

15-441 Computer Networking Lecture 7 - Ethernet

Peter Steenkiste
Departments of Computer Science and
Electrical and Computer Engineering

15-441 Networking, Spring 2008
<http://www.cs.cmu.edu/~dga/15-441/S08>

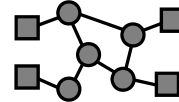
1

Problem: Sharing a Wire

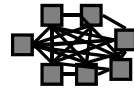


Learned how to connect hosts

- ... But what if we want more hosts?



Switches



Wires for everybody!

- Expensive! How can we share a wire?



2

Listen and Talk



- Natural scheme – listen before you talk...
 - » Works well in practice

3

Listen and Talk



- Natural scheme – listen before you talk...
 - » Works well in practice

4

Listen and Talk



- Natural scheme – listen before you talk...
 - » Works well in practice
- But sometimes this breaks down
 - » Why? How do we fix/prevent this?

5

Problem: Who is this packet for?



- Need to put an address on the packet
- What should it look like?
- How do you determine your own address?
- How do you know what address you want to send it to?

6

Outline

- Aloha
- Ethernet MAC
- Collisions
- Ethernet Frames

7

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 and ALOHA
 - » CSMA and CSMA/CD

8

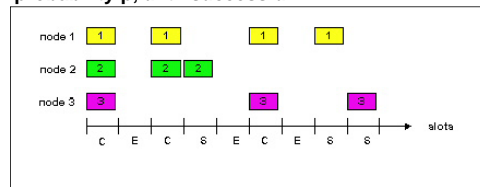
Aloha - Basic Technique

- First random MAC developed
 - » For radio-based communication in Hawaii (1970)
- Basic idea:
 - » When you are ready, transmit
 - » Receivers send ACK for data
 - » Detect collisions by timing out for ACK
 - » Recover from collision by trying after random delay
 - Too short → large number of collisions
 - Too long → underutilization

9

Slotted Aloha

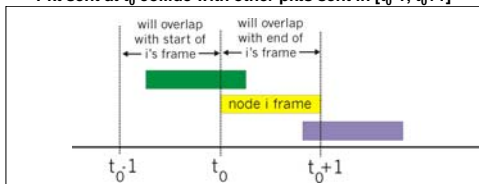
- Time is divided into equal size slots
 - » Equal to packet transmission time
- Node (w/ packet) transmits at beginning of next slot
- If collision: retransmit pkt in future slots with probability p , until successful



10

Pure (Unslotted) ALOHA

- Unslotted Aloha: simpler, no synchronization
- Pkt needs transmission:
 - » Send without awaiting for beginning of slot
- Collision probability increases:
 - » Pkt sent at t_0 collide with other pkts sent in $[t_0-1, t_0+1]$



11

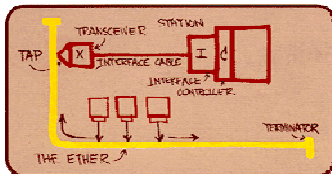
Outline

- Aloha
- Ethernet MAC
- Collisions
- Ethernet Frames

12

Ethernet

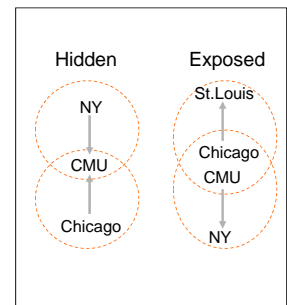
- First practical local area network, built at Xerox PARC in 70's
- "Dominant" LAN technology:
 - » Cheap
 - » Kept up with speed race: 10, 100, 1000 Mbps



13

Ethernet MAC - Carrier Sense

- Basic idea:
 - » Listen to wire before transmission
 - » Avoid collision with active transmission
- Why didn't ALOHA have this?
 - » In wireless, relevant contention at the **receiver**, not sender
 - Hidden terminal
 - Exposed terminal



14

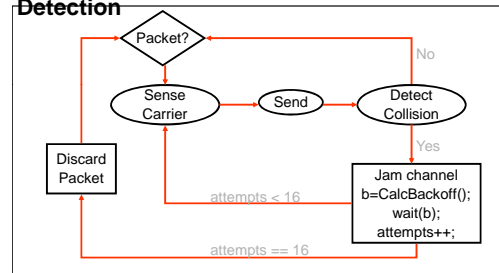
Ethernet MAC - Collision Detection

- But: ALOHA has collision detection also?
 - » That was very slow and inefficient
- Basic idea:
 - » Listen while transmitting
 - » If you notice interference → assume collision
- Why didn't ALOHA have this?
 - » Very difficult for radios to listen and transmit
 - » Signal strength is reduced by distance for radio
 - Much easier to hear "local, powerful" radio station than one in NY
 - You may not notice any "interference"

