ECE 435 – Network Engineering
Lecture 19

Vince Weaver

https://web.eece.maine.edu/~vweaver

vincent.weaver@maine.edu

30 March 2023

Announcements

- HW#8 due tomorrow

Ethernet History

- Proposed by Bob Metcalfe in 1973
  (went on to found 3Com)
  2022 Turing Award

- Metcalfe, Boggs, Thacker, and Lampson listed on patent

- Inspired by ALOHAnet, a wireless network in Hawaii, allow users on various islands to connect to server on Oahu

- Various competing local networks, Ethernet won in the end

Token Ring (Ethernet Competitor)

- Guaranteed Deterministic Delivery (vs Ethernet: best effort)
- Dates to 1970s
- Standardized by IBM, 1984, IEEE 802.5 (note, not RFC)
- 4Mbps, eventually shielded twisted pair, eventually 16Mbps, 100Mbps and 1Gbps
- 3-byte frame passed around gives permission to transmit
- More complex, no crossover cable (direct connect two machines),

- Supports multiple identical MAC addresses
- Deterministic time to get to transmit
- Frames can have different access priorities
- Empty token passed around. If data to transmit, put in. Then passes around until it gets to receiver, removed, and back to passing empty token. When gets back to originator it knows it has been received.
- Token Bus, GM, IEEE802.4 (withdrawn) like ring, but virtual ring. Needed to know neighbors to pass token. Guaranteed worst case transmit time.

Why did Ethernet win?

- Other competitors: FDDI, ATM, DQDB.
- Why did it win? Simpler and thus cheaper.
- Why simpler?
  No priority mechanism, no QoS, no central control
- Could use cheaper twisted pair cable
- Token ring cards generally a lot more expensive than Ethernet

The Ethernet Progression

- Low speed (3Mbps) $\rightarrow$ High speed (400Gbps)
- Shared media $\rightarrow$ dedicated media
- LAN $\rightarrow$ WAN

Ethernet History

- 1972 – experimental 3Mbps
- 1981 – DIX (DEC/Intel/Xerox) ver 1 (10Mbps)
- Standardized in 1981
- 1982 – DIX ver 2

Ethernet Naming

- Naming: Speed/BROAD,BASE,PASS/PHY
- Almost all is baseband (narrow frequency, vs broadband).
- PHY originally was distance could travel (in 100m) but now medium type.

Thick Ethernet

- 1983 – IEEE 802.3/10BASE5
- "Thick Ethernet", up to 500m often yellow or orange (standard suggests yellow)
- Looks like garden hose.
- Vampire Tap, AUI connector, drill into cable, at 2.5m intervals (to avoid reflections)
- Terminated on each end One bad connection could ruin for all

Thin Ethernet

- 1985 – 10BASE2
- "thin net", thinner connections, BNC connectors, T connectors (185m, rounded up to 200m)
- 50 ohm terminator
- Issues with grounding loops (one and only one must be tied to ground, otherwise can get current flow in shielding)
- How to detect network problem? Send pulse, look for echo

10BASE-T

- 1990
- 10BASE-T – twisted pair (Cat3) , needed hub
- 1993 – 10BASE-F – fiber

Faster Ethernet

- We will talk about this in more detail later
- 1995 – 100BASE-T, 100BASE-TX 4B5B MLT-3 cat5 two twisted pairs
- 1997 – Full-duplex
- 1998/1999 – 1000BASE-TX PAM-5, four twisted pairs, can transfer in both directions on one pair using DSP/echo cancellation
- 2006 – 10GBASE-T
- 2010 – 40G and 100G

- 2017 – 400GB

“Classic” Ethernet Overview

- Not really used anymore, but a classic example of what a relatively easy-to-understand link-level interface is like

Ethernet MAC

- CSMA/CD "Carrier sense multiple access with collision detection"
- First senses cable (how? – see later)
- If busy, waits
- Sends. If collision, jams the cable aborts transmission, waits random back off time before retrying.
- Exponential backoff. Randomly choose time from 0 to

$2^k - 1$ where k is number of tries (capping at 10). Time slot is 512 bits for 10/100, 4096 for 1Gbs

- on newer full-duplex links no need for carrier sense and collision detection not needed

Ethernet Collisions

- In order to work properly, twice round-trip time needs to be less than time needed to transmit minimal (64-byte) frame, otherwise not possible to notice collision in time and frame loss

- This limits network size to collision domain

- Bits wasted is not bad, collision often caught in the preamble

Manchester Encoding

- Does not use 0V for 0 and 5V for 1.
  Why? Idle is 0, so how can you tell how many zeros at beginning of signal?
- Could use +1V/-1V, but still would need way to sync signal on long runs of 0 or 1
- Manchester encoding
  - 1 is high to low transition.
  - 0 is low to high transition.
  - Always a transition in the middle of an interval.

