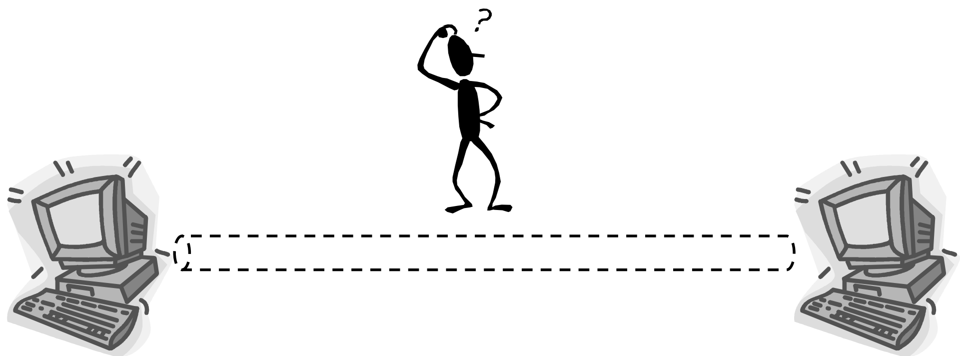
Link Layer (L2) Review

Problem

- How to connect 2 (or more) computers directly to each other?
  - physical cable?
  - bit encoding, framing, error detection?
  - reliable transfer mechanisms?
  - what if several hosts share the transmission medium?

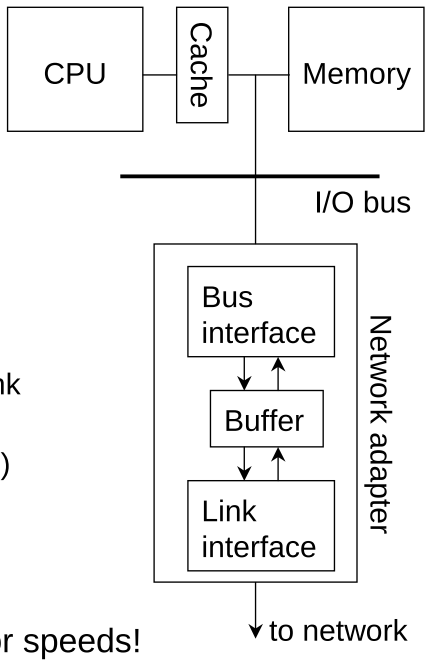


Outline

- Host HW and physical link technologies
- Encoding
- Framing
- Error detection/correction
- Reliable transmission
- Multiple access techniques
  - Ethernet
  - Token ring and FDDI
  - Wireless

Network nodes

- Node types: end hosts, routers
  - end hosts e.g. PCs, routers special purpose hw
- End host structure
  - Memory
    - packets waiting stored in memory
    - finite and (possibly) scarce
  - Network adapter
    - connects host to physical medium
    - delivers packets from memory to physical link
    - small buffer between bus i/f and link i/f (I/O bus and link operate at different speeds)
    - L2 functions implemented in adapter hw
  - Device driver
    - manages the adapter
- End hosts run at memory speeds, not processor speeds!
  - memory delay halves every 10 yrs, processor speeds double every 1.5 yrs



Physical link technologies

- Variety of physical media:
  - twisted pair (telephone line)
  - coaxial cable (TV cable)
  - optical fiber
  - wireless (infrared, microwave)

- Selection of media based on type of network
  - connections within a building/campus area
  - connections across a city/country
  - “last mile” connections

Short/long distance connections

- Connecting inside a building/campus area
  - string the nodes together by cable
  - cable type depends on technology
  - category 5 twisted pair is common indoor, fiber between buildings

Cable	Bandwidth	Distance
Cat 5 twisted pair	10-100 Mbps	100 m
Thin net coax	10-100 Mbps	200 m
Thick net coax	10-100 Mbps	500 m
Multimode fiber	100 Mbps	2 km
Single mode fiber	100-2400 Mbps	40 km

- Connecting across the country/city
  - cannot install cable yourself $\Rightarrow$ leased lines
  - rent a “logical connection” from a service provider (telephone company)

Service	Bandwidth
E1	1.920 Mbps
E3	34.3 Mbps
STM-1	155 Mbps
STM-4	620 Mbps
STM-16	2.4 Gbps

“Last mile” links (1)

- “Last mile”: last leg from home/office to a service provider’s network
- POTS
  - dial up modem connections upto 56 kbps, uses twisted pair, cheapest
  - technology at its bandwidth limit
- ISDN
  - offers two 64 kbps channels (2B+D) = max. 128 kbps
  - uses twisted pair, requires separate terminal adapter at home
  - late 70’s vision: nobody needs more than ISDN!
- xDSL (DSL=Digital Subscriber Line): collection of technologies that offer more bandwidth than ISDN over standard twisted pair
  - ADSL (Asymmetric DSL)
    - downstream (nw → user) speed 1.5 Mbps (5.5 km) - 8.4 Mbps (2.7 km)
    - upstream (user → nw) speed 16 kbps - 640 kbps
  - VDSL (Very high data rate DSL)
    - higher bw than ADSL, shorter distances (requires extra hw in subscriber loop)
    - 12.96 Mbps (1.4 km) - 55.2 Mbps (0.3 km)

