmit frames only at beginning of slots
- nodes are synchronized
- if 2 or more nodes transmit in slot, all nodes detect collision

Operation
- when node obtains fresh frame, it transmits in next slot
- no collision, node can send new frame in next slot
- if collision, node retransmits frame in each subsequent slot with prob. until success

Pros
- single active node can continuously transmit at full rate of channel
- highly decentralized: only slots in nodes need to be in sync
- simple

Cons
- collisions, wasting slots
- idle slots
- nodes may be able to detect collision in less than time to transmit packet
- clock synchronization
**Slotted Aloha efficiency**

- Efficiency is the long-run fraction of successful slots when there are many nodes, each with many frames to send.
- Suppose $N$ nodes with many frames to send, each transmits in slot with probability $p$.
- Prob that node 1 has success in a slot: $p(1-p)^{N-1}$.
- Prob that any node has a success: $Np(1-p)^{N-1}$.

**Pure Aloha efficiency**

- $P(\text{success given node}) = P(\text{node transmits})$.
- $P(\text{any other node transmits in } [a, b]) = (1-p)^N$.
- $P(\text{other node transmits in } [a, b]) = (1-p)^N$.
- $P(\text{success given node}) = p(1-p)^N$.
- Choosing optimum $p$ and letting $N \to \infty$.
- Even worse: $1/(2e) = 0.368$.

**CSMA (Carrier Sense Multiple Access)**

- **CSMA** listens before transmit:
  - If channel sensed idle: transmit entire frame
  - If channel sensed busy: defer transmission
- Human analogy: don’t interrupt others!

**Pure (unslotted) ALOHA**

- Unslotted Aloha: simpler, no synchronization
- When frame first arrives:
  - Transmit immediately
- Collision probability increases:
  - Frame sent at $t_0$ collides with other frames sent in $[t_0, 1/e]$.

**CSMA/CD (Collision Detection)**

- **CSMA/CD**: carrier sensing, deferral as in CSMA
  - Collisions detected within short time
  - Colliding transmissions aborted, reducing channel waste
- Collision detection:
  - Easy in wired LANS: measure signal strengths, compare transmitted, received signals
  - Difficult in wireless LANS: receiver shuts off while transmitting
- Human analogy: the polite conversationalist
**CSMA/CD collision detection**

- Diagram showing collision detection.

**"Taking Turns" MAC protocols**

- Channel partitioning MAC protocols:
  - shared channel efficiently and fairly at high load
  - inefficient at low load: delay in channel access, 1/N bandwidth allocated even if only 1 active node

- Random access MAC protocols:
  - efficient at low load: single node can fully utilize channel
  - high load: collision overhead

- "Taking Turns" protocols look for best of both worlds

**"Taking Turns" MAC protocols**

- Polling:
  - master node "invites" slave nodes to transmit in turn
  - concerns:
    - polling overhead
    - latency
    - single point of failure (master)

- Token passing:
  - control token passed from one node to next sequentially
  - token message
  - concerns:
    - token overhead
    - latency
    - single point of failure (token)

**Summary of MAC protocols**

- What do you do with a shared media?
  - Channel Partitioning, by time, frequency or code
    - Time Division Frequency Division
  - Random partitioning (dynamic),
    - ALOHA, S-ALOHA, CSMA, CSMA/CD
    - carrier sensing easy in some technologies (wire), hard in others (wireless)
    - CSMA/CD used in Ethernet
    - CSMA/CD used in 802.11
  - Taking Turns
    - polling from a central site, token passing

**LAN technologies**

- Data link layer so far:
  - services, error detection/correction, multiple access

- Next: LAN technologies
  - addressing
  - Ethernet
  - hubs, switches
  - ppp

**Link Layer**

- 5.1 Introduction and services
- 5.2 Error detection and correction
- 5.3 Multiple access protocols
- 5.4 Link Layer Addressing
- 5.5 Ethernet

- 5.6 Hubs and switches

- 5.8 Link Virtualization: ATM
MAC Addresses and ARP

- 32-bit IP address:
  - network-layer address
  - used to get datagram to destination IP subnet
- MAC (or LAN or physical or Ethernet) address:
  - used to get frame from one interface to another physically-connected interface (same network)
  - 48-bit MAC address (for most LANs)
    - burned in the adapter ROM

LAN Address (more)

