Chapter 2
Getting Connected

Five Issues for Physical Connection

- To successfully exchange packets, five issues need to be solved
  - **Encoding:** map bits onto the signal on wire or fiber (medium)
  - **Framing:** assemble bits into a frame (complete message)
  - **Error detection:** detect the corruption during transmission
  - **Reliable delivery:** retransmit a message when it is failed
  - **Medium access control:** manage the medium shared by multiple hosts

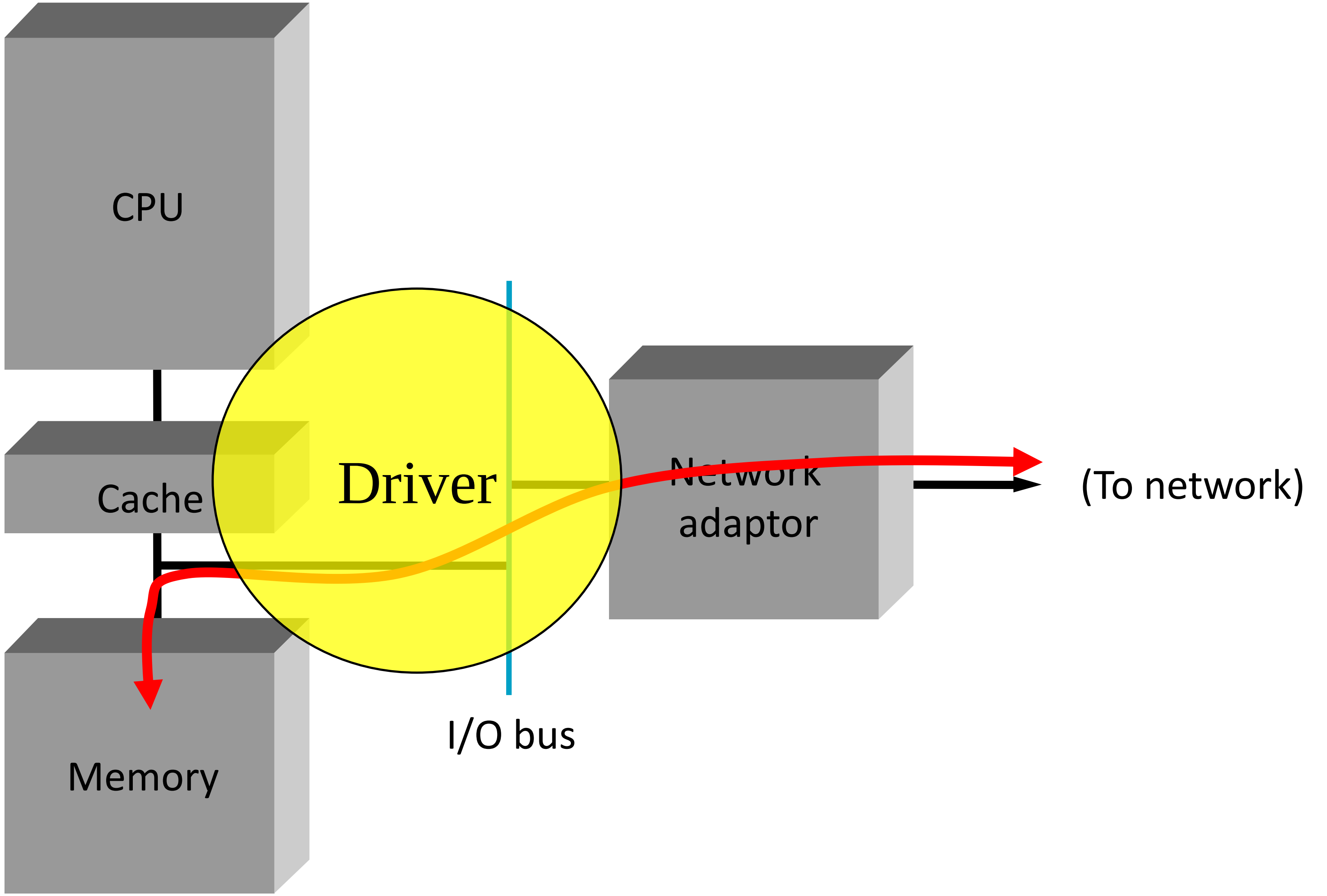
Network Technologies

- There are four specific network technologies, including
  - **Point-to-point** links
  - **Carrier Sense Multiple Access (CSMA)** networks: **Ethernet**
  - **Token rings:** IEEE Standard **802.5** and **FDDI**
  - **Wireless** networks**:** IEEE Standards **802.11, 802.16**

Nodes

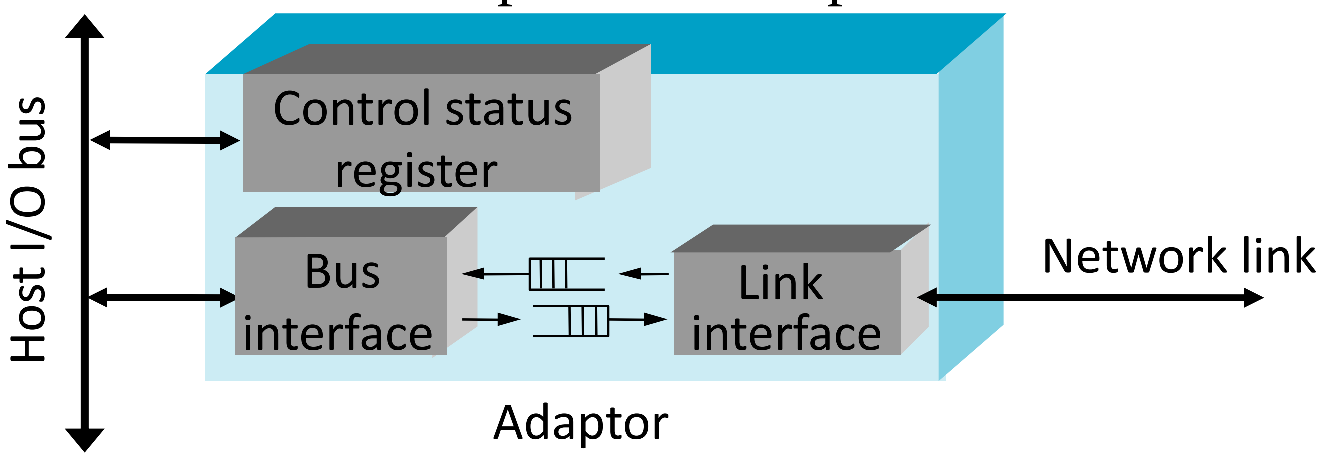
- A network node is a computer connecting to the network
  - Node connects the network via a **network adaptor**
- **A network adaptor**
  - Delivers data between the memory and the network link
  - The device **driver** manages the adaptor
- Network performance:
  - As a network node, the node runs at **memory speed**, not CPU speed
  - Memory access will limit the network transmission

