CSCI 466: Networks

Link Layer

Reese Pearsall

Fall 2022

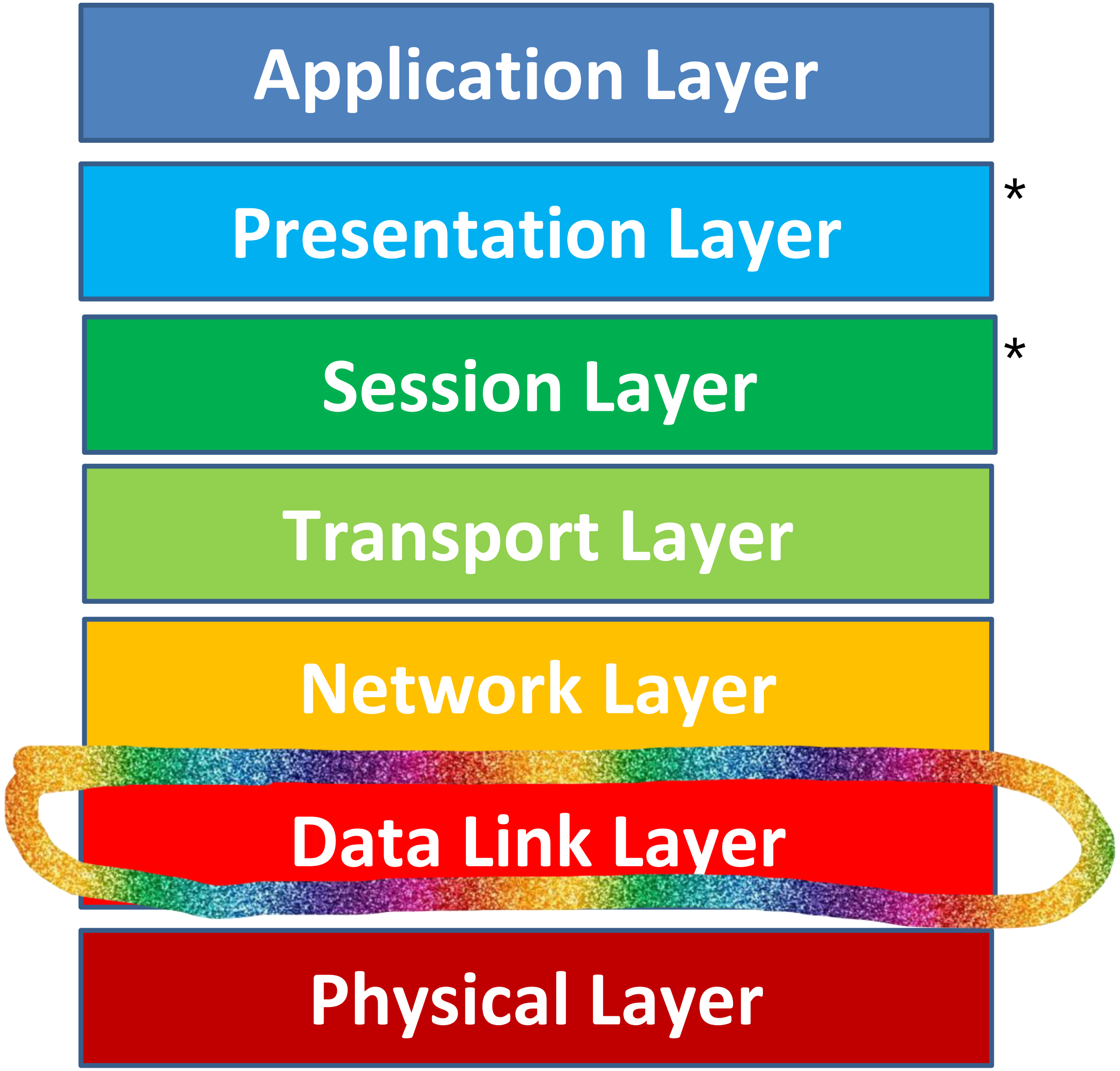
**All images are stolen from the internet*