15

Ethernet MAC (CSMA/CD)

- Carrier Sense Multiple Access/Collision Detection



16

Ethernet CSMA/CD: Making it work

- **Jam Signal:** make sure all other transmitters are aware of collision; 48 bits;
- **Exponential Backoff:**
 - If deterministic delay after collision, collision will occur again in lockstep
 - Why not random delay with fixed mean?
 - » Few senders → needless waiting
 - » Too many senders → too many collisions
 - **Goal:** adapt retransmission attempts to estimated current load
 - » heavy load: random wait will be longer

17

Ethernet Backoff Calculation

- Exponentially increasing random delay
 - » Infer senders from # of collisions
 - » More senders → increase wait time
- First collision: choose K from {0,1}; delay is K x 512 bit transmission times
- After second collision: choose K from {0,1,2,3}...
- After ten or more collisions, choose K from {0,1,2,3,4,...,1023}

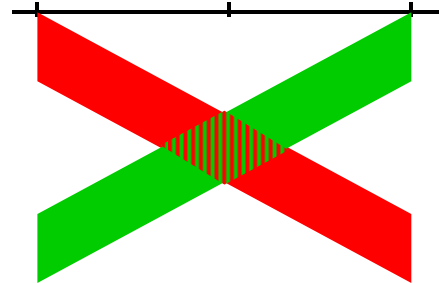
18

Outline

- Aloha
- Ethernet MAC
- **Collisions**
- Ethernet Frames

19

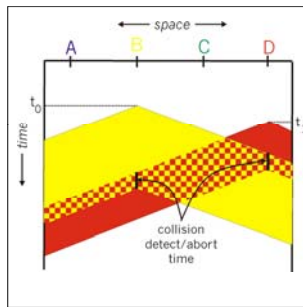
Collisions



20

Minimum Packet Size

- What if two people sent really small packets
 - » How do you find collision?



21

Ethernet Collision Detect

- Min packet length > 2x max prop delay
 - » If A, B are at opposite sides of link, and B starts one link prop delay after A
- Jam network for 32-48 bits after collision, then stop sending
 - » Ensures that everyone notices collision

22

End to End Delay

- c in cable = 60% * c in vacuum = 1.8×10^8 m/s
- Modern 10Mb Ethernet {
 - » 2.5km, 10Mbps
 - » $\approx 12.5\mu s$ delay
 - » +Introduced repeaters (max 5 segments)
 - » Worst case – 51.2us round trip time!
- Slot time = 51.2us = 512bits in flight
 - » After this amount, sender is guaranteed sole access to link
 - » 51.2us = slot time for backoff

23

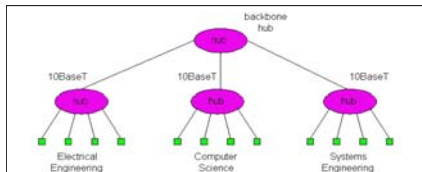
Packet Size

- What about scaling? 3Mbit, 100Mbit, 1Gbit...
 - » Original 3Mbit Ethernet did not have minimum packet size → bonus question!
 - Max length = 1Km and No repeaters
 - » For higher speeds must make network smaller, minimum packet size larger or both
- What about a maximum packet size?
 - » Needed to prevent node from hogging the network
 - » 1500 bytes in Ethernet

24

10BaseT and 100BaseT

- 10/100 Mbps rate; latter called “fast ethernet”
- T stands for Twisted Pair (wiring)
- Minimum packet size requirement
 - » Make network smaller → solution for 100BaseT



25

Gbit Ethernet

- Minimum packet size requirement
 - » Make network smaller?
 - 512bits @ 1Gbps = 512ns
 - $512\text{ns} * 1.8 * 10^8 = 92\text{meters} = \text{too small !!}$
 - » Make min pkt size larger!
 - Gigabit Ethernet uses collision extension for small pkts and backward compatibility
- Maximum packet size requirement
 - » 1500 bytes is not really “hogging” the network
 - » Defines “jumbo frames” (9000 bytes) for higher efficiency

26

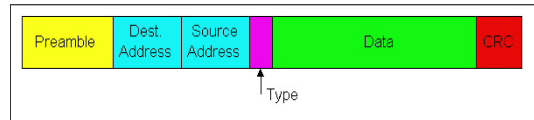
Outline

- Aloha
- Ethernet MAC
- Collisions
- Ethernet Frames

27

Ethernet Frame Structure

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



28

Ethernet Frame Structure (cont.)