Nodes



Network Adaptor

- Functions of a network adaptor contain:
  - **Point-to-point automatic repeat request (ARQ):**
    - The **lowest-level protocol** on the host
  - **Framing**, **error detection**, and **media access control**
- Two main components in a network adaptor:
  - **Bus interface:** to communicate with the host
  - **Link interface:** to speak correct protocol on the network



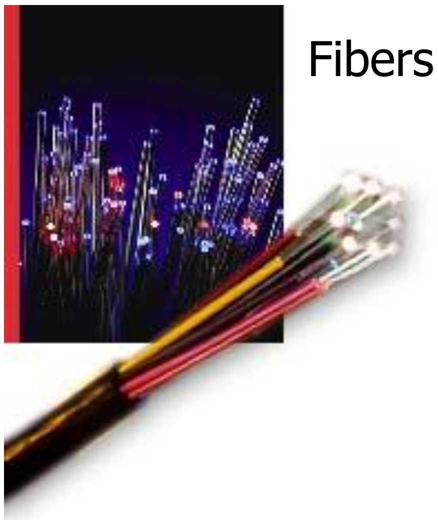
Network Adaptor Components

- **Bus interface:** designed for a specific I/O bus
  - Defines a protocol that is used by
    - the host's CPU to program the adaptor
    - the adaptor to interrupt the host's CPU
    - the adaptor to read and write memory on the host
- **Link interface:** such as Ethernet
  - Implemented by
    - A chip set, or
    - A field-programmable gate array (FPGA)
- The host's bus and the network link are running at **different speeds**
  - A small amount of **buffer** is required (FIFO queue)