Announcements

NO CLASS next Friday 11/11

Might have to also cancel next Monday (11/7)*

OSI Model



Application Layer

Presentation Layer *

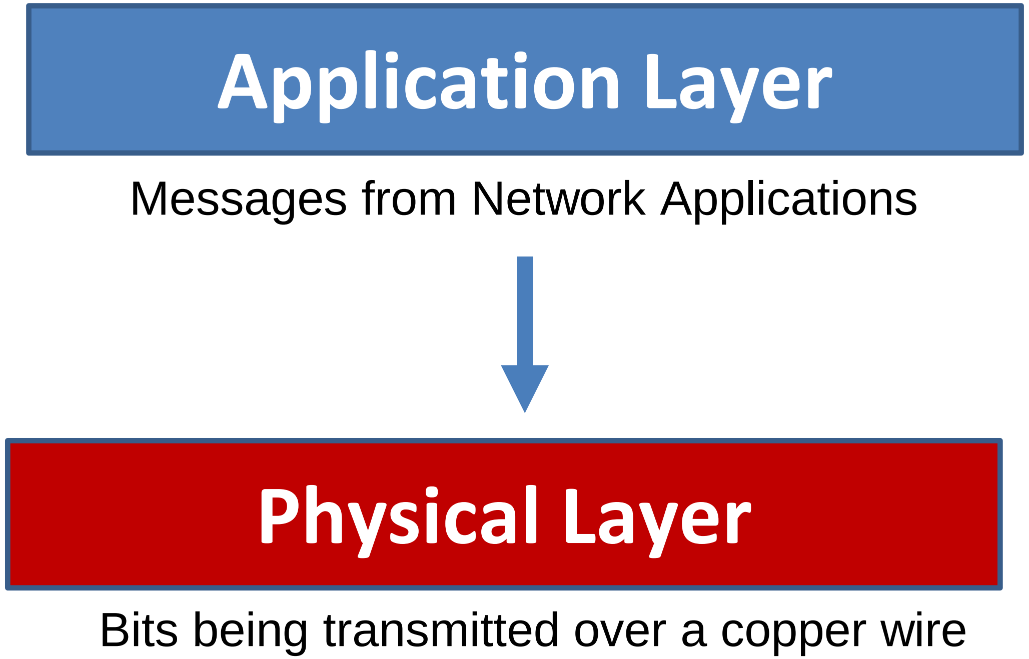
Session Layer *

Transport Layer

Network Layer

Data Link Layer

Physical Layer



Application Layer

Messages from Network Applications

Physical Layer

Bits being transmitted over a copper wire

*In the textbook, they condense it to a 5-layer model, but 7 layers is what is most used

Data Link Layer

The link layer is responsible for the **actual node-to-node delivery** of data and ensure error-free transmission of information

terminology:

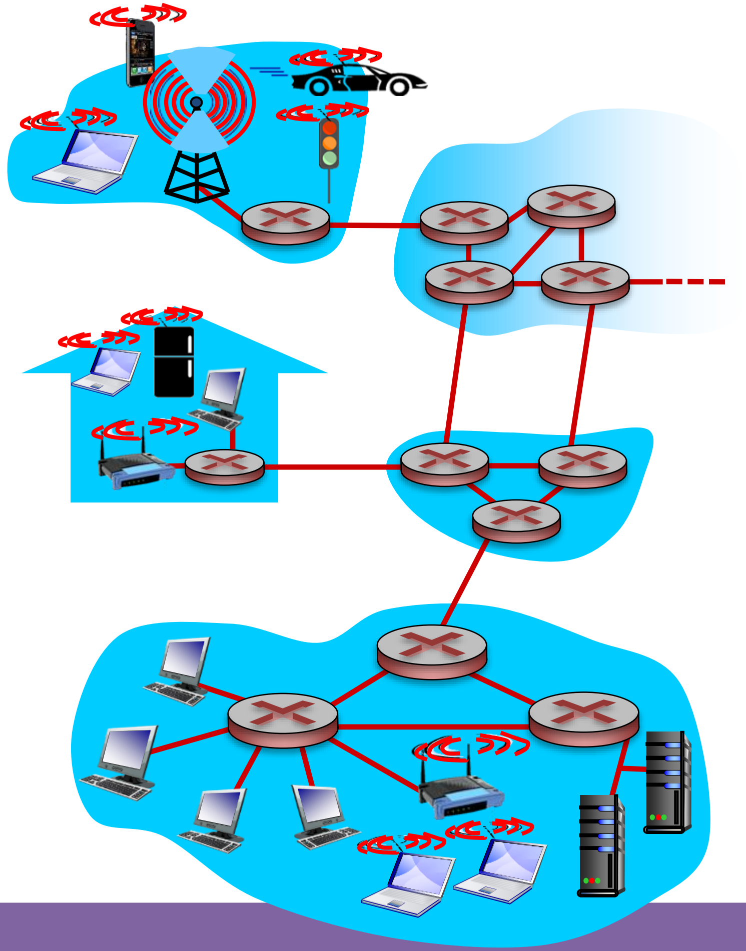
hosts and routers: **nodes**

communication channels that connect adjacent nodes along communication path: **links**

- wired links
- wireless links
- LANs

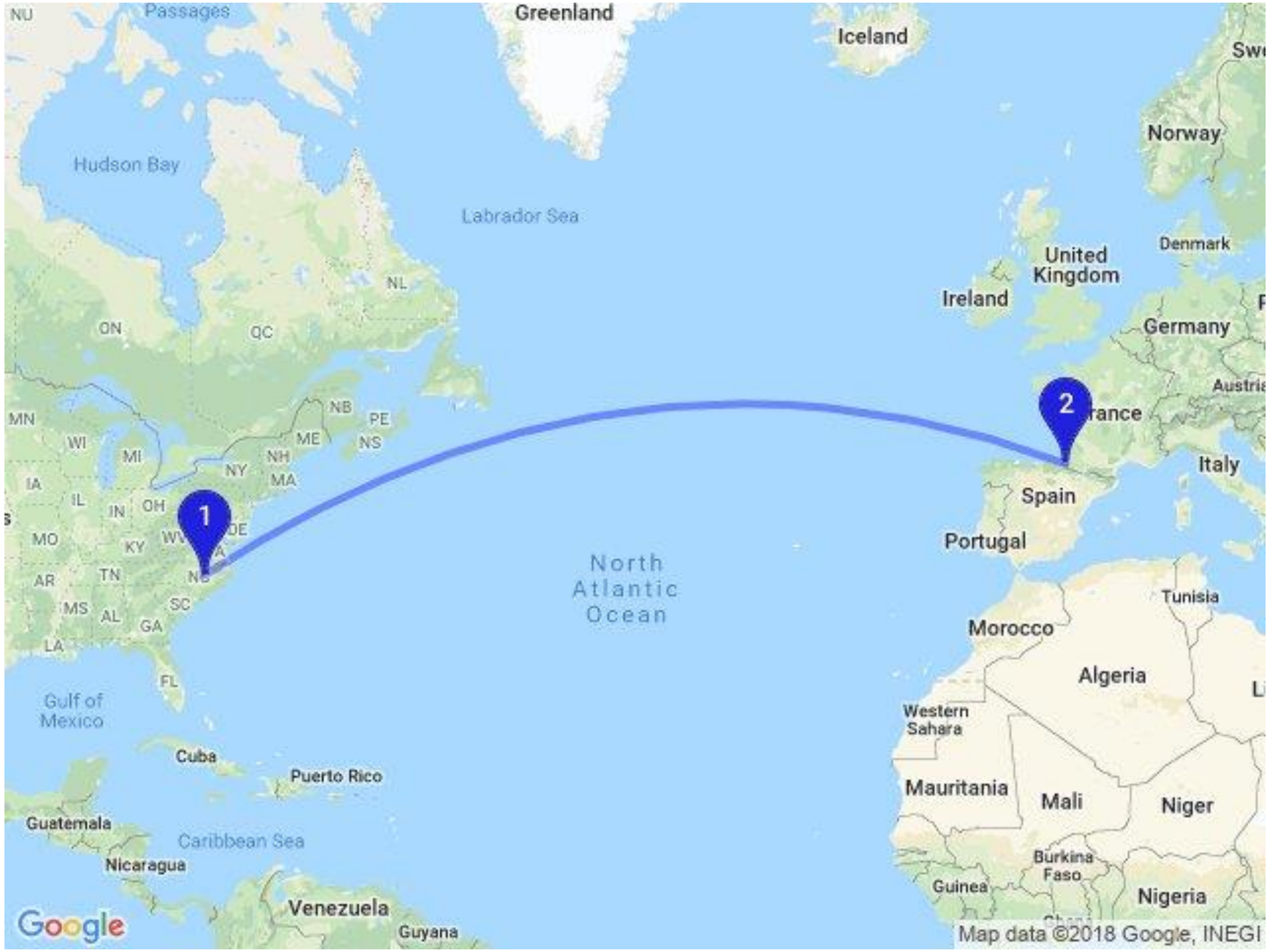
layer-2 packet: frame, encapsulates **datagram**