“Last mile” links (2)

- Cable modems
  - alternative to xDSL, popular in the States
  - uses existing cable TV network (reaches 95% of homes in US)
  - asymmetric bw: upto 40 Mbps downstream, 20 Mbps upstream
  - shared channel: cable TV network is a tree structured distribution network
    - problems (congestion) when the number of users in the tree grows
- Wireless links
  - dial up modem connections over GSM
  - low orbit satellite network
    - Teledesic: 288 satellites connected with 155 Mbps wireless links each offering 1440 16 kbps channels (128 channels offer 1.92 Mbps)
  - WLAN & Bluetooth provide wireless access at rates 1-54 Mbps (depending on standard version) for distances of about 10 m (office room environment) in the 2.4 or 5 GHz frequency band
  - infrared and microwave point-to-point links

Outline

- Host HW and physical link technologies
- Encoding
- Framing
- Error detection/correction
- Reliable transmission
- Multiple access techniques
  - Ethernet
  - Token ring and FDDI
  - Wireless

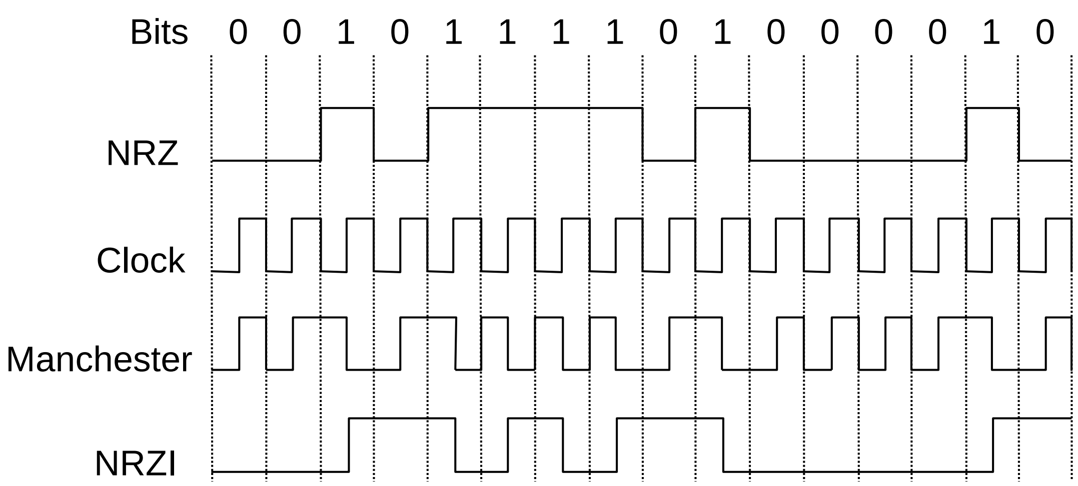
Encoding (1)

- Task:
  - encode the user's bits into electrical (optical) signals, decoding is the reverse process
- NRZ (Non-return to Zero)
  - encoding: 0 = low signal amplitude, 1 = high signal amplitude
  - long strings of consecutive 1's or 0's create problems
    - signal value stays the same for a long time period
- Problems with NRZ:
  - Clock recovery: during a string of 0's or 1's, time synchronization information is lost (signal changes used to synchronize the receiver's clock)
  - Baseline wander
    - reception of 1 or 0 is done by comparing against an average signal level that is measured
    - long string of 1's results in an increase in the average $\Rightarrow$ higher probability for false detection

Encoding (2)

- NRZI (non return to zero inverted)
  - encode a transition for each 1, do nothing for 0's
  - synchronization problem with consecutive 0's
- Manchester
  - XOR operation of the signal and clock ⇒ transition for every bit
  - problem: 2 pulses used for every transmitted bit ⇒ 50% efficiency
- 4B/5B
  - encode each received 4 bit pattern with 5 bit patterns ⇒ 80% efficiency
  - each pattern chosen s.t. at most 3 consecutive 0's possible
  - each 5 bit pattern encoded with NRZI

Encoding (3)

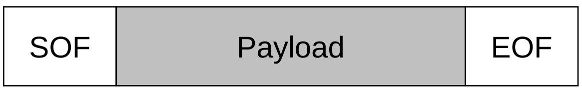


Outline

- Host HW and physical link technologies
- Encoding
- Framing
- Error detection/correction
- Reliable transmission
- Multiple access techniques
  - Ethernet
  - Token ring and FDDI
  - Wireless

Framing

- Problem:
  - applications generate (variable length) packets (called frames at layer 2)
  - how are the individual frames distinguished by end hosts?
- General approach
  - add extra uniquely distinguishable parts to the packet to such that start of frame (SOF) and end of frame (EOF) become uniquely identifiable



  - Framing error: frame boundary information is lost or corrupted
- Framing performed at every protocol layer
- Framing needed
  - for encapsulating higher layer protocol inside frame of lower layer
  - for multiplexing low speed connections onto high speed links
  - for multiplexing multiple higher layer protocols into lower layer frame