Network Components

Links

Fibers



Coaxial
cable

Interfaces

Ethernet card

Wireless card

Switches/routers

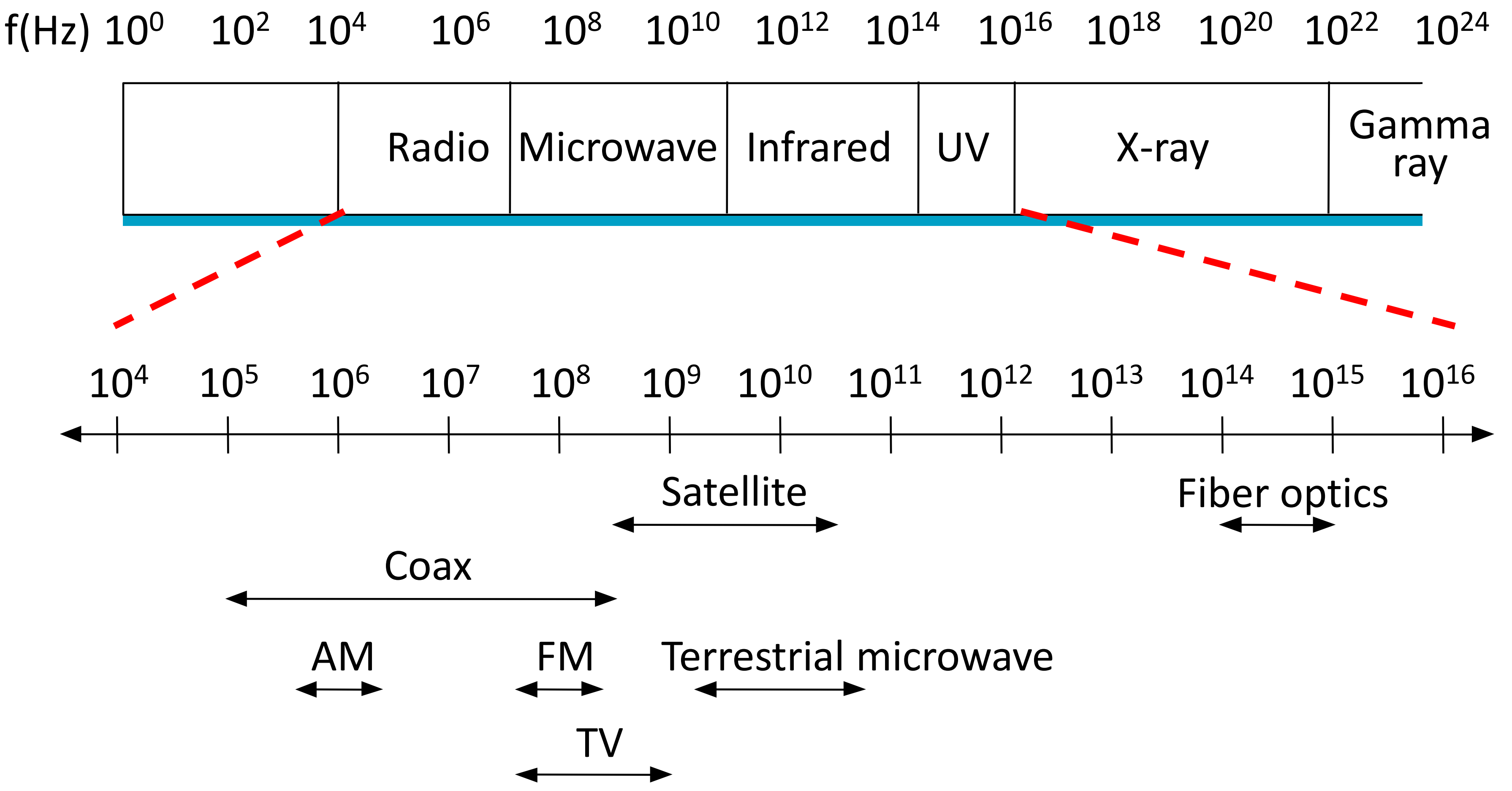
Large
router

Telephone
switch

Physical Medium

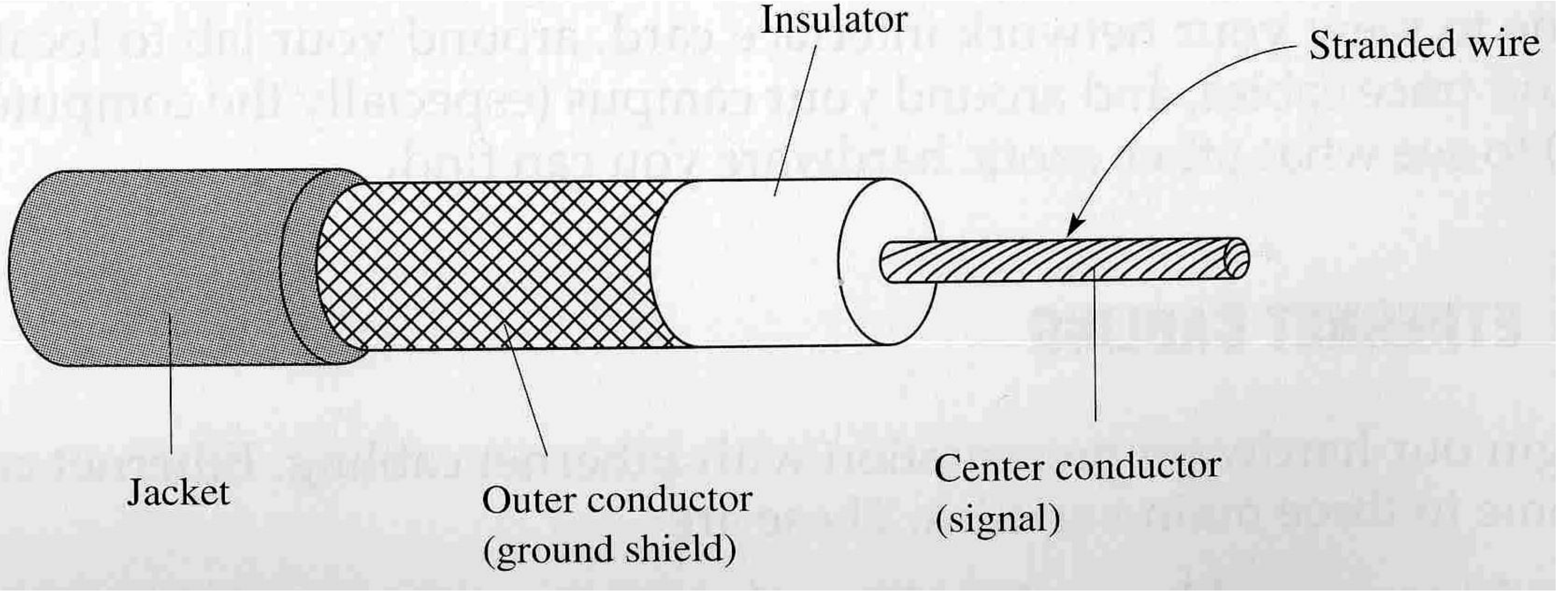
Links (Medium)

- **Space:** for wireless links

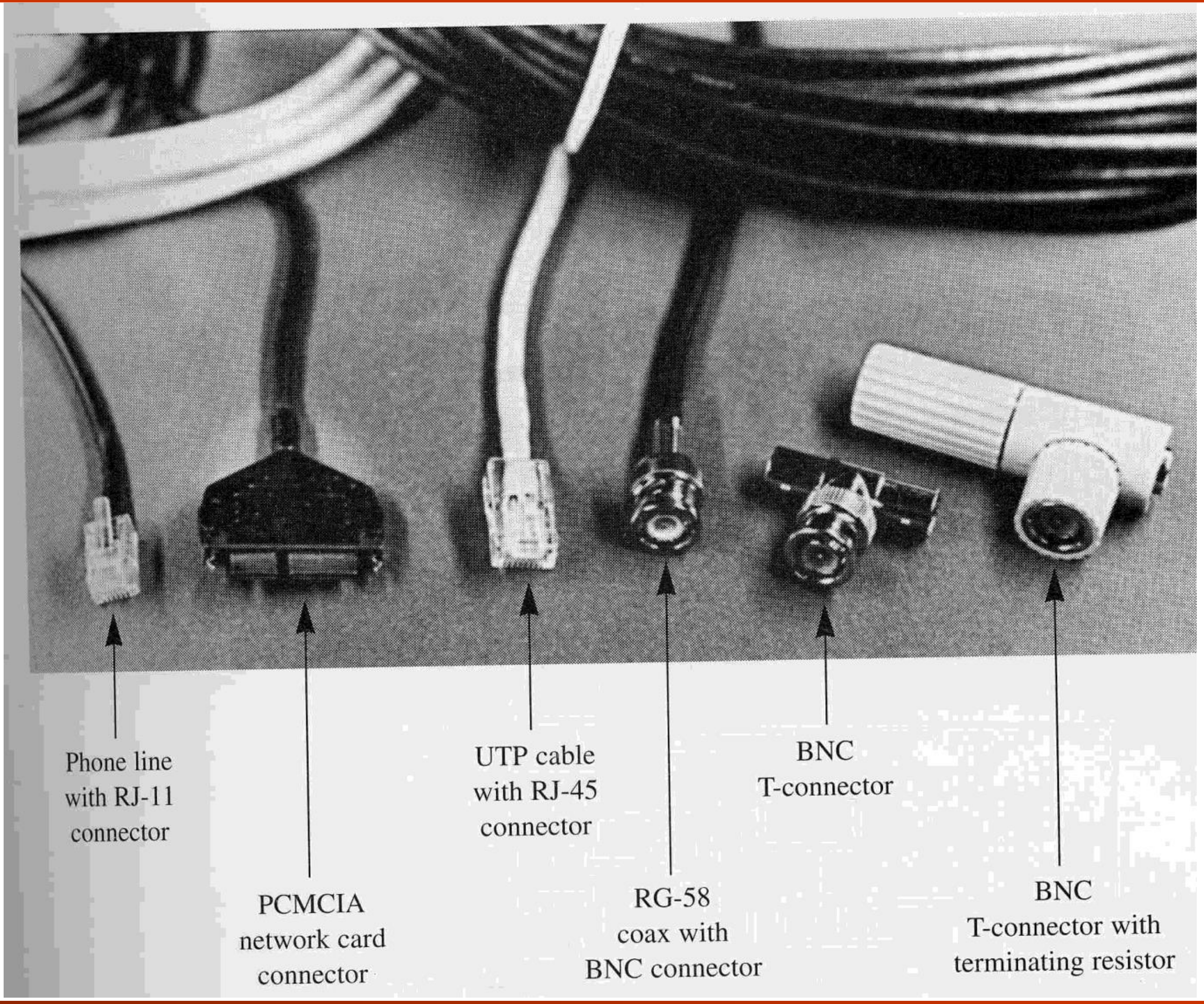


Links (Medium)