- MAC address allocation administered by IEEE
  - manufacturer buys portion of MAC address space to assure uniqueness
  - Analogy:
    - (a) MAC address like Social Security Number
    - (b) IP address like postal address
  - MAC flat address portability
    - can move LAN card from one LAN to another
  - IP hierarchical address NOT portable
    - depends on IP subnet to which node is attached

ARP: Address Resolution Protocol

- Each IP node (host, router) on LAN has ARP table
- ARP Table IP/MAC address mappings for some LAN nodes
- TTL address, MAC address, TTL
  - TLI (Time To Live) time after which address mapping will be forgotten (typically 20 min)

ARP protocol: Same LAN (network)

- A wants to send datagram to B, and B's MAC address not in A's ARP table
- A broadcasts ARP query packet containing B's IP address
- Duplicate MAC address = FF-FF-FF-FF-FF-FF
  - all machines on LAN receive ARP query
- B receives ARP packet, replies to A with its 6 bytes MAC address
- frame sent to A, MAC address (encaps)

Routing to another LAN

- walkthrough: send datagram from A to B via R
- assume A knows B's IP address
  - Two ARP tables in router R, one for each IP network (LAN)
Link Layer

- 5.1 Introduction and services
- 5.2 Error detection and correction
- 5.3 Multiple access protocols
- 5.4 Link Layer Addressing
- 5.5 Ethernet

Ethernet

"Dominant" wired LAN technology:
- cheap $20 for 100Mbps
- first widely used LAN technology
- simpler, cheaper than token LANs and ATM
- kept up with speed race: 10 Mbps - 10 Gbps

Star topology

- Bus topology popular through mid 90s
- Now star topology prevails
- Connection choices: hub or switch (more later)

Ethernet Frame Structure

Sending adapter encapsulates IP datagram (or other network layer protocol packet) in Ethernet frame.

Preamble:
- 7 bytes with pattern 10101010 followed by one byte with pattern 10101011
- used to synchronize receiver, sender clock rates

Ethernet Frame Structure (more)

- Addresses: 6 bytes
  - if adapter receives frame with matching destination address, or with broadcast address (eg ARP packet), it passes data to frame to network layer protocol
  - otherwise, adapter discards frame
- Type: indicates the higher layer protocol (mostly IP but others may be supported such as Novell SPX and AppleTalk)
- CRC: checked at receiver, if error is detected, the frame is simply dropped
Unreliable, connectionless service

- Connectionless: No handshaking between sending and receiving adapter.
- Unreliable: Receiving adapter doesn’t send acks or nacks to sending adapter.
- A stream of datagrams passed to network layer can have gaps.
- Gaps will be filled if app is using TCP.
- Otherwise, app will see the gaps.

Ethernet uses CSMA/CD

- No slots.
- Adapter doesn’t transmit if it senses that some other adapter is transmitting, that is, carrier sense.
- Transmitting adapter aborts when it senses that another adapter is transmitting, that is, collision detection.

Before attempting a retransmission, adapter waits a random time, that is, random access.

Ethernet CSMA/CD algorithm

1. Adapter receives datagram from network layer & creates frame.
2. If adapter senses channel idle, it starts to transmit frame. If it senses channel busy, waits until channel idle and then transmits.
3. If adapter transmits entire frame without detecting another transmission, the adapter is done with frame.
4. If adapter detects another transmission while transmitting, aborts and sends jam signal.
5. After aborting, adapter enters exponential backoff. After the $m$th collision, adapter chooses a $K$ at random from $(0,1,2,\ldots,2^{m})$. Adapter waits $K \cdot 2^{m}$ bit times and returns to Step 2.

Ethernet's CSMA/CD (more)

- Jam Signal: A signal all other transmitters are aware of collision. 48-bit.
- Bit time: 1 microsecond for 10 Mbps Ethernet.
- For $K$-1023, wait time is about 50 microseconds.
- Exponential Backoff:
  - Ideal: After backoff, tries to transmit.
  - Actual: After collision, waits $K \cdot 2^{m}$ bit times.

See/Interact with Java applet on Web site. Highly recommended.

CSMA/CD efficiency

- $T_{prop} = \text{max prop between 2 nodes in LAN}$
- $T_{trans} = \text{time to transmit max-size frame}$

Efficiency $\frac{1}{1 + \frac{T_{prop}}{T_{trans}}}$

- Efficiency goes to 1 as $T_{prop}$ goes to 0.
- Goes to 1 as $T_{trans}$ goes to infinity.
- Much better than ALOHA, but still decentralized, simple, and cheap.