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



We have not addressed how we will overcome various transmission mediums!

Data Link Layer

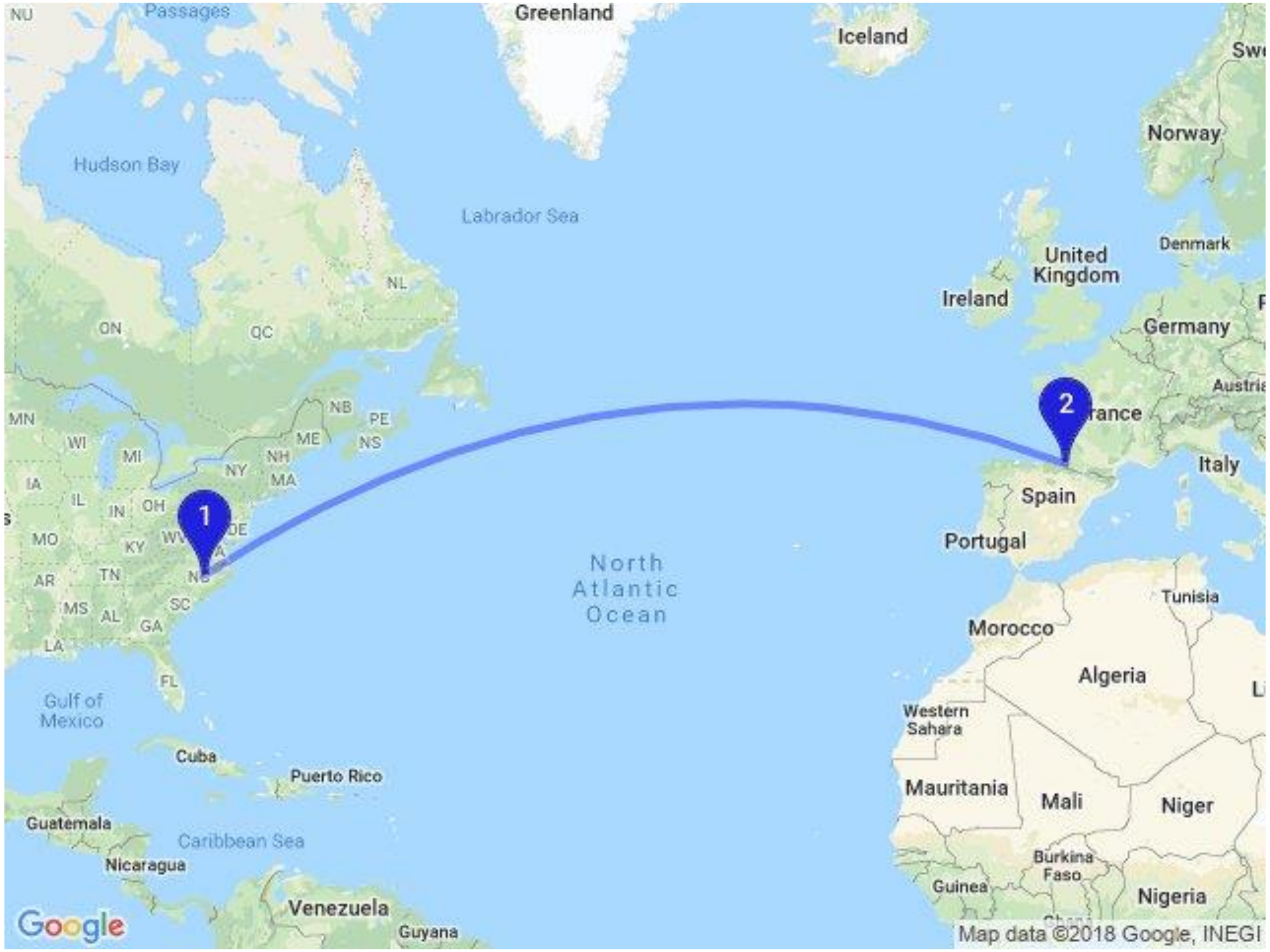


Ways to get form US to Paris?

We can visit a travel agent that will give us a travel plan

1. Take a car to the airport
2. Take a plane to France
3. Take a train from Airport to Paris
4. Take a bus from the train stop to the Eifel tower

Data Link Layer



Ways to get form US to Paris?