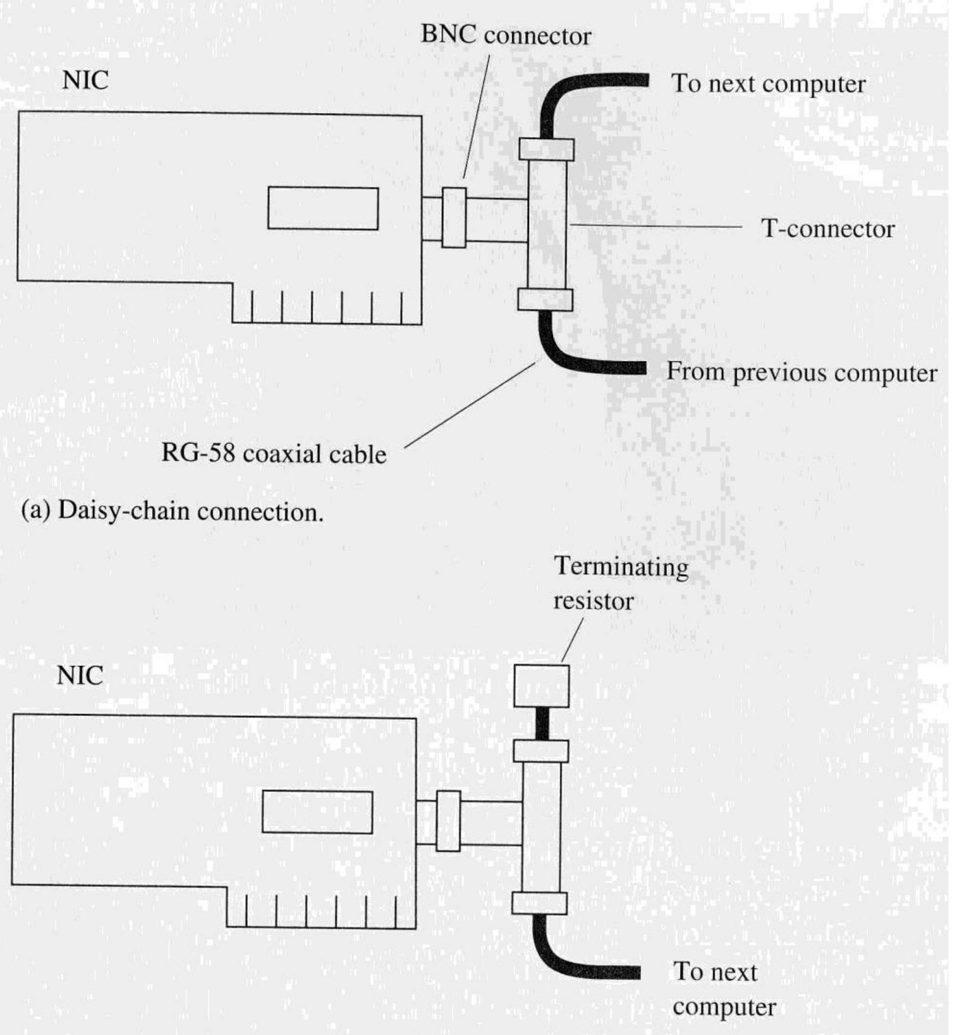
- **Twisted pair:** telephone wire
- **Coaxial cable:** TV cable
- **Optical fiber:** high-bandwidth and long-distance link



Links (Medium)