Sentinel approach to framing

- Sentinel approach
  - beginning and end of frame are signaled by use of a unique bit pattern, e.g., 01111110
- Byte (bit) stuffing: unique flag pattern can occur in the payload
  - sender: if 01111110 is part of data, add (stuff) extra 01111110 in the stream
  - receiver: if 01111110 is seen twice, remove the second
  - Result: size of frames depends on the data
- Byte oriented (e.g. PPP) or bit oriented (e.g. HDLC)
  - PPP: commonly used in dial-up modem connections
    - Flag=01111110, Address and Control contain default values, Protocol used for identifying higher level protocols
    - During PPP connection set up some protocol elements are negotiated, also configuration information is exchanged

8	8	8	8/16	N	16/32	8
Flag	Address	Control	Protocol	Payload	CRC	Flag

Other approaches to framing

- Byte counting (e.g. DDCMP)
  - include length of the payload in a field preceding actual payload
- Clock based framing (SONET, SDH)
  - addresses both encoding and framing, also multiplexing of lower bit rate links onto a single high bit rate link
  - fixed length frames (125 μs long)
    - no byte stuffing
    - frame begins with “a known” bit sequence
    - receiver expects this pattern to repeat every 125 μs (at the beginning of each frame)
    - if this happens enough often, receiver is synchronized and can interpret the frame

Outline

- Host HW and physical link technologies
- Encoding
- Framing
- Error detection/correction
- Reliable transmission
- Multiple access techniques
  - Ethernet
  - Token ring and FDDI
  - Wireless

Error detection vs. correction

- Method: add *k* redundant extra bits computed from the original message to detect (and to correct) bit error(s)
- Error detection
  - idea: $k << n$ (n = length of the original message)
  - In Ethernet $k = 32$ and $n = 1500*8 = 12\ 000$
  - bit error patterns can be detected but not corrected
  - reasonable if packets (frames) are corrupted relatively seldom
    - in modern optical networks bit error rates ~ 10^{-12}
    - corrupted frames are retransmitted (and with high probability correctly)
- Error correction
  - now: $k \sim n$
  - what errors can be corrected depends on the used algorithm (codes) and *k*
  - useful in an environment where packets are frequently corrupted
    - in wireless, bit error rates are often ~ 10^{-6} - 10^{-4}
    - ⇒ retransmitted frames encounter errors with high probability!

CRC (Cyclic Redundancy Check)

- One of the most common error detection techniques
  - used in almost all L2 protocols (HDLC, Ethernet, Token Ring, ATM)
- The method:
  - Let C(x) = divisor polynomial of degree k, e.g., x^3+x+1
  - Let T(x) = original message with k zeros appended
  - Divide T(x) with C(x) (modulo 2 logic), subtract the remainder from T(x)
    - the result is now exactly divisible with C(x)
  - Thus, if e.g. original message and the remainder are transmitted in a frame, the receiver can determine if the message is corrupted
  - Advantage: can be implemented efficiently in hardware using shift registers
- General error detecting properties of C(x) with degree k
  - All single bit errors if $x^k = x^0 = 1$
  - All double bit errors if C(x) has a factor with at least three terms
  - Any odd number of errors if C(x) contains the term (x+1)
  - Any burst error for which the length of the burst is less than k bits (most burst errors of larger length can also be detected)

Internet checksum

- CRC provides strong protection and is used in (almost) all L2 protocols
- Thus, in Internet (L3 and L4) protocols strong protection not necessary
- Simple checksum based method is used in Internet
- Method:
  - A message is viewed as a sequence of 16 bit integers
  - Add all these up using ones complement arithmetic
  - Take ones complement of the result ⇒ checksum
  - Ones complement with 4 bits
    - A negative integer -x is represented by the complement of x
    - {+5 = 0101, -5 = 1010}, {+3=0011, -3 = 1100}
    - Carry bit: -5 + -3 = 1010 + 1100 = 0110 + “carry bit” = 0111

Outline

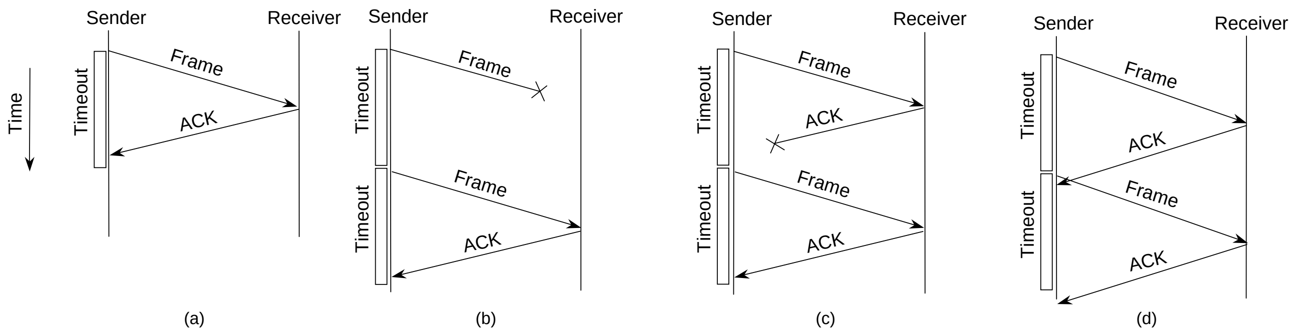
- Host HW and physical link technologies
- Encoding
- Framing
- Error detection/correction
- Reliable transmission
- Multiple access techniques
  - Ethernet
  - Token ring and FDDI
  - Wireless