10BaseT and 100BaseT

- 10/100 Mbps rate, latter called "fast ethernet".
- T stands for Twisted Pair.
- Nodes connect to a hub: "star topology": 100 m max distance between nodes and hub.

DIAGRAM: A hub with twisted pair cables connected to several nodes.
Hubs

- Hubs are essentially physical-layer repeaters:
  - Bits coming from one link go out all other links at the same rate.
  - No frame buffering.
  - No CSMA/CD at hub, adapters detect collisions.
  - Provides net management functionality.

Manchester encoding

- Used in 10BaseT.
- Each bit has a transition.
- Allows clocks in sending and receiving nodes to synchronize to each other.
- No need for a centralized global clock among nodes.
- Hey, this is physical-layer stuff!

Gbit Ethernet

- Uses standard Ethernet frame format.
- Allows for point-to-point links and shared broadcast channels.
- In shared mode, CSMA/CD is used; short distances between nodes required for efficiency.
- Uses hubs, called here "Buffered Distributors".
- Full-Duplex at 1 Gbps for point-to-point links.
- 10 Gbps now!

Link Layer

- 5.1 Introduction and services
- 5.2 Error detection and correction
- 5.3 Multiple access protocols
- 5.4 Link layer: Addressing
- 5.5 Ethernet

Interconnecting with hubs

- Backbone hub connects LAN segments.
- Extends max distance between nodes.
- But individual segment collision domains become one large collision domain.
- Can't interconnect 10BaseT & 100BaseT.

Switch

- Link layer device.
  - Stores and forwards Ethernet frames.
  - Examines frame header and selectively forwards frame based on MAC dest address.
  - When frame is to be forwarded on segment, uses CSMA/CD to access segment.
  - Transparent.
  - Hosts are unaware of presence of switches.
  - Plug-and-play, self-learning.
  - Switches do not need to be configured.
**Forwarding**

- How do determine onto which LAN segment to forward frame?
- Looks like a routing problem...

**Filtering/Forwarding**

When switch receives a frame:

index switch table using MAC dest address
if entry found for destination
   if dest on segment from which frame arrived
      then drop the frame
   else forward the frame on interface indicated
else flood
   forward on all but the interface on which the frame arrived

**Switch example**

Suppose C sends frame to D

Switch receives frame from from C
- noted in bridge table that C is a destination
- because C is in table, switch forwards frame into interface 1
- frame received by D

**Self learning**

- A switch has a switch table
- entry in switch table:
  - (MAC Address, Interface, Time Stamp)
  - stale entries in table dropped (TTL can be 60 min)
- switch learns which hosts can be reached through which interfaces
- when frame received, switch "learns" location of sender: incoming LAN segment
- records sender/location pair in switch table

**Switch: traffic isolation**

- switch installation breaks subnet into LAN segments
- switch filters packets:
  - some LAN segment frames not usually forwarded onto other LAN segments
  - segments become separate collision domains
**Switches: dedicated access**

- Switch with many interfaces
- Hosts have direct connection to switch
- No collisions; full duplex

Switching A-to-A and B-to-B simultaneously, no collisions

**More on Switches**

- Cut-through switching: frame forwarded from input to output port without first collecting entire frame
- Slight reduction in latency
- Combinations of shared/dedicated, 10/100/1000 Mbps interfaces

---

**Institutional network**

**Switches vs. Routers**

- Both store-and-forward devices
  - Routers network layer devices (examine network layer headers)
  - Switches are link-layer devices
- Routers maintain routing tables, implement routing algorithms
- Switches maintain switch tables, implement filtering, learning algorithms

---

**Summary comparison**

<table>
<thead>
<tr>
<th></th>
<th>Hubs</th>
<th>Routers</th>
<th>Switches</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traffic</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Isolation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Plug &amp; Play</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Optimal Routing</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Cut Through</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

---

**Chapter 5: Summary**

- Principles behind data link layer services
  - Error detection, correction
  - Sharing a broadcast channel: multiple access
- Link layer addressing
- Instantiation and implementation of various link layer technologies
  - Ethernet
  - Switched LANs
  - SONET
- Virtualized networks on a link layer: ATM, MPLS