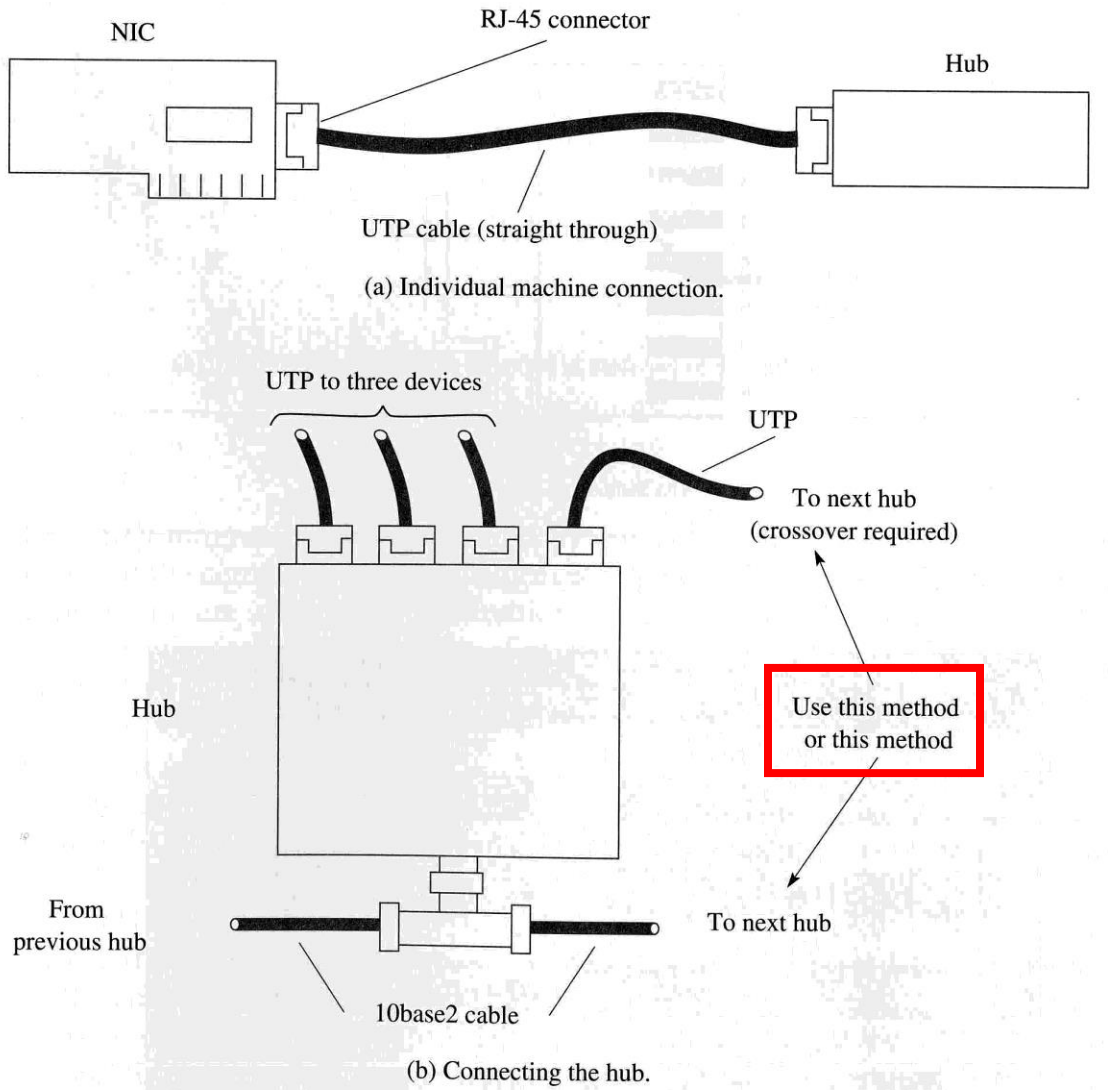
10Base2 Ethernet Wiring



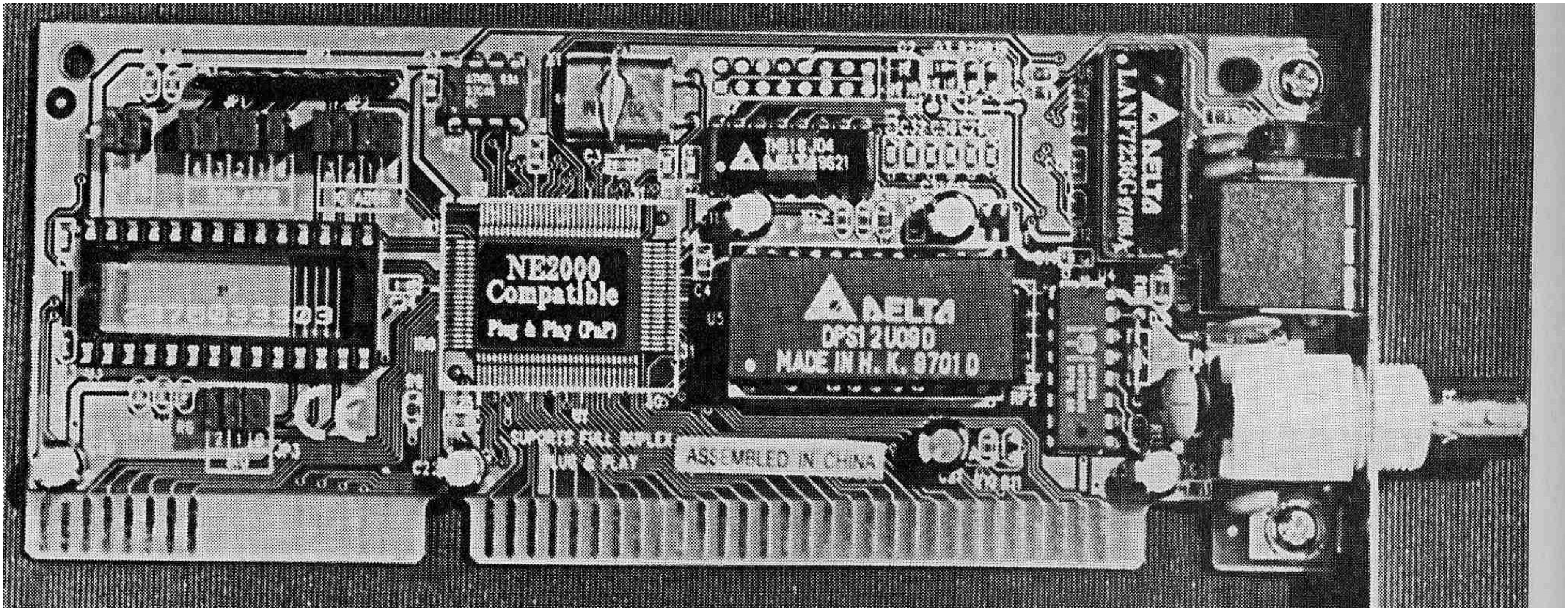
(a) Daisy-chain connection.

(b) Terminating connection (required at each end of the cable).