- Preamble: 8 bytes
 - » 101010...1011
 - » Used to synchronize receiver, sender clock rates
- CRC: 4 bytes
 - » Checked at receiver, if error is detected, the frame is simply dropped

29

Ethernet Frame Structure (cont.)

- Each protocol layer needs to provide some hooks to upper layer protocols
 - » Demultiplexing: identify which upper layer protocol packet belongs to
 - » E.g., port numbers allow TCP/UDP to identify target application
 - » Ethernet uses Type field
- Type: 2 bytes
 - » Indicates the higher layer protocol, mostly IP but others may be supported such as Novell IPX and AppleTalk

30

Addressing Alternatives

- **Broadcast** → all nodes receive all packets
 - » Addressing determines which packets are kept and which are packets are thrown away
 - » Packets can be sent to:
 - Unicast – one destination
 - Multicast – group of nodes (e.g. “everyone playing Quake”)
 - Broadcast – everybody on wire
- **Dynamic addresses (e.g. Appletalk)**
 - » Pick an address at random
 - » Broadcast “is anyone using address XX?”
 - » If yes, repeat
- **Static address (e.g. Ethernet)**

31

Ethernet Frame Structure (cont.)

- **Addresses: 6 bytes**
 - » Each adapter is given a globally unique address at manufacturing time
 - Address space is allocated to manufacturers
 - 24 bits identify manufacturer
 - E.g., 0:0:15:* → 3com adapter
 - Frame is received by all adapters on a LAN and dropped if address does not match
 - » **Special addresses**
 - Broadcast – FF:FF:FF:FF:FF:FF is “everybody”
 - Range of addresses allocated to multicast
 - Adapter maintains list of multicast groups node is interested in

32

Why Did Ethernet Win?

- **Failure modes**
 - » Token rings – network unusable
 - » Ethernet – node detached
- **Good performance in common case**
 - » Deals well with bursty traffic
 - » Usually used at low load
- **Volume** → lower cost → higher volume
- **Adaptable**
 - » To higher bandwidths (vs. FDDI)
 - » To switching (vs. ATM)
- **Easy incremental deployment**
- **Cheap cabling, etc**

33

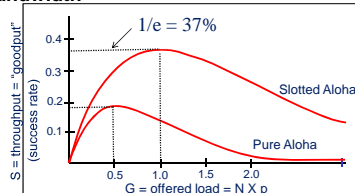
And .. It is Easy to Manage

- **You plug in the host and it basically works**
 - » No configuration at the datalink layer
 - » Today: may need to deal with security
- **Protocol is fully distributed**
- **Broadcast-based.**
 - » In part explains the easy management
 - » Some of the LAN protocols (e.g. ARP) rely on broadcast
 - Networking would be harder without ARP
 - » Not having natural broadcast capabilities adds complexity to a LAN
 - Example: ATM

34

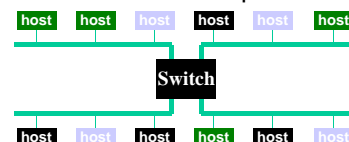
Ethernet Problems: Unstable at High Load

- **Peak throughput worst with**
 - » More hosts – more collisions to identify single sender
 - » Smaller packet sizes – more frequent arbitration
 - » Longer links – collisions take longer to observe, more wasted bandwidth
- **But works well**
 - » Can improve efficiency by avoiding above conditions



Virtual LANs

- **Single physical LAN infrastructure that carries multiple “virtual” LANs simultaneously.**
- **Each virtual LAN has a LAN identifier in the packet.**
 - » Switch keeps track of what nodes are on each segment and what their virtual LAN id is
- **Can bridge and route appropriately.**
- **Broadcast packets stay within the virtual LAN.**
 - » Limits the collision domain for the packet



36

Summary

- CSMA/CD → carrier sense multiple access with collision detection
 - » Why do we need exponential backoff?
 - » Why does collision happen?
 - » Why do we need a minimum packet size?
 - How does this scale with speed?
- Ethernet
 - » What is the purpose of different header fields?
 - » What do Ethernet addresses look like?
- What are some alternatives to Ethernet design?

37

EXTRA SLIDES

38

Outline

- Random Access Analysis

39

Slotted Aloha Efficiency

- Q:** What is max fraction slots successful?
- A:** Suppose N stations have packets to send
- » Each transmits in slot with probability p
 - » Prob. successful transmission S is:
- by single node: $S = p(1-p)^{N-1}$
- by any of N nodes
- $$S = \text{Prob (only one transmits)}$$
- $$= N p (1-p)^{N-1}$$
- ... choosing optimum p as $N \rightarrow \infty \rightarrow$
- $$\dots p = 1/N$$
- $$= 1/e = .37 \text{ as } N \rightarrow \infty$$

At best:

40

Pure Aloha (cont.)