Reliable transmission

- Packets may be corrupted or simply dropped (due to congestion)
- In packet networks lost packets are retransmitted
- How is this achieved?
  - Acknowledgements (ACKs):
    - short control packet from the receiver
    - acknowledges a successfully received packet
  - Time outs:
    - sender waits for a “reasonable” time for an ACK
    - if an ACK is not received, a time out occurs $\Rightarrow$ packet is retransmitted
- Two ARQ (Automatic Repeat Request) algorithms
  - Stop-and-wait
  - Sliding window

Stop-and-wait

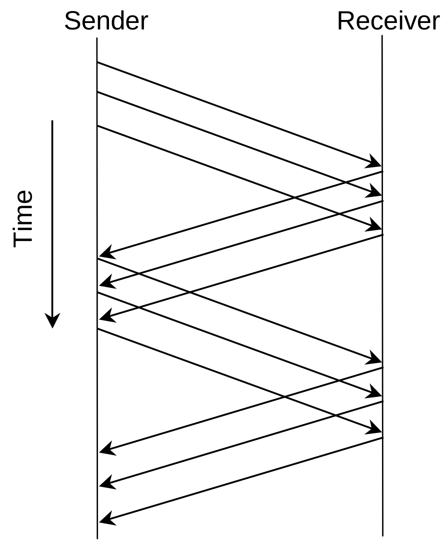
- The sender stops after each packet and waits for an ACK (a)
  - different error situations in b,c,d



- Problem: keeping the pipe full
  - C = 10 Mbps, RTT = 30 ms ⇒ RTT*C = 38 KB
  - sending only one 1500 B packet at a time gives utilization of 1.5/38 ≈ 4 %!
  - simple solution: to send for example 3 frames, use 3 stop-and-wait processes in parallel (concurrent logical channels, used in ARPANET)

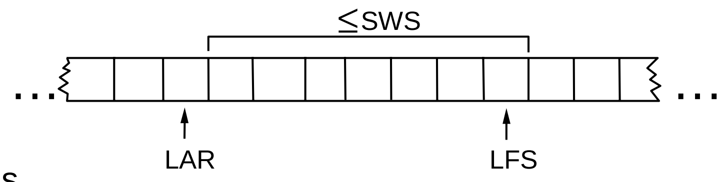
Sliding window

- Algorithm for handling multiple outstanding ACKs
  - the method used in Internet retransmission schemes
- An upper bound is set on the number of outstanding ACKs
  - called a *window*
  - protocol ensures that the number of unacknowledged packets is less than the window size
- In practice, bandwidth-delay product varies over time
  - need to manage the window size dynamically
  - TCP with congestion control



Sliding window - sender side operations

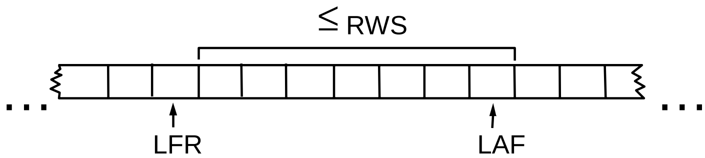
- A sequence number (SeqNum) is assigned to each frame
  - assume for the time being that SeqNum can grow infinitely large
- Three state variables
  - Send Window Size (SWS) (upper bound on nof outstanding ACKs)
  - Last Acknowledgement Received (LAR)
  - Last Frame Sent (LFS)
- Sender maintains: LFS - LAR ≤ SWS



- Sender's actions
  - For each ACK, LAR is incremented and a new packet can be transmitted
  - For each packet, a timeout timer is associated (retransmission if time expires)
  - Thus, sender must buffer upto SWS amount of packets

Sliding window - receiver side operations

- Three state variables
  - Receive Window Size (RWS) (bound on out of order frames)
  - Largest Acceptable Frame (LAF)
  - Last Frame Received (LFR)
- Receiver maintains: LAF - LFR ≤ RWS



- Receiver's actions:
  - If SeqNum ≤ LFR or SeqNum > LAF, the frame is discarded
  - If LFR < SeqNum ≤ LAF, the frame is accepted
  - If SeqNum = LFR + 1, LFR is incremented and frame SeqNum is ACKed
  - If SeqNum > LFR + 1, no ACK is generated until the missing frames arrive (sends cumulative ACKs)

Sliding window operation

- If packets arrive in order and none are lost
  - receiver keeps increasing its LFR and acknowledges each packet
  - sender receives a flow of ACKs and sends a new packet for each ACK
    - pipe stays full
- If packets arrive out of order at the receiver
  - the receiver does not generate ACKs
    - the sender is throttled (cannot send new packets)
  - if missing packets are lost, sender's timeout mechanism takes care of retransmissions
    - sending NAKs not useful
    - how quickly the timeout mechanism detects the missing packet becomes important
  - sending selective ACKs (ACKs for each received packet even if they are out of order) is possible but increases protocol complexity