We can visit a travel agent that will give us a travel plan

1. Take a car to the airport
2. Take a plane to France
3. Take a train from Airport to Paris
4. Take a bus from the train stop to the Eifel tower

- Tourist = Datagram
- Transportation Segment = Link
- Airport, Bus Stop, Train Stop = Node
- Transport Mode = Link Layer Protocol
- Travel Agent = Routing Protocol (Network Layer)

Data Link Layer

Services offered by the Link Layer

- Framing
  - → Encapsulate a network layer Datagram in *another* header
- Link access
  - → LL dictate the rules and process of transmitting a frame over a link
- Reliable Delivery
  - → For unreliable link, some reliable delivery mechanisms may need to be used
- Error Detection and Correction
  - → Bits can get messed up as the are transmitted through a medium

Why do we need RDT and error detection in the link layer when it is also offered in the transport layer?

Data Link Layer

Services offered by the Link Layer

- Framing
  - → Encapsulate a network layer Datagram in *another* header
- Link access
  - → LL dictate the rules and process of transmitting a frame over a link
- Reliable Delivery
  - → For unreliable link, some reliable delivery mechanisms may need to be used
- Error Detection and Correction
  - → Bits can get messed up as the are transmitted through a medium

Why do we need RDT and error detection in the link layer when it is also offered in the transport layer?

Some packets of data don’t even travel through the transport layer…

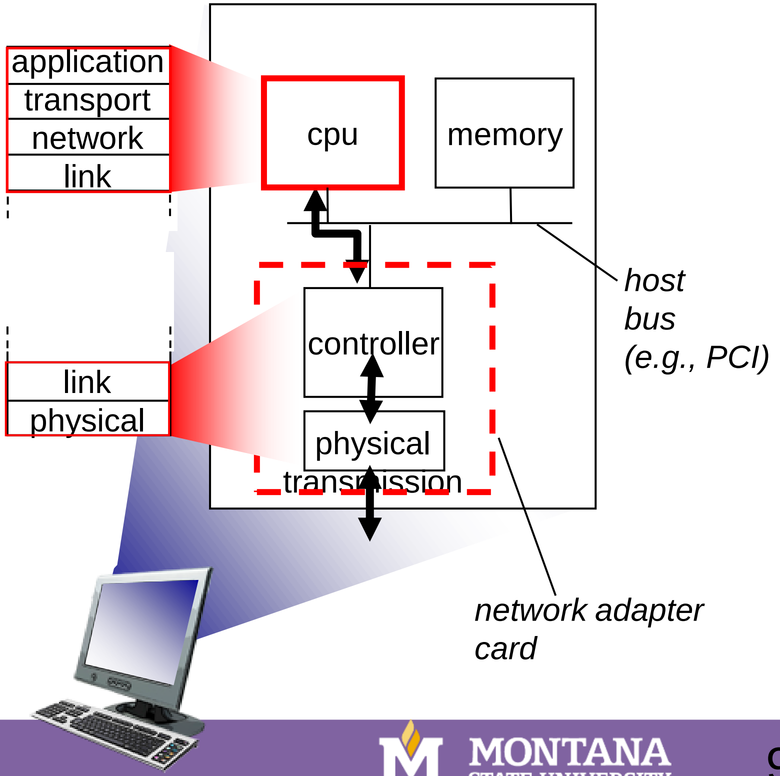
Data Link Layer

Implementation of Link Layer

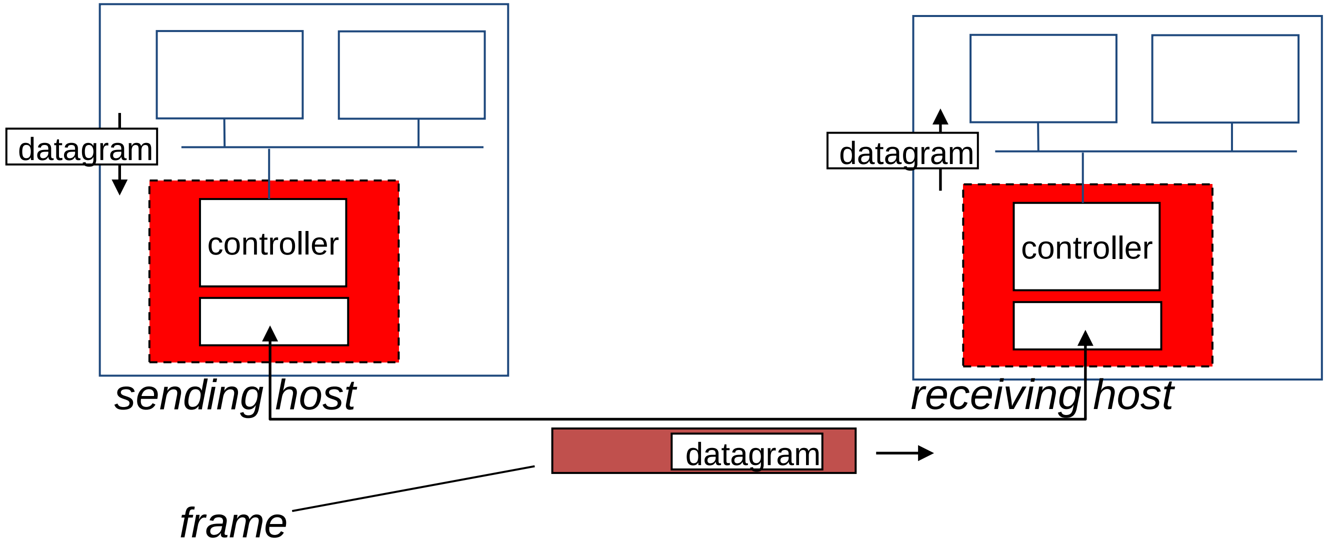
- Implemented within the hardware of your computer

NIC (Network Interface Controller)- Integrated into the motherboard and allows the machine to use LL services such as ethernet (combination of hardware, software, and some firmware)

Wireshark uses your NIC to determine which packets should be sniffed!