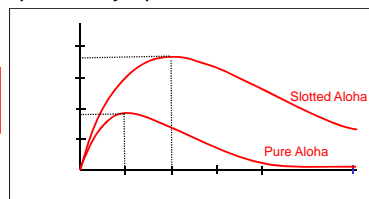
$P(\text{success by given node}) = P(\text{node transmits}) \times P(\text{no other node transmits in } [p_0-1, p_0]) \times P(\text{no other node transmits in } [p_0, p_0+1])$

$$= p \times (1-p)^{N-1} \times (1-p)^{N-1}$$

$P(\text{success by any of } N \text{ nodes}) = N p \times (1-p)^{N-1} \times (1-p)^{N-1} = 1/(2e) = .18$

... choosing optimum p as $N \rightarrow \infty \rightarrow p = 1/2N \dots$

protocol



41

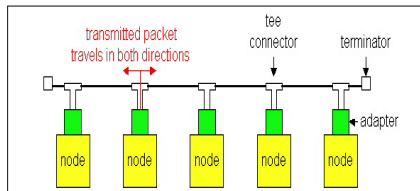
Simple Analysis of Efficiency

- Key assumptions
 - » All packets are same, small size
 - Packet size = size of contention slot
 - » All nodes always have pkt to send
 - » p is chosen carefully to be related to N
 - p is actually chosen by exponential backoff
 - » Takes full slot to detect collision (i.e. no “fast collision detection”)

42

Ethernet Technologies: 10Base2

- **10:** 10Mbps; **2:** under 185 (~200) meters cable length
- Thin coaxial cable in a bus topology



- Repeaters used to connect up to multiple segments
- Repeater repeats bits it hears on one interface to its₄₃

Gbit Ethernet

- Use 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 to be efficient
- Uses hubs, called here "Buffered Distributors"
- Full-Duplex at 1 Gbps for point-to-point links

44

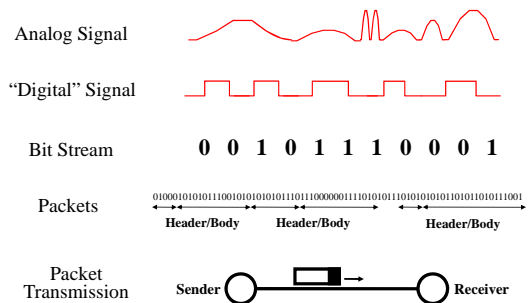
Lecture 6 Datalink - Framing, Switching

Peter Steenkiste
Departments of Computer Science and
Electrical and Computer Engineering
Carnegie Mellon University

15-441 Networking, Spring 2008
<http://www.cs.cmu.edu/~dga/15-441/S08>

45

From Signals to Packets



46

Datalink Functions

- **Framing:** encapsulating a network layer datagram into a bit stream.
 - » Add header, mark and detect frame boundaries, ...
- **Media access:** controlling which frame should be sent over the link next.
 - » Easy for point-to-point links; half versus full duplex
 - » Harder for multi-access links: who gets to send?
- **Error control:** error detection and correction to deal with bit errors.
 - » May also include other reliability support, e.g. retransmission
- **Flow control:** avoid that the sender outruns the receiver.

47

Datalink Lectures

- Framing and error coding.
- Datalink architectures.
- Switch-based networks.
 - » Packet forwarding
 - » Flow and error control
- Taking turn protocols.
- Contention-based networks: basic Ethernet.
- Ethernet bridging and switching.
- Connectivity to the home.
- Circuit-based communication

48

Framing

- A link layer function, defining which bits have which function.
- Minimal functionality: mark the beginning and end of packets (or frames).
- Some techniques:
 - › out of band delimiters (e.g. FDDI 4B/5B control symbols)
 - › frame delimiter characters with character stuffing
 - › frame delimiter codes with bit stuffing
 - › synchronous transmission (e.g. SONET)

49

Character and Bit Stuffing

- Mark frames with special character.
 - › What happens when the user sends this character?
 - › Use escape character when controls appear in data:
"abc"def -> "abc\"def
 - › Very common on serial lines, in editors, etc.
- Mark frames with special bit sequence
 - › must ensure data containing this sequence can be transmitted
 - › example: suppose 11111111 is a special sequence.
 - › transmitter inserts a 0 when this appears in the data:
11111111 -> 111111101
 - › must stuff a zero any time seven 1s appear:
11111110 -> 111111100
 - › receiver unstuffs.