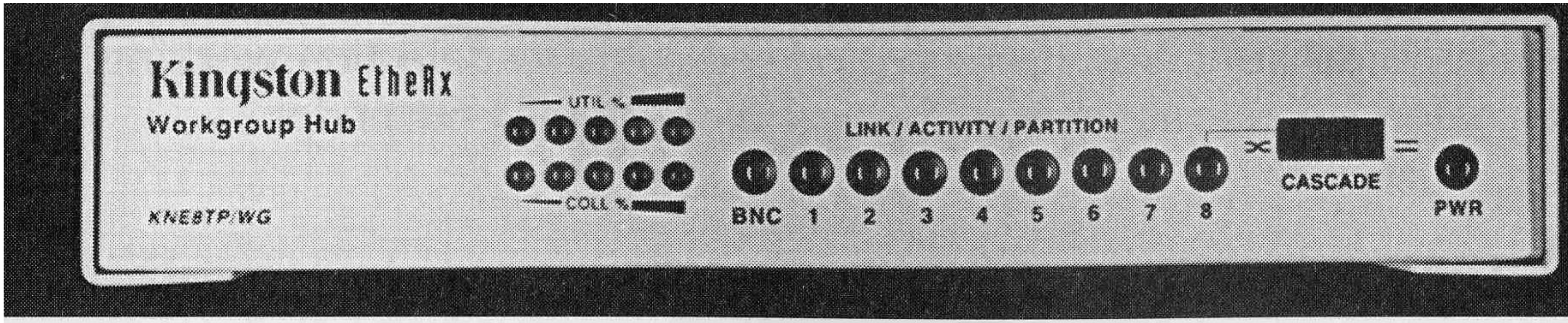
Hub and User Terminal



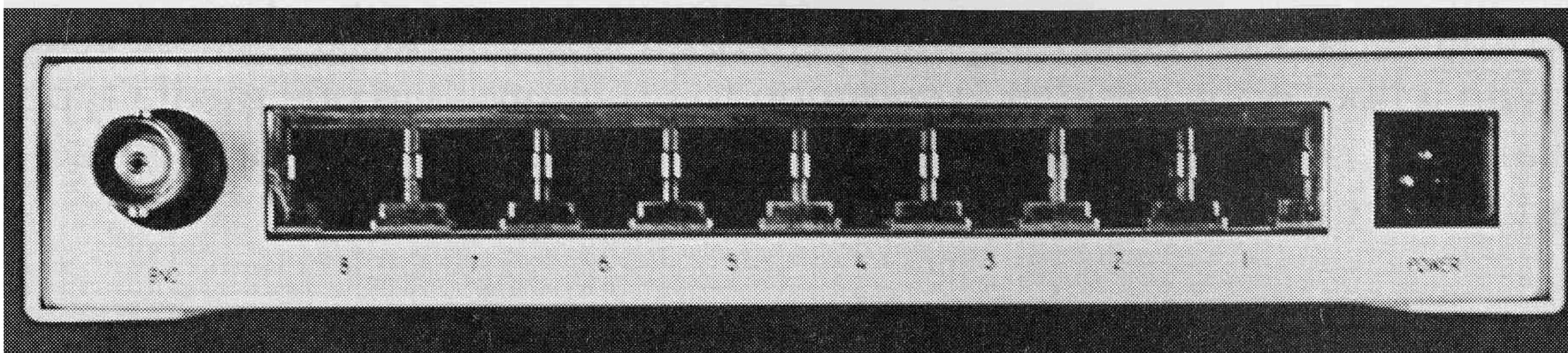
Network Interface Card



Hub

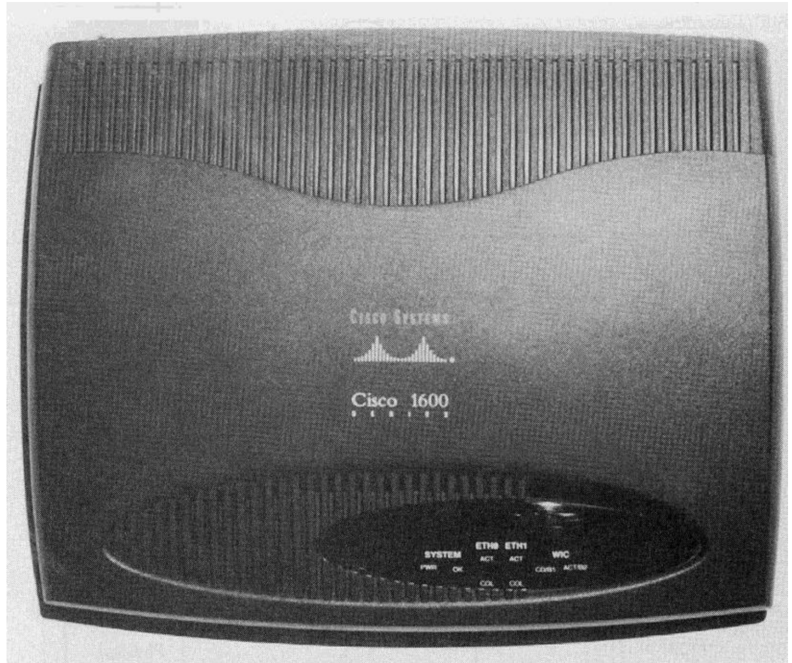


Front view.



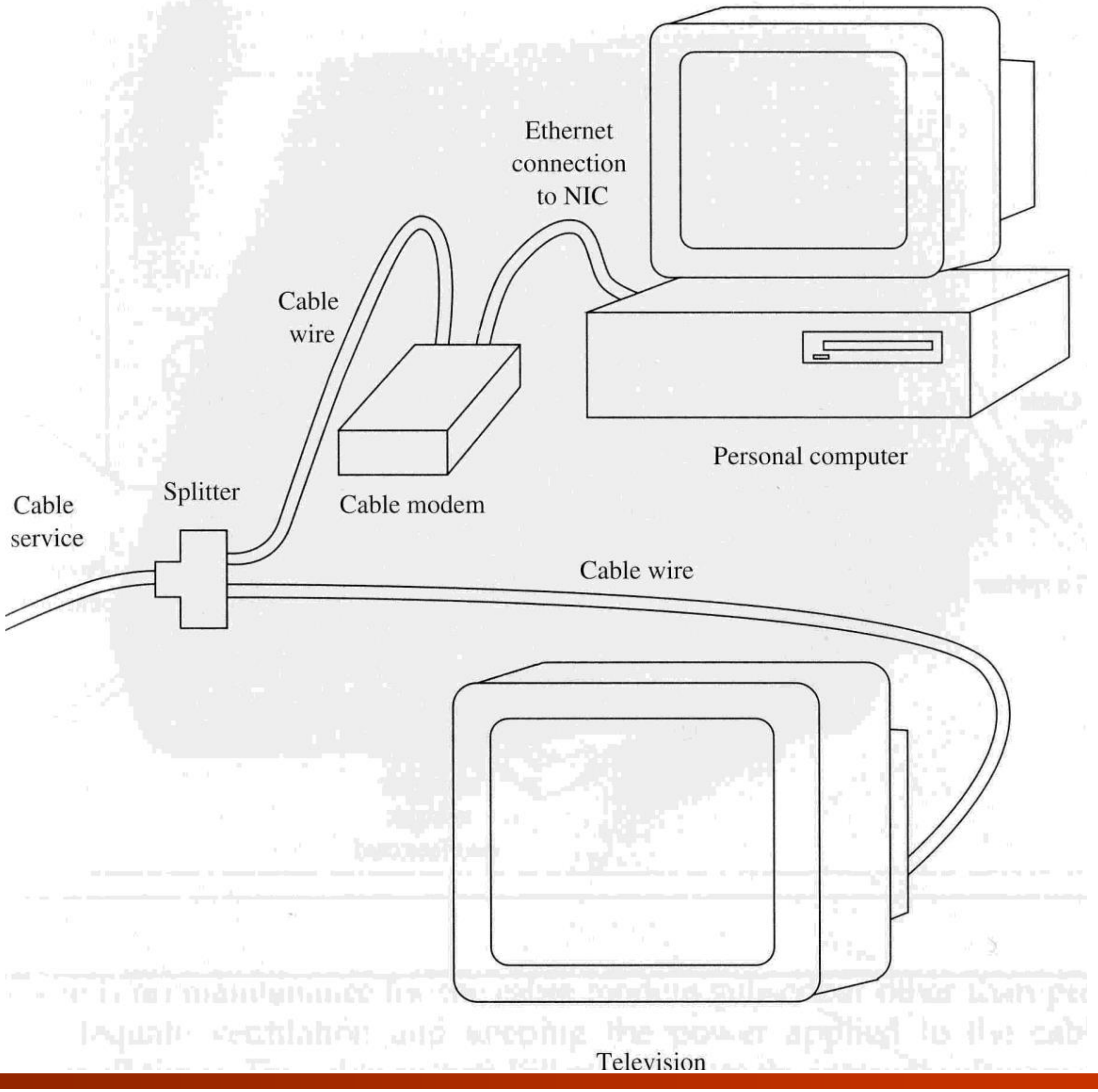
Rear view.