Finite sequence numbers

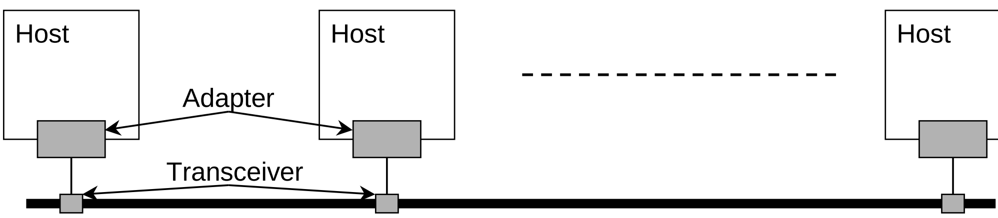
- In practice, SeqNum is a field in the protocol header $\Rightarrow$ SeqNum finite
  - sequence numbers bound to wrap around during operation
- Problem:
  - How big must MaxSeqNum be in order to guarantee that receiver never mistakes a received SeqNum to the previous "round's" same SeqNum?
- Answer: if SWS = RWS, then SWS $\leq$ (MaxSeqNum + 1) / 2
  - SWS must be smaller than half of the nof values of MaxSeqNum
  - Interpretation: assume MaxSeqNum = SWS + 1 and SWS = RWS = 7
    - in the worst case sender sends frames 0, ... , 6
    - assume that all ACKs for the packets are lost
    - sender retransmits again all frames 0, ... , 6
    - receiver is expecting frames 7, 8, 0, 1, ..., but receives 0, ... , 6 interpreting them as belonging to the "next round"!
    - MaxSeqNum must be big enough that all retransmitted frames still "fit" in the same round, i.e., in this case SeqNo range = {0, ..., 13}

Outline

- Host HW and physical link technologies
- Encoding
- Framing
- Error detection/correction
- Reliable transmission
- Multiple access techniques
  - Ethernet
  - Token ring and FDDI
  - Wireless

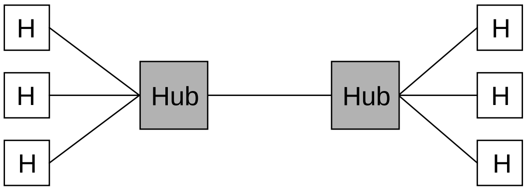
Ethernet overview

- History
  - developed by Xerox in mid 70s, roots in Aloha packet-radio network
  - standardized by Xerox, DEC, and Intel in 1978, (later IEEE 802.3 standard)
- CSMA/CD
  - carrier sense: nodes detect if line is idle or busy
  - multiple access: multiple stations share the bandwidth
  - collision detection: stations listen to their transmission and detect collisions
- Bandwidth: 10Mbps, 100Mbps, 1Gbps
- Ethernet segment (different coaxial cables, max 500 m):
  - transceiver: detects if line is idle, sends the electrical signals
  - adapter: implements the Ethernet MAC protocol (in hardware)



Collision domain

- Using max 4 repeaters at most 5 segments can be connected
  - max 2500 m distance between any two nodes
- Hubs can be used to create a star (hierarchical) topology
  - used in 10BaseT networks with twisted pair cabling
  - 10 = 10 Mbit/s, Base = Baseband system, T=twisted pair (< 100 m)



- Repeaters and hubs are layer 1 devices connecting Ethernet segments
  - data transmitted by any host on that Ethernet is received by all hosts
  - all compete for the same resource
  - all hosts are in the same *collision domain*

Ethernet frame format and addresses

- Frame Format (field lengths in bits)
  - max body length 1500 bytes
  - min body length 46 bytes (long enough to detect a collision)

64	48	48	16		32
Preamble	Dest addr	Src addr	Type	Body	CRC

- Addresses
  - unique, 48-bit unicast address assigned to each adapter
  - example: 8:0:e4:b1:2
  - broadcast: all 1s
  - multicast: first bit is 1
- Receiver functionality simple:
  - adapter forwards to the host all unicast traffic directed to it, all broadcast traffic and the multicast traffic it has subscribed to
- Problem: Distributed algorithm that provides fair access
  - Media Access Protocol (MAC)

Transmit algorithm (1)

- No centralized access control
  - collisions occur and they are detected
- If line is idle…
  - send immediately
  - upper bound message size of 1500 bytes
  - must wait 9.6 μs between back-to-back frames
- If line is busy…
  - wait until idle and transmit immediately
  - called 1-persistent (special case of p-persistent)
    - p-persistent: if line is idle, transmit with probability p
    - idea: many hosts may be waiting for the line to become idle and we do not want them all to start transmitting (minimizes prob. of collisions)

Transmit algorithm (2)

- If collision…
  - jam for 32 bits, then stop transmitting frame
  - minimum frame is 64 bytes (header + 46 bytes of data)
    - long enough to fill a 2500 m Ethernet operating at 10 Mbps
  - delay and try again
    - 1st time: 0 or 51.2us
    - 2nd time: 0, 51.2, or 102.4us
    - 3rd time51.2, 102.4, or 153.6us
    - nth time: k x 51.2us, for randomly selected k=0..2n - 1
    - give up after several tries (usually 16)
    - exponential backoff