50

Example: Ethernet Framing



- Preamble is 7 bytes of 10101010 (5 MHz square wave) followed by one byte of 10101011
- Allows receivers to recognize start of transmission after idle channel

51

SONET

- SONET is the Synchronous Optical Network standard for data transport over optical fiber.
- One of the design goals was to be backwards compatible with many older telco standards.
- Beside minimal framing functionality, it provides many other functions:
 - › operation, administration and maintenance (OAM) communications
 - › synchronization
 - › multiplexing of lower rate signals
 - › multiplexing for higher rates

52

Standardization History

- Process was started by divestiture in 1984.
 - › Multiple telephone companies building their own infrastructure
- SONET concepts originally developed by Bellcore.
- First standardized by ANSI T1X1 group for the US.
- Later picked up by CCITT and developed its own version.
- SONET/SDH standards approved in 1988.

53

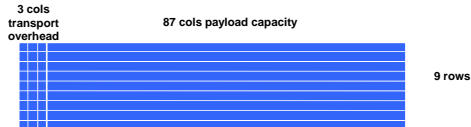
A Word about Data Rates

- Bandwidth of telephone channel is under 4KHz, so when digitizing:
 $8000 \text{ samples/sec} \times 8 \text{ bits} = 64 \text{ Kbits/second}$
- Common data rates supported by telcos in North America:
 - › Modem: rate improved over the years
 - › T1/DS1: 24 voice channels plus 1 bit per sample
 $(24 \times 8 + 1) \times 8000 = 1.544 \text{ Mbits/second}$
 - › T3/DS3: 28 T1 channels:
 $7 \times 4 \times 1.544 = 44.736 \text{ Mbits/second}$

54

Synchronous Data Transfer

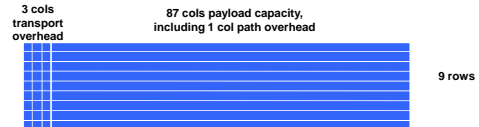
- Sender and receiver are always synchronized.
 - Frame boundaries are recognized based on the clock
 - No need to continuously look for special bit sequences
- SONET frames contain room for control and data.
 - Data frame multiplexes bytes from many users
 - Control provides information on data, management, ...



55

SONET Framing

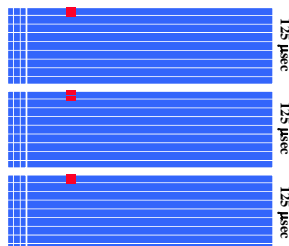
- Base channel is STS-1 (Synchronous Transport System).
 - Takes 125 μ sec and corresponds to 51.84 Mbps
 - 1 byte/frame corresponds to a 64 Kbs channel (voice)
 - Transmitted on an OC-1 optical carrier (fiber link)
- Standard ways of supporting slower and faster channels.
 - Support both old standards and future (higher) data rates
- Actual payload frame "floats" in the synchronous frame.
 - Clocks on individual links do not have to be synchronized



56

How Do We Support Lower Rates?

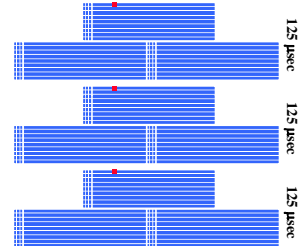
- 1 Byte in every consecutive frame corresponds to a 64 Kbit/second channel.
 - 1 voice call.
- Higher bandwidth channels hold more bytes per frame.
 - Multiples of 64 Kbit/second
- Channels have a "telecom" flavor.
 - Fixed bandwidth
 - Just data – no headers
 - SONET multiplexers remember how bytes on one link should be mapped to bytes on the next link
 - Byte 33 on incoming link 1 is byte 97 on outgoing link 7



57

How Do We Support Higher Rates?

- Send multiple frames in a 125 μ sec time slot.
- The properties of a channel using a single byte/ST-1 frame are maintained!
 - Constant 64 Kbit/second rate
 - Nice spacing of the byte samples
- Rates typically go up by a factor of 4.
- Two ways of doing interleaving.
 - Frame interleaving
 - Column interleaving
 - concatenated version, i.e. OC-3c