Data Link Layer



sending side:

encapsulates datagram in frame

adds error checking bits, rdt, flow control, etc.

receiving side

looks for errors, rdt, flow control, etc.

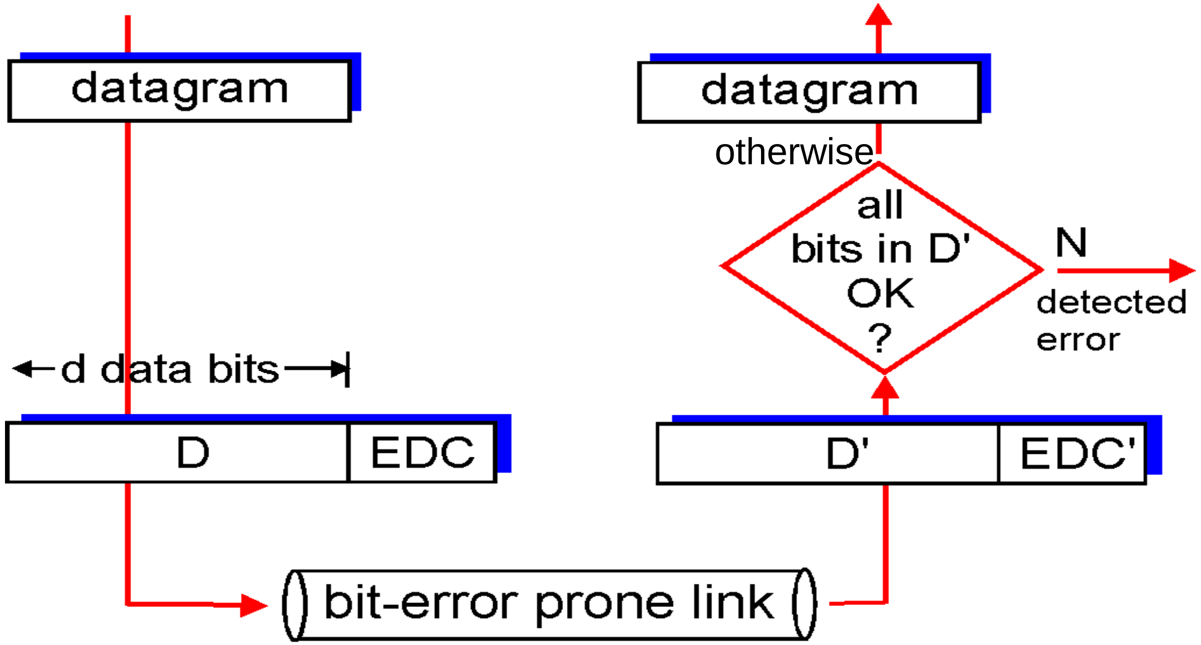
extracts datagram, passes to upper layer at receiving side

Data Link Layer

Bits can get messed during the physical layer and link layer

- Faulty wires
- NIC issues
- Unreliable mediums

The Data Link Layer implements services for **detecting** and **correcting errors**



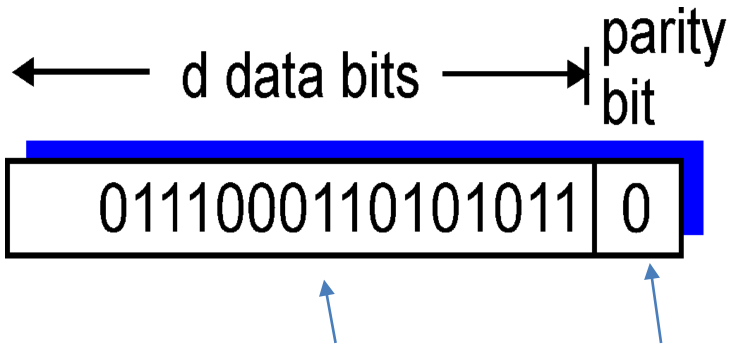
EDC= Error Detection and Correction bits
D = Data protected by error checking, may include header fields

- Error detection not 100% reliable!
- protocol may miss some errors, but rarely
- larger EDC field yields better detection and correction

Data Link Layer

Single bit parity:

Detect single bit errors



d data bits

parity bit

0111000110101011

0

Data

Is the number of 1s odd (0) or even (1) ? (Including the parity bit)

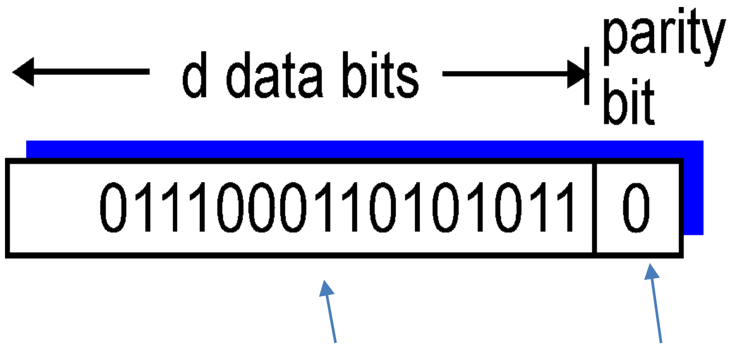
0111000100101011|0

The receiver counts eight 1s, but the parity bit tells us it should be an odd number of 1s → **ERROR DETECTED**

Data Link Layer

Single bit parity:

Detect single bit errors



Data

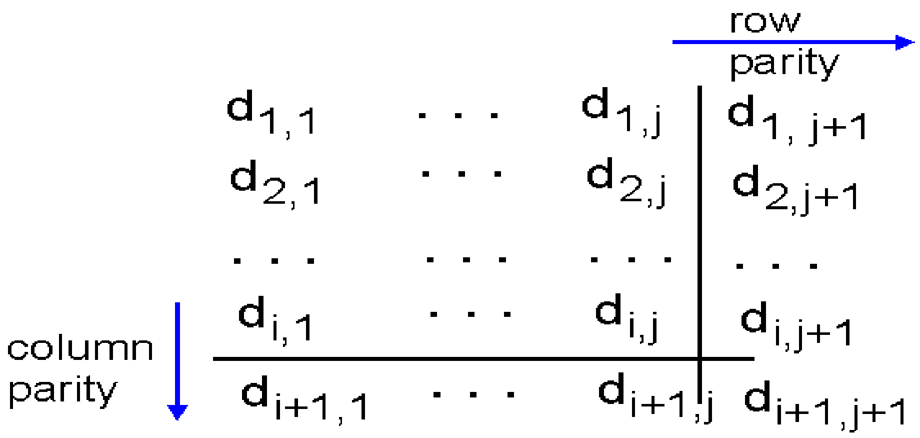
Is the number of 1s odd (0) or even (1) ? (Including the parity bit)

0111000**0**0101011|0

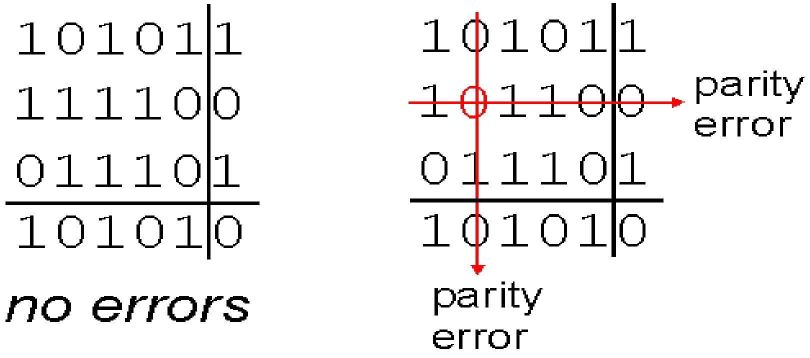
The receiver counts eight 1s, but the parity bit tells us it should be an odd number of 1s → **ERROR DETECTED**

Two-dimensional bit parity:

Detect and **correct** single bit errors



$$\begin{array}{cccc|c} d_{1,1} & \cdots & & d_{1,j} & d_{1,j+1} \\ d_{2,1} & \cdots & & d_{2,j} & d_{2,j+1} \\ \cdots & \cdots & & \cdots & \cdots \\ d_{i,1} & \cdots & & d_{i,j} & d_{i,j+1} \\ \hline d_{i+1,1} & \cdots & & d_{i+1,j} & d_{i+1,j+1} \end{array}$$



```
10101|1
11110|0
01110|1
-------
10101|0
```

no errors

```
10101|1
10110|0
01110|1
-------
10101|0
```

correctable single bit error

Data Link Layer

Checksum

(Sender)

0110011001100000

+ 0101010101010101

+ 1000111100001100

0100101011000010 Binary sum of words

(one’s complement)

1011010100111101 Checksum!

(Receiver)

0110011001100000

+ 0101010101010101

+ 1000111100001100

+ 0100101011000010

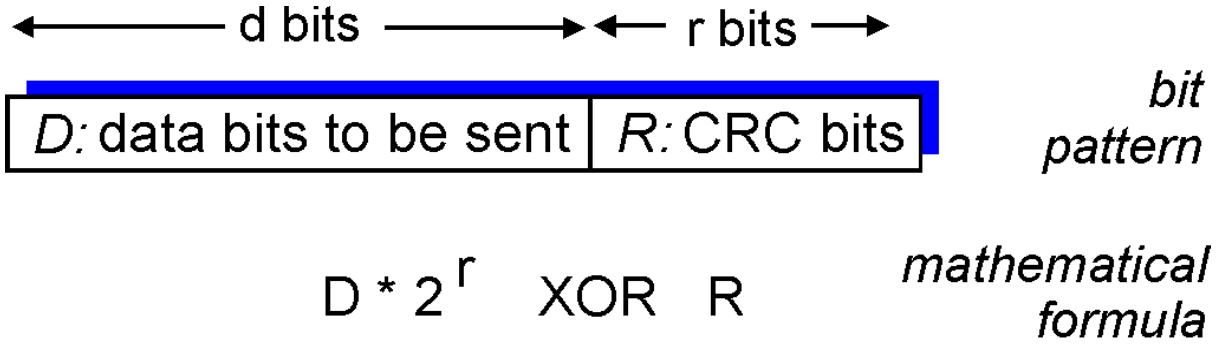
(Binary Sum → One’s Complement)

= 1111111111111111

All 1s = No error!