Cisco 1600 Router



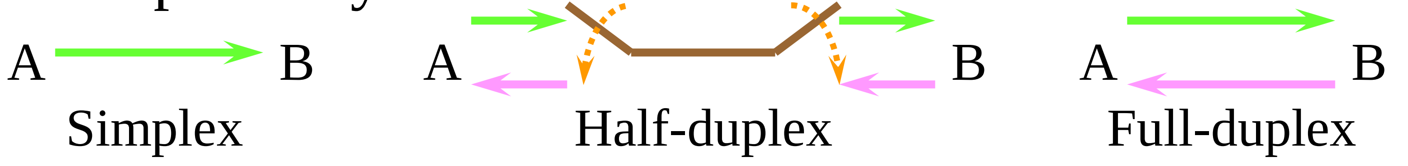
(b)

Cable Modem

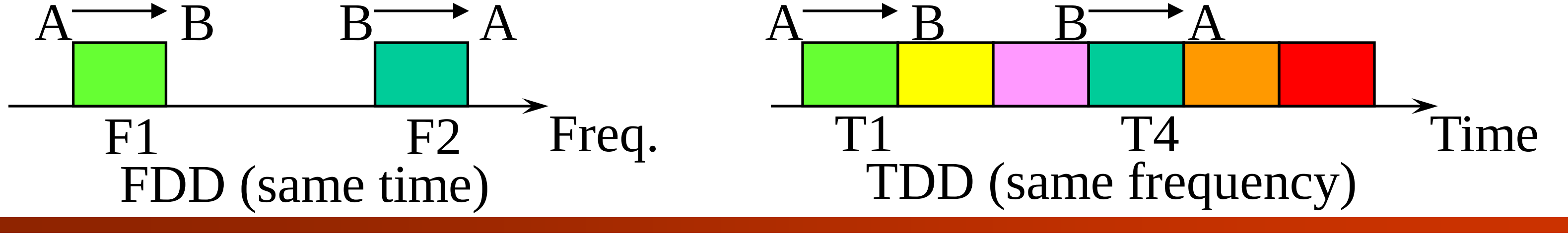

Ethernet
connection
to NIC
Cable
wire
Cable modem
Personal computer
Cable
service
Splitter
Cable wire
Television

Duplex

- **Simplex:** one-way communications
  - Paging system (Now have two-way paging system)
- **Half-duplex:** two-way comm. (One-way at any instant)
  - Dispatch communications system
- **Full-duplex:** two-way communications (at any instant)
  - Telephone system



- **FDD:** Frequency Division Duplex
- **TDD:** Time Division Duplex



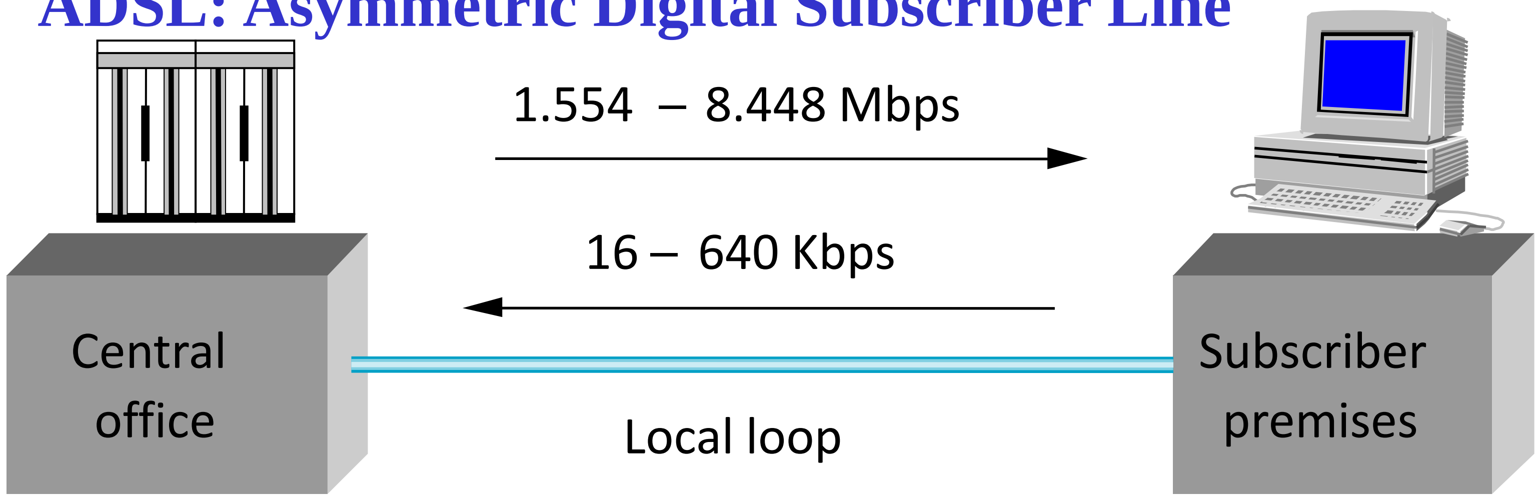
Local Links, Last-Mile Links

Cable	Typical Bandwidth	Distances
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