58

The SONET Signal Hierarchy

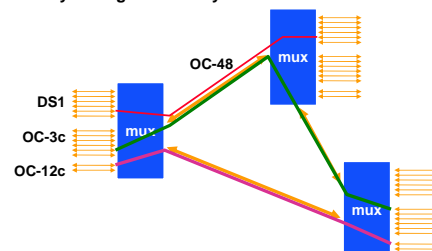
STS-1 carries one DS-3 plus overhead

Signal Type	line rate	# of DS0
DS0 (POTS)	64 Kbs	1
DS1	1.544 Mbs	24
DS3	44.736 Mbs	672
OC-1	51.84 Mbs	672
OC-3	155 Mbs	2,016
OC-12	622 Mbs	8,064
STS-48	2.49 Gbs	32,256
STS-192	9.95 Gbs	129,024
STS-768	39.8 Gbs	516,096

59

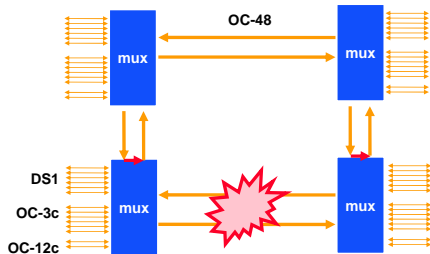
Using SONET in Networks

Add-drop capability allows soft configuration of networks, usually managed manually.



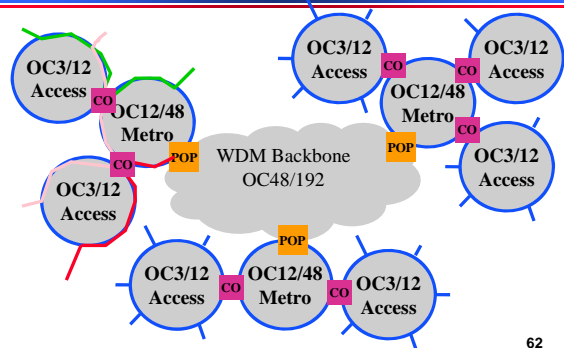
60

Self-Healing SONET Rings



61

SONET as Physical Layer



62

Error Coding

- **Transmission process may introduce errors into a message.**
 - › Single bit errors versus burst errors
- **Detection:**
 - › Requires a convention that some messages are invalid
 - › Hence requires extra bits
 - › An (n,k) code has codewords of n bits with k data bits and $r = (n-k)$ redundant check bits
- **Correction**
 - › Forward error correction: many related code words map to the same data word
 - › Detect errors and retry transmission

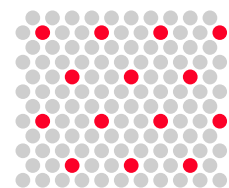
63

Basic Concept: Hamming Distance

- **Hamming distance** of two bit strings = number of bit positions in which they differ.
- If the valid words of a code have minimum Hamming distance D , then $D-1$ bit errors can be detected.
- If the valid words of a code have minimum Hamming distance D , then $\lfloor (D-1)/2 \rfloor$ bit errors can be corrected.

1 0 1 1 0
1 1 0 1 0 HD=2

HD=3



64

Examples

- A $(4,3)$ parity code has $D=2$:
0001 0010 0100 0111 1000 1011 1101 1110
(last bit is binary sum of previous 3, inverted - "odd parity")
- A $(7,4)$ code with $D=3$ (2ED, 1EC):
0000000 0001101 0010111 0011010
0100011 0101110 0110100 0111001
1000110 1001011 1010001 1011100
1100101 1101000 1110010 1111111
- 1001111 corrects to 1001011
- Note the inherent risk in correction; consider a 2-bit error resulting in 1001011 \rightarrow 1111011.
- There are formulas to calculate the number of extra bits that are needed for a certain D .

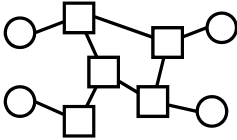
65

Cyclic Redundancy Codes (CRC)

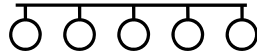
- Commonly used codes that have good error detection properties.
 - › Can catch many error combinations with a small number of redundant bits
- Based on division of polynomials.
 - › Errors can be viewed as adding terms to the polynomial
 - › Should be unlikely that the division will still work
- Can be implemented very efficiently in hardware.
- Examples:
 - › CRC-32: Ethernet
 - › CRC-8, CRC-10, CRC-32: ATM

66

Datalink Architectures



- Packet forwarding.
- Error and flow control.



- Media access control.
- Scalability.