Data Link Layer

- more powerful error-detection coding
- view data bits, D, as a binary number
- choose r+1 bit pattern (generator), G
- goal: choose r CRC bits, R, such that
  - ❑ <D,R> exactly divisible by G (modulo 2)
  - ❑ receiver knows G, divides <D,R> by G. If non-zero remainder: error detected!
  - ❑ can detect all burst errors less than r+1 bits
- widely used in practice (Ethernet, 802.11 WiFi, ATM)



← d bits → ← r bits →

D: data bits to be sent | *R:* CRC bits

bit pattern

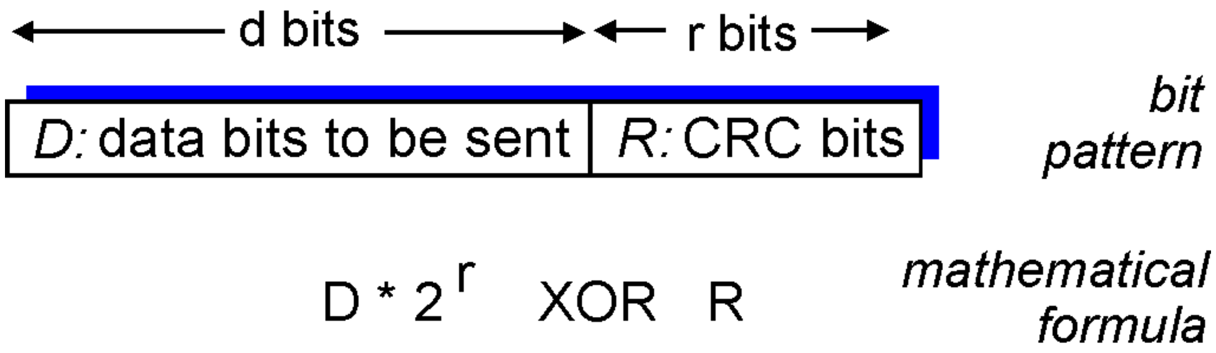
$D * 2^r \text{ XOR } R$

mathematical formula

Sender/Receiver has D and G. Need to compute R

Data Link Layer

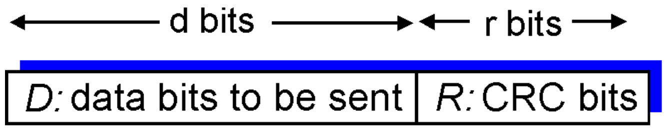
- more powerful error-detection coding
- view data bits, D, as a binary number
- choose r+1 bit pattern (generator), G
- goal: choose r CRC bits, R, such that
  - □ <D,R> exactly divisible by G (modulo 2)
  - □ receiver knows G, divides <D,R> by G. If non-zero remainder: error detected!
  - □ can detect all burst errors less than r+1 bits
- widely used in practice (Ethernet, 802.11 WiFi, ATM)



$D * 2^r \quad \text{XOR} \quad R$

Sender/Receiver has D and G. Need to compute R

Data Link Layer



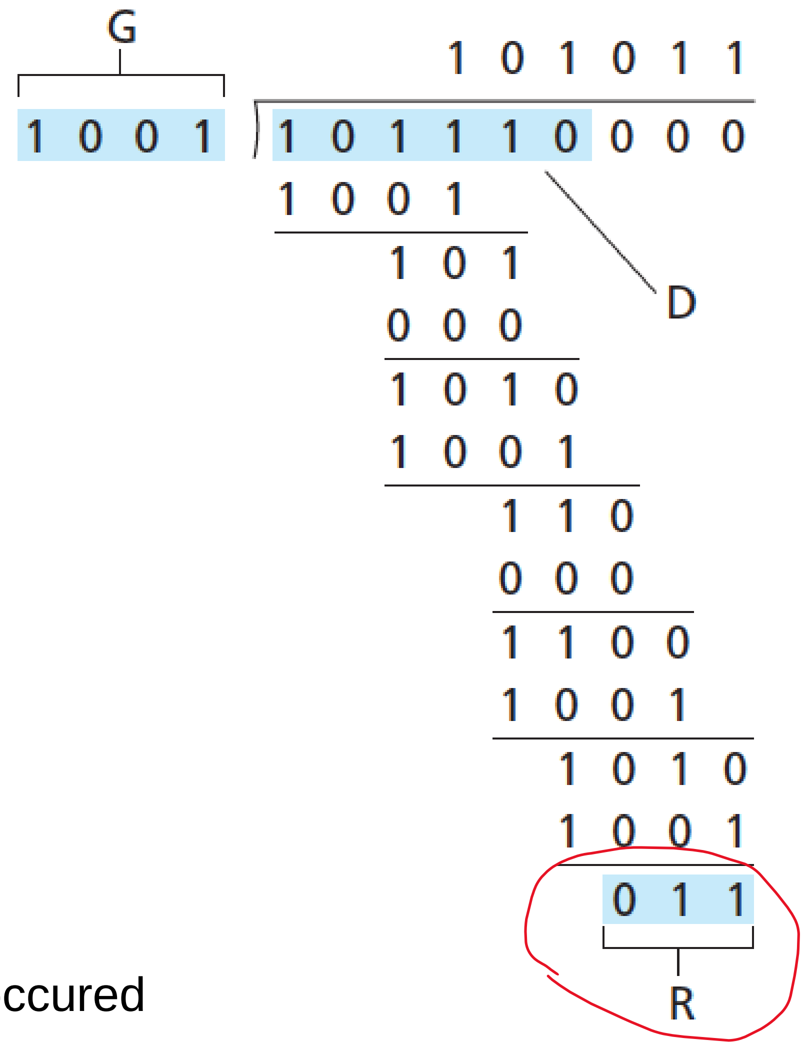
bit pattern

$D * 2^r \quad \text{XOR} \quad R$

mathematical formula

(Do some algebra to find R)

$$R = remainder[\frac{D \cdot 2^r}{G}]$$



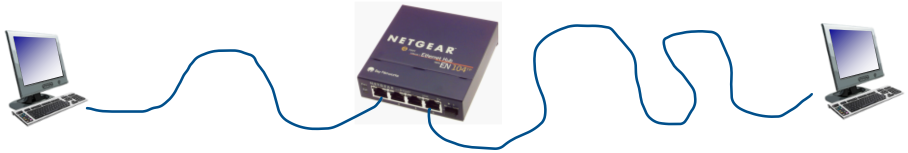
Sender sends D + R bits.