- Disadvantage, need twice as much bandwidth
- Differential Manchester
  - transition at start of interval means 0
  - lack of transition means 1
  - Still transition in the middle
  - More complex but better noise handling
- Ethernet uses Manchester, Token Ring uses differential Manchester
- Ethernet high 0.85V and low -0.85V

```
Manchester
  1       0    0      1       1
__   !   ___!   ___!___   !___
  |  !  |   !  |   !   |  !   |
  ---!---   !---   !   ---!   ---
     !      !      !      !

Differential Manchester
         0    0     1      1
__   !__   !__   !   __!__
  |  !  |  !  |  !  |  !  |
  ---!  ---!  ---!---  !   ---
```

Ethernet Frame layout

This is Ethernet II/DIX standard, the most common

```
 ---------- ----- ---- ---- ----- ----------- -----
| Preamble | SFD | DA | SA | T/L |    Data     | FCS |
 ---------- ----- ---- ---- ----- ----------- -----
   7 bytes     1     6    6      2    46-1500       4
```

Frame size is variable. Often first two fields are excluded and said that Ethernet frames are between 64 and 1518 bytes long

Ethernet Frame Preamble

- Fixed 1010...1010 in transmission order (LSB, least significant bit first)
- On original Ethernet this was 10MHz 6.4us pulse used to synch clocks
- PHY might do other things (100BASE-X uses 4B/5B stuff, so different pattern)

Ethernet Start Frame Delimiter (SFD)

- SFD - indicates the start of the frame
- value 10101011 in transmission order
- Original Ethernet declared 8 bytes of same pattern, but on modern first 7 bytes might be different

Ethernet MAC addresses

- DA = 48 bit destination MAC address
- SA = 48 bit source MAC address
  - First 3 bytes the OUI (organization unique identifier)
  - Next 3 bytes supposed to be a unique ID
- Ethernet packets put on the wire least-significant bit first (as if shifted right out of a shift register)
- Multicast if the "first" bit (meaning 0x1, not 0x80) is set in the first octet (e.g. 01-80-C2-00-00-00)
- Broadcast if all bits set ff:ff:ff:ff:ff:ff

Ethernet Type/Length Field

- Originally type field
- 802.3 makes it length of *data* (not length of frame)
- In 1997 802.3 approved as type too, so dual meaning
- How tell difference?
  - Max len 1500, value bigger than 0x600 (1536) is type
  - 0x0800 = IPv4, 0x86dd = IPv6, 0x0806 = ARP
  - How tell length if type? Detect end of signal or inter-frame gap, or valid checksum (this is most common usage)

◦ How tell type if length? Will have 802.2 header?

Ethernet Frame – Data

- Data – data from 46 to 1500 bytes
- Why limit 1500B? because RAM was expensive in 1978.
- If smaller than 46 bytes padded. Makes sure checksum works.
- Also if too short, could be done transmitting before a collision can be detected (light travel to furthest node and back)

Ethernet Frame Check Sequence (FCS)

- FCS – a 32-bit CRC code.
- Somewhat complicated, magic number at end 0xc704dd7b
  - Hard to get a clear description of what it looks like w/o a lot of math
  - You can implement with linear feedback shift register (shift register with xors)
  - Possibly the CRC is calculated MSB first rather than LSB first?

- If incorrect FCS, silently drops

End of Frame / Inter Packet Gap

- At end of frame, drop carrier
- More modern Ethernet might signal this with a symbol
- Inter-packet gap, 96 bits (12 bytes) of idle before sending next frame
- Gives receiver time to handle frame before next starts

ARP – address resolution protocol

- On local network, how do we find MAC address if we know IP?
  Hard-code in /etc/ethers?
  Request somehow?

- ARP (RFC826)
  - Device first checks ARP cache to see if already knows
  - Otherwise, broadcasts to ff:ff:ff:ff:ff:ff "who has this IP"

- ◦ Device reply with its IP and MAC (unicast)
- ◦ These are cached
- ◦ Timeout in case you reassign
- ◦ ARP announcement: can broadcast when your address changes so they can update (gratuitous ARP)
- ◦ Other optimizations(?)