67

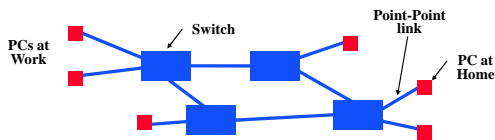
Media Access Control

- How do we transfer packets between two hosts connected to the same network?
- Switches connected by point-to-point links – store-and-forward.
 - › Used in WAN, LAN, and for home connections
 - › Conceptually similar to “routing”
 - But at the datalink layer instead of the network layer
 - › Today
- Multiple access networks – contention based.
 - › Multiple hosts are sharing the same transmission medium
 - › Used in LANs and wireless
 - › Need to control access to the medium
 - › Mostly Thursday lecture

68

A Switch-based Network

- Switches are connected by point-point links.
- Packets are forwarded hop-by-hop by the switches towards the destination.
 - › Forwarding is based on the address
- How does a switch work?
- How do nodes exchange packets over a link?
- How is the destination addressed?



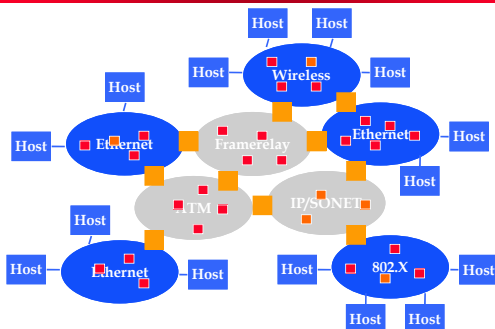
69

Switching Introduction

- Idea: forward units of data based on address in header.
- Many data-link technologies use switching.
 - › Virtual circuits: Frame Relay, ATM, X.25, ..
 - › Packets: Ethernet, MPLS, ...
- “Switching” also happens at the network layer.
 - › Layer 3: Internet protocol
 - › In this case, address is an IP address
 - › IP over SONET, IP over ATM, ..
 - › Otherwise, operation is very similar
- Switching is different from SONET mux/demux.
 - › SONET channels statically configured - no addresses

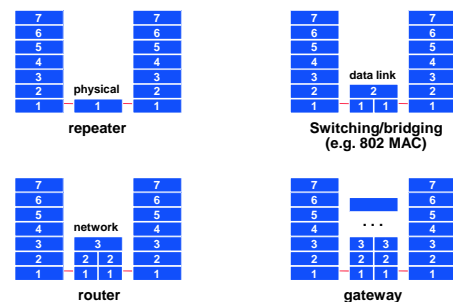
70

An Inter-network



71

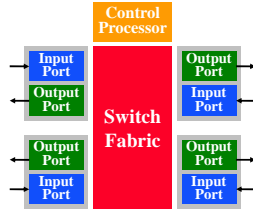
Internetworking Options



72

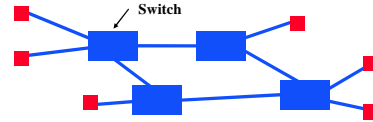
Switch Architecture

- Takes in packets in one interface and has to forward them to an output interface based on the address.
 - › A big intersection
 - › Same idea for bridges, switches, routers: address look up differs
- Control processor manages the switch and executes higher level protocols.
 - › E.g. routing, management, ..
- The switch fabric directs the traffic to the right output port.
- The input and output ports deal with transmission and reception of packets.



73

Packet Forwarding: Address Lookup



Address	Next Hop	Info
131.12.81.2508	3	13
89.136.30.2137	3	-
43.182.36.76998	0	-
128.2.15.3	1	(2,34)

- Address from header.
 - › Absolute address (e.g. Ethernet)
 - › (IP address for routers)
 - › (VC identifier, e.g. ATM)
- Next hop: output port for packet.
- Info: priority, VC id, ..
- Table is filled in by routing protocol.

74

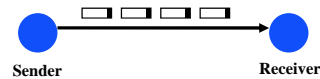
Link Flow Control and Error Control

- Naïve protocol.
- Dealing with receiver overflow: flow control.
- Dealing with packet loss and corruption: error control.
- Meta-comment: these issues are relevant at many layers.
 - › Link layer: sender and receiver attached to the same "wire"
 - › End-to-end: transmission control protocol (TCP) - sender and receiver are the end points of a connection
- How can we implement flow control?
 - › "You may send" (windows, stop-and-wait, etc.)
 - › "Please shut up" (source quench, 802.3x pause frames, etc.)
 - › Where are each of these appropriate?

75

A Naïve Protocol

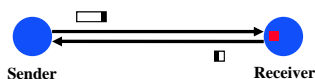
- Sender simply sends to the receiver whenever it has packets.
- Potential problem: sender can outrun the receiver.
 - › Receiver too slow, buffer overflow, ..
- Not always a problem: receiver might be fast enough.



76

Adding Flow Control

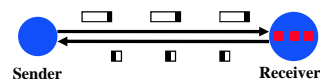
- Stop and wait flow control: sender waits to send the next packet until the previous packet has been acknowledged by the receiver.
 - › Receiver can pace the receiver
- Drawbacks: adds overheads, slowdown for long links.