Ethernets in practice

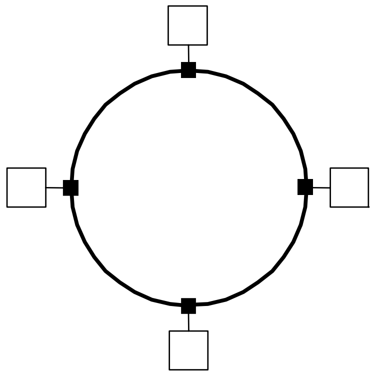
- Performance
  - Ethernet works efficiently in light load
  - 30% load is considered a heavy load and too much of Ethernet's capacity is wasted on collisions
  - no flow control in Ethernet (flow control implemented in IP protocols)
- Nof hosts
  - theoretical maximum 1024 hosts
  - in reality most have < 200 hosts
- Length
  - theoretical maximum 2500 m with round-trip delay 51.2 μs
  - in practice, delay is closer to 5 μs
- Ethernet advantages:
  - easy to manage and administer (add/remove hosts, no route configuration)
  - cheap

Outline

- Host HW and physical link technologies
- Encoding
- Framing
- Error detection/correction
- Reliable transmission
- Multiple access techniques
  - Ethernet
  - Token ring and FDDI
  - Wireless

Token ring overview

- Examples
  - 16Mbps IEEE 802.5 (based on earlier IBM ring)
  - 100Mbps Fiber Distributed Data Interface (FDDI)
- Similarities with Ethernet
  - shared medium with a distributed access algorithm
  - all nodes see all frames
- Differences to Ethernet
  - ring topology
  - access to the ring is tightly controlled (tokens)
    - NOT random access



Token ring operation

- Idea
  - frames flow in one direction: upstream to downstream
  - special bit pattern (token, length 24 bits) rotates around ring
  - if a node has frame(s) to transmit and sees the token
    - node inserts its frame into the ring
    - each node forwards the frame, receiver copies it
    - node can transmit for upto THT (Token Holding Time, default 10 ms)
  - release token after done transmitting
    - immediate release (token is inserted before last frame has been sent)
    - delayed release (token is inserted after last frame has been sent)
  - remove your frame when it comes back around
  - stations get round-robin service
- Additional features
  - supports unicast, broadcast and multicast addresses
  - reliable frame delivery: receiver sets A and C bits if frame OK
  - supports priorities

Token ring maintenance

- Designated monitor station guards operation of the ring
  - any station can be the monitor
  - healthy monitor issues regular control messages
  - if control msgs are not received for some period of time
    - any station can try to become new monitor by issuing “claim token” msg
    - new monitor is elected based on “highest address” rule
- Monitor functions
  - adds extra bits of delay if necessary (ring must be at least 24 “bits long”)
  - makes sure that token is not lost (host crash, bit errors, ...)
    - maximum rotation time timer (NumStations x THT + RingLatency)
  - check for corrupted or orphaned frames
  - detection of dead stations

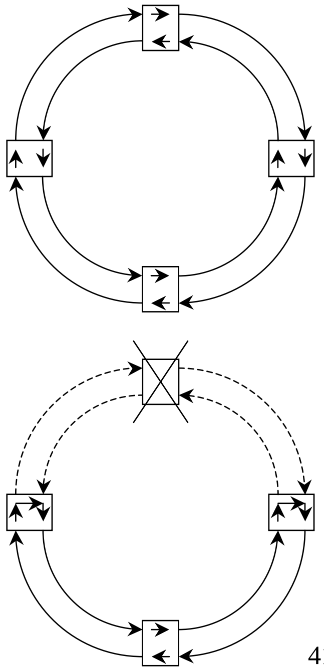
Physical properties and frame format of token ring

- Properties:
  - Robustness:
    - in a ring, if any station fails the ring is inoperable
    - solution: an electromechanical relay in the network adapter is open as long as the station is operating
  - data rate: 4 Mbps (old version) or 16 Mbps
  - bit encoding: Manchester
  - upto 260 stations (250 in IEEE 802.5)
  - physical medium: twisted pair
- Frame Format

8	8	8	48	48		32	8	24
Start of frame	Access control	Frame control	Dest addr	Src addr	Body	CRC	End of frame	Status

  - 48 bit addresses (as in Ethernet)
  - frame status contains the A and C bits

FDDI overview



- Dual ring
  - second ring not used in normal operation
  - if a node fails or a single link fails, ring loops back to form a complete ring
- Dual ring expensive
  - SAS (single attachment station) possible
  - multiple SAS connected to a DAS (dual attachment station) (=concentrator)
  - in case of SAS failure, DAS bypasses the faulty SAS
- Physical properties
  - at most 500 stations
  - max cable length 200 km (100 km ring length)
  - physical media: coax, twisted pair, fiber
  - encoding: 4B/5B
- Frame format (almost) the same as in Token ring

Timed Token Algorithm (1)

- Algorithm for controlling token holding time: Timed Token Algorithm
  - ensures that every station gets to transmit within a defined period of time
- Token Holding Time (THT)
  - upper limit on how long a station can hold the token, configured value
  - same as in Token ring
- Token Rotation Time (TRT)
  - how long it takes the token to traverse the ring
  - TRT <= ActiveNodes x THT + RingLatency
- Target Token Rotation Time (TTRT)
  - agreed-upon upper bound on TRT
  - decided during token generation (ring initialization)