- Used for many higher protocols, but not IPv6 which uses Neighborhood Discovery Protocol (ND)
- Security: ARP spoofing

RARP/BOOTP

- Some cases need to do RARP (Reverse ARP) (RFC 903) have own MAC, find IP (netbooting is common reason)
- ARP packets not forwarded, so extension called BOOTP that allowed network booting.
- BOOTP automated by DHCP.
- IPv6 has IND (Inverse Neighborhood Discovery Protocol)

Naming note

- Why IEEE standards start with 802
  Next available? Also co-incidentally first meeting was Feb. 1980
  For example, IEEE floating point is IEEE 754 but first meeting was not April 1975

Ethernet Transmission

- Break data into frame
- In half-duplex CSMDA/CD senses carrier. Waits until channel clear
- Wait for an inter-frame-gap (IFG) 96 bit times. Allows time for receiver to finish processing
- Start transmitting frame
- In half-duplex, transmitter should check for collision.
  Co-ax, higher voltage than normal
  For twisted pair, noticing signal on the receive while

transmitting

- If no collision, then done
- If collision detected, a *jam* signal is sent for 32-bits to ensure everyone knows. Pattern is unspecified (can continue w data, or send alternating 1s and 0s)
- Abort the transmission
- Try 16 times. If can't, give up
- Exponential backoff. Randomly choose time from 0 to $2^k - 1$ where k is number of tries (capping at 10). Time slot is 512 bits for 10/100, 4096 for 1Gbs
- Wait the backoff time then retry

Ethernet Receiving

- Physical layer receives it, recording bits until signal done. Truncated to nearest byte.
- If too short (less than 512 bits) treated as collision
- If destination is not the receiver, drop it
- If frame too long, dropped and error recorded
- If incorrect FCS, dropped and error recorded
- If frame not an integer number of octets dropped and error recorded
- If everything OK, de-capsulated and passed up

- Frame passed up (minus preamble, SFD, and often crc)
- Promiscuous mode?

Maximum Frame Rate

- 7+1 byte preamble 64-byte frame, IFG of 12 bytes between transmissions. equals 672 bits. In 100Mbps system 148,800 frames/second

Ethernet Flow Control

- Flow control is optional
- In half duplex a receiver can transmit a “false carrier” of 1010..10 until it can take more.
- Congested receiver can also force a collision, causing a backoff and resend. Sometimes called force collision
- Above schemes called “back pressure”
- For full duplex can send a PAUSE frame that specifies how much time to wait.

Full Duplex MAC (requires switch)

- Early Ethernet was coaxial in a bus
- Twisted pair has replaced this, usually in a hub/or switch star topology
- 10BASE-T and 100BASE-TX pair for transmit or receive
- inefficient. Since point to point, why do you need arbitration?
- Full-duplex introduced in 1997. Must be able to transmit/receive w/o interference, and be point to point.
- Full duplex effectively doubles how much bandwidth

between. Also it lifts the distance limit imposed by collision detection

Router vs Hub vs switch

- Hub all frames are broadcast to all others
  Bandwidth is shared (only say 100MB for all)
- Switch – direct connection, no broadcast. Has to be intelligent. Each point to point connection full bandwidth.
  no collisions. Internally either own network to handle collisions, or else just buffer RAM that can hold onto frames until the coast is clear.
- Multi-speed hubs?

When 10/100MB first came out, cheap hubs could only run at 10MB or 100MB. But switches *really* expensive. They had a compromise 10/100MB hub that internally had a hub for both then a mini-switch to bridge the gap.

- Router will move frames from one network to another
- Lights. How many ports? Uplink ports?

Direct Connection Ethernet

- Direct connect two machines with one cable
- Used to need special "crossover" cable to swap TX and RX lines
- Modern cards can detect direct connect and swap the wires for you

Ethernet Security

- Traditional hub, all machines saw all packets
- With tcpdump could monitor all packets on network, back in day all plain text. e-mail, web-browsing, chat, passwords, telnet
- tcpdump put card in “promiscuous” mode which let it intercept all packets instead of ignoring ones not to system
- Why so low security? Old day trust people at your work/office, also was probably expensive/difficult to get

an unauthorized UNIX workstation with ethernet card and root access on the local network

Power over Ethernet

- Method B: In 10/100 Base T, only 2 of the 4 pairs in Cat5 used. So send voltage down spare pairs
- Method A: send DC voltage down with the signals floating on top
- Original 44 VDC, 15.4W
- POE+ 25W
- Need special switch to send power, and device on other end has to support it.
- Raspberry Pi hat