77

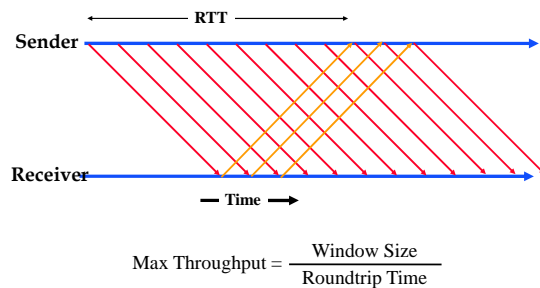
Window Flow Control

- Stop and wait flow control results in poor throughput for long-delay paths: packet size/roundtrip-time.
- Solution: receiver provides sender with a window that it can fill with packets.
 - › The window is backed up by buffer space on receiver
 - › Receiver acknowledges the a packet every time a packet is consumed and a buffer is freed



78

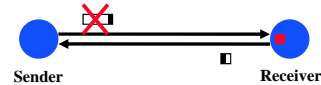
Bandwidth-Delay Product



79

Dealing with Errors Stop and Wait Case

- Packets can get lost, corrupted, or duplicated.
 - » Error detection or correction turns corrupted packet in lost or correct packet
- Duplicate packet: use sequence numbers.
- Lost packet: time outs and acknowledgements.
 - » Positive versus negative acknowledgements
 - » Sender side versus receiver side timeouts
- Window based flow control: more aggressive use of sequence numbers (see transport lectures).



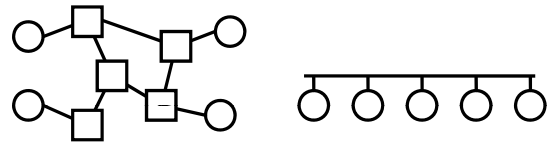
80

What is Used in Practice?

- No flow or error control.
 - » E.g. regular Ethernet, just uses CRC for error detection
- Flow control only.
 - » E.g. Gigabit Ethernet
- Flow and error control.
 - » E.g. X.25 (older connection-based service at 64 Kbs that guarantees reliable in order delivery of data)

81

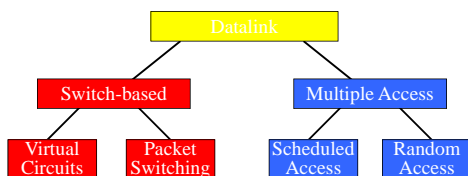
Datalink Layer Architectures



- Packet forwarding.
- Error and flow control.
- Media access control.
- Scalability.

82

Datalink Classification



83

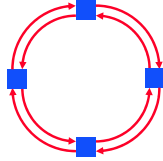
Multiple Access Protocols

- Prevent two or more nodes from transmitting at the same time over a broadcast channel.
 - » If they do, we have a collision, and receivers will not be able to interpret the signal
- Several classes of multiple access protocols.
 - » Partitioning the channel, e.g. frequency-division or time division multiplexing
 - With fixed partitioning of bandwidth –
 - Not flexible; inefficient for bursty traffic
 - » Taking turns, e.g. token-based, reservation-based protocols, polling based
 - » Contention based protocols, e.g. Aloha, Ethernet
 - Next lecture

84

Fiber Distributed Data Interface (FDDI)

- One token holder may send, with a time limit
 - » Provides known upper bound on delay.
- Optical version of 802.5 token ring, but multiple packets may travel in train: token released at end of frame
- 100 Mbps, 100km
- Optional dual ring for fault tolerance
- Concerns:
 - » Token overhead
 - » Latency
 - » Single point of failure



85

Other "Taking Turn" Protocols

- Central entity polls stations, inviting them to transmit
 - » Simple design – no conflicts
 - » Not very efficient – overhead of polling operation
 - » Example: the "Point Control Function" mode for 802.11
- Stations reserve a slot for transmission.
 - » For example, break up the transmission time in contention-based and reservation based slots
 - Contention based slots can be used for short messages or to reserve time slots
 - Communication in reservation based slots only allowed after a reservation is made
 - » Issues: fairness, efficiency

86

MAC Protocols - Discussion

- Channel partitioning MAC protocols:
 - » Share channel efficiently at high load
 - » Inefficient at low load: delay in channel access, 1/N bandwidth allocated even if only 1 active node!
- "Taking turns" protocols
 - » More flexible bandwidth allocation, but
 - » Protocol can introduce unnecessary overhead and access delay at low load
- Random access MAC protocols (next lecture)
 - » Efficient at low load: single node can fully utilize channel
 - » High load: collision overhead

87