Timed Token Algorithm (2)

- Each node measures TRT between successive tokens
  - if measured-TRT > TTRT: token is late so don't send
  - if measured-TRT < TTRT: token is early so OK to send
- Problem:
  - an upstream node with lot of data to send hogs all bw before the token reaches downstream nodes which might have delay critical data
- Two classes of traffic
  - synchronous: can always send
  - asynchronous: can send only if token is early
- Problem: synchronous traffic can take all bw
- Solution: max TTRT amount of synch data can be sent during token round
  - nodes first send TTRT worth of asynchronous data and then other nodes send TTRT worth of synchronous data
  - worst case: measured-TRT can be 2xTTRT between seeing token
  - in the next token round, token is already late and asynchronous data cannot be sent ⇒ back-to-back 2xTTRT rotations not possible

Token maintenance

- Lost Token
  - no token when initializing ring
  - bit error corrupts token pattern
  - node holding token crashes
  - token loss monitored by all stations
    - stations should see valid frames or tokens every now and then
    - if nothing is seen for 2.5 ms, node issues a "claim" msg
- Generating a Token (and agreeing on TTRT)
  - execute when joining ring or suspect a failure
  - send a claim frame that includes the node's TTRT bid
  - when receive claim frame, update the bid and forward
  - if your claim frame makes it all the way around the ring:
    - your bid was the lowest
    - everyone knows TTRT
    - you insert new token

Outline

- Host HW and physical link technologies
- Encoding
- Framing
- Error detection/correction
- Reliable transmission
- Multiple access techniques
  - Ethernet
  - Token ring and FDDI
  - Wireless

Wireless LANs

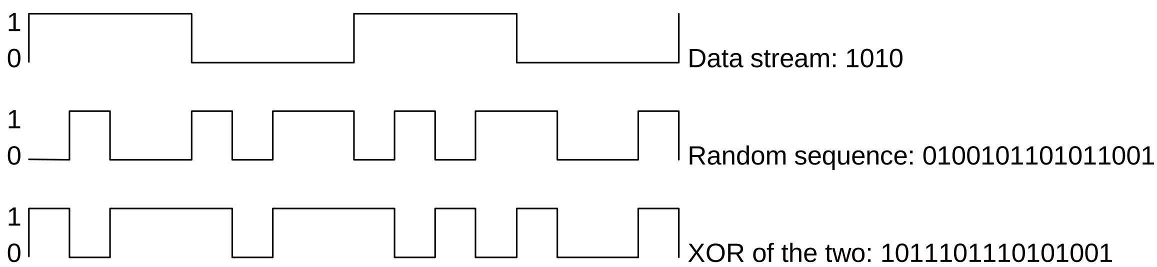
- (Original) Wireless LAN standard: IEEE 802.11
  - limited geographical coverage
  - defines MAC protocol suitable for wireless environment
  - additional features: real time support, power mgmt, security
- Physical properties
  - bandwidth: 1 or 2 Mbps
  - physical media
    - 2 media based on spread spectrum radio operating in 2.4GHz frequency range
    - diffused infrared (sender and receiver do not need to have line of sight contact), distance limitation approx. 10 m
- New standards
  - IEEE 802.11a and IEEE 802.11b
  - Higher data rates: 10 Mbit/s upto 54 Mbit/s
  - New frequency range: 5 GHz

Spread spectrum techniques (1)

- General principles
  - signal spread over wider frequency band than required
  - minimizes impact of interference from other devices
  - originally military technology, deigned to thwart jamming
  - transmission “coded” such that the signal appears as noise to an observer not knowing the “key”
    - possible to trade off capacity and amount of noise
- Frequency hopping
  - signal transmitted over random sequence of frequencies
  - sender and receiver share…
    - pseudorandom number generator
    - seed
  - ⇒ receiver can hop frequencies in sync
  - 802.11 uses 79 x 1MHz-wide frequency bands

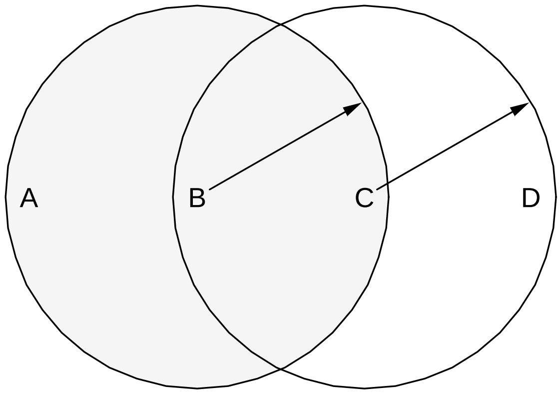
Spread spectrum techniques (2)

- Direct sequence
  - for each bit, send XOR of that bit and n random bits
  - random sequence known to both sender and receiver
  - called n-bit chipping code
  - 802.11 defines an 11-bit chipping code