Receiver divides D + R bits by G. Result should always be Zero if no errors occured

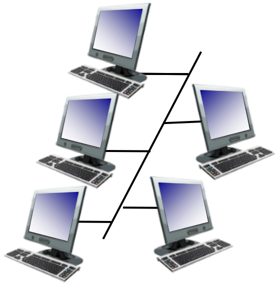
Data Link Layer

Access links

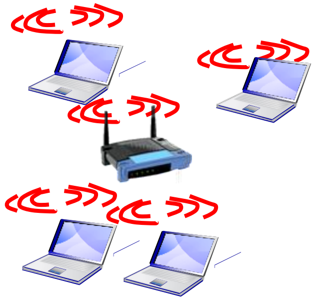
- Point to Point – Single sender, Single Receiver at each end of link



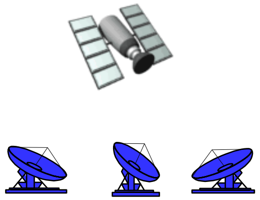
- Broadcast – shared medium



shared wire (e.g., cabled Ethernet)



shared RF (e.g., 802.11 WiFi)



shared RF (satellite)

humans at a cocktail party (shared air, acoustical)

Data Link Layer

MAC (Media Access Control) Addresses

32-bit IP address:

network-layer address for interface

used for layer 3 (network layer) forwarding

MAC (or LAN or physical or Ethernet) address:

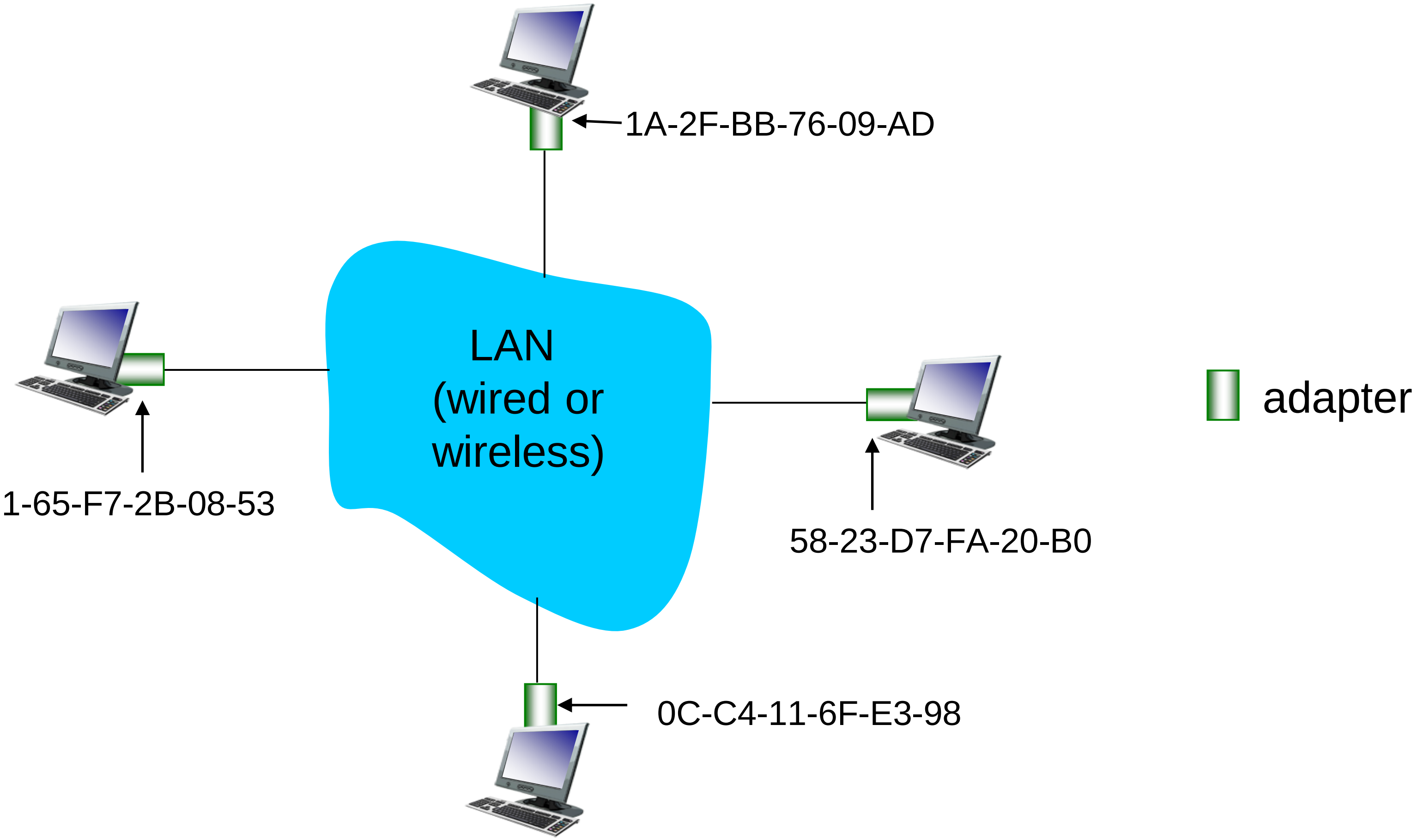
- function: *used 'locally" to get frame from one interface to another physically-connected interface (same network, in IP-addressing sense)*
- 48 bit MAC address (for most LANs) burned in NIC ROM, also sometimes software settable
- e.g.: 1A-2F-BB-76-09-AD

hexadecimal (base 16) notation
(each "numeral" represents 4 bits)

How do we know two NICs wont have the same MAC address?

MAC Address is your SSN, IP address is your Postal code ☺

Data Link Layer


1A-2F-BB-76-09-AD
LAN
(wired or
wireless)
71-65-F7-2B-08-53
58-23-D7-FA-20-B0
0C-C4-11-6F-E3-98
adapter