MAC for wireless

- Idea to provide similar random access as in Ethernet, but ...
  - in wireless environment not all nodes are always within reach of each other
- Problem 1: hidden nodes
  - Assume node A and C want to transmit to B
  - A and C are unaware of each other
  - transmissions collide at B, but A and C do not know about that
- Problem 2: exposed nodes
  - suppose B is sending to A
  - C hears this
  - however, C can still transmit to D



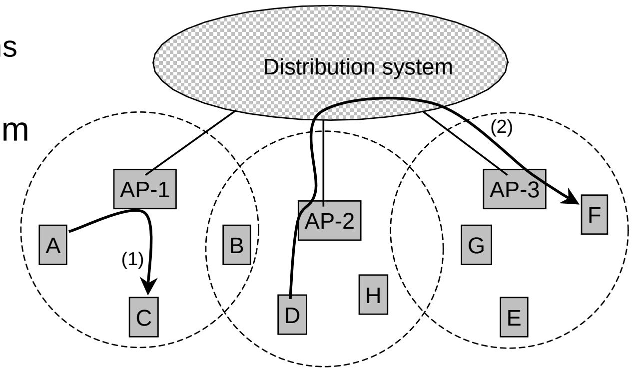
- Wireless MAC addresses the problems by collision avoidance strategy

MACAW

- MACAW (MACA for Wireless LANs)
  - MACA = Multiple Access with Collision Avoidance
  - idea: nodes ask for permission to send
- MACAW operation:
  - sender transmits RequestToSend (RTS) frame
  - receiver replies with ClearToSend (CTS) frame
  - neighbors…
    - that see CTS: keep quiet (they are too close to sender)
    - that see RTS but not CTS: ok to transmit
  - receiver sends ACK when it has received the frame
    - neighbors silent until see ACK
  - Collisions (= multiple RTS frames sent at the same time)
    - no collision detection
    - known when senders do not receive CTS
    - exponential backoff

Supporting mobility: Access Points (AP)

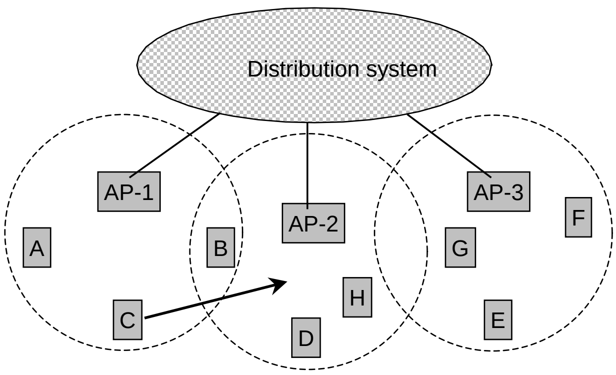
- Each AP serves hosts within a cell
  - cf. base stations in cellular systems
- APs connected to distribution system
  - 802.11 does not specify what (can be e.g. Ethernet)



- Each mobile node associates with an AP
  - hierarchical network
  - process of making associations called scanning
- Routing
  - within a cell transmissions through AP (1)
  - transmitting to a node in neighboring AP done via distribution network (2)

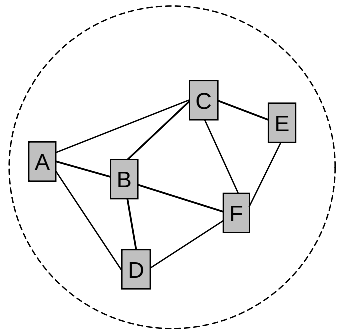
Associating with an Access Point

- Active scanning
  - node C sends Probe frame
  - all APs within reach reply with ProbeResponse
  - at some point node C selects AP-2 and sends a new AssociationRequest
  - AP-2 replies with AssociationResponse and notifies AP-1 that host C has moved



- Passive scanning
  - APs periodically send Beacon frames
  - host can decide to join at will by replying with AssociationRequest

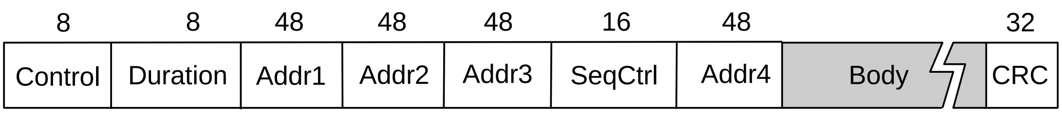
Supporting mobility: ad hoc networks



- Ad hoc network
  - IEEE 802.11 stations can dynamically form a network without APs
  - each host acts as a “switch” that relays packets based on information about neighboring host location
  - mesh type network topology
  - ad hoc routing a very active research field
- Applications
  - laptop meetings in a conference
  - interconnection of personal devices (e.g. in a house)
  - battlefield

- IETF MANET (Mobile Ad hoc Networks) working group

802.11 frame format



- Control field
  - indicates if frame is a data frame; an RTS or CTS frame; or is used by the scanning algorithm
  - ToDS and FromDS bits (used with 4 address fields)
- 4 address fields
  - if sender and receiver in same cell
    - ToDS = FromDS = 0
    - Addr1 = target, Addr2 = source
  - if sender and receiver in different cells
    - ToDS = FromDS = 1
    - Addr1 = target, Addr4 = source
    - Addr2 = AP that sent frame to target
    - Addr3 = AP that received frame from source