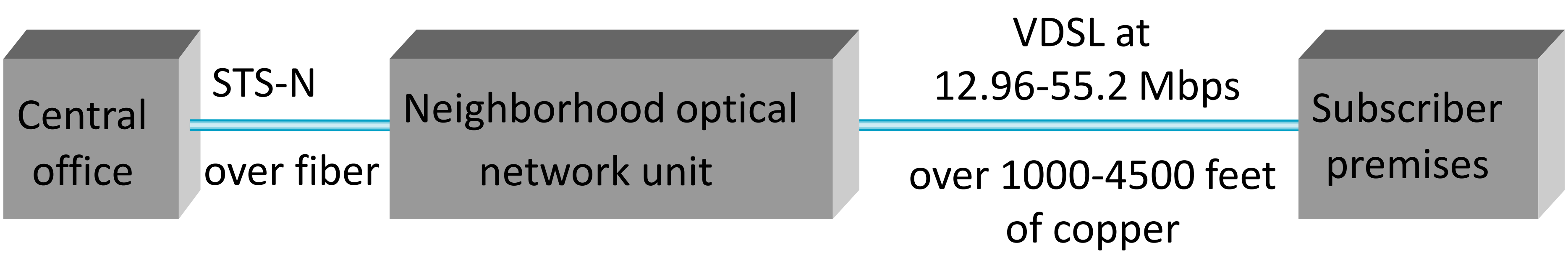
Service	Bandwidth
POTS	28.8 – 56 kbps
ISDN	64 – 128 kbps
xDSL	16 kbps – 55.2 Mbps
CATV	20 – 40 Mbps

ADSL & VDSL

- **ADSL: Asymmetric Digital Subscriber Line**



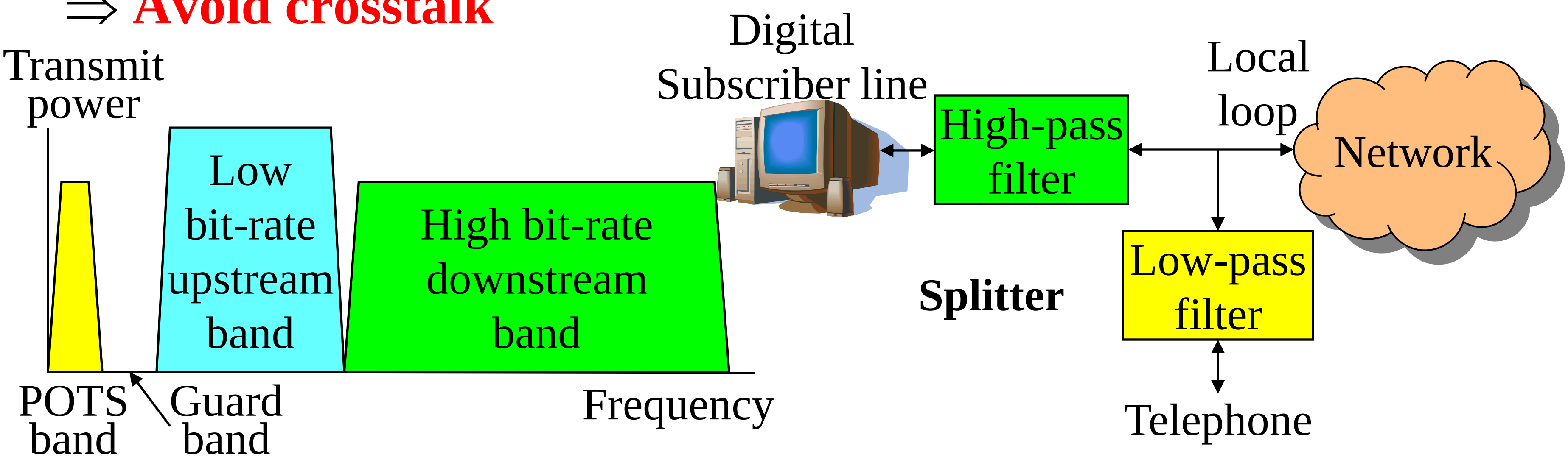
- **VDSL: Very high data rate Digital Subscriber Line**



Asymmetric Digital Subscriber Lines

- ADSL is designed to **simultaneously** support three services on a single twisted-wire pair: **(using FDM)**
  - Data rate of downstream (to subscriber): up to **12 Mbps**
  - Data rate of upstream (from subscriber): up to **1 Mbps**
  - **Plain old telephone service (POTS)**
- Upstream and downstream are placed in different band

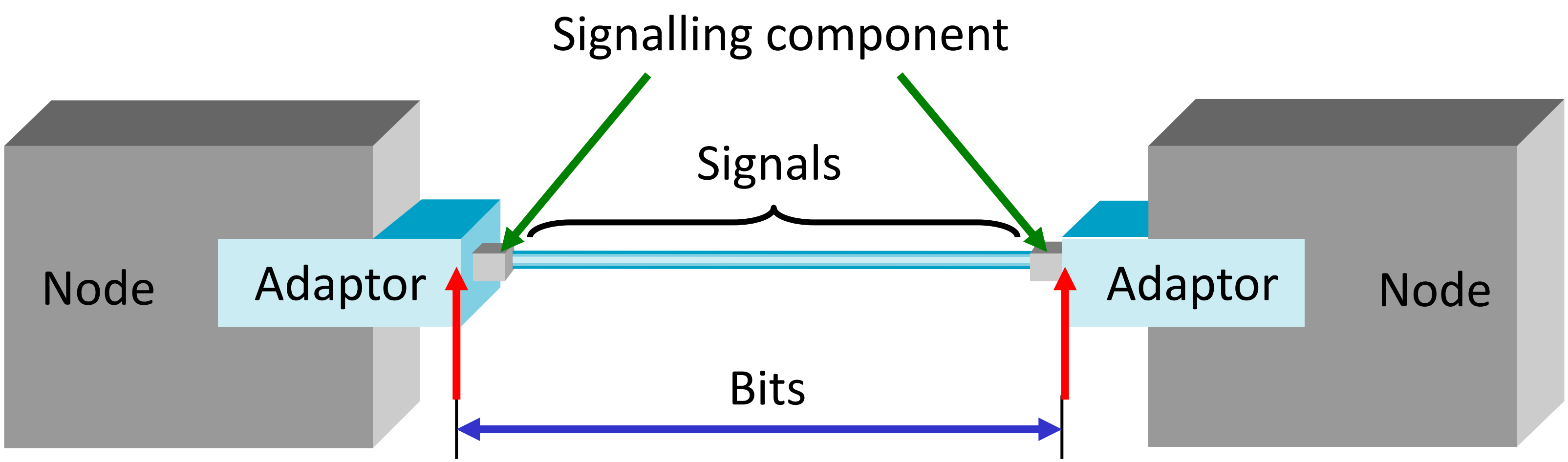
$\Rightarrow$ **Avoid crosstalk**



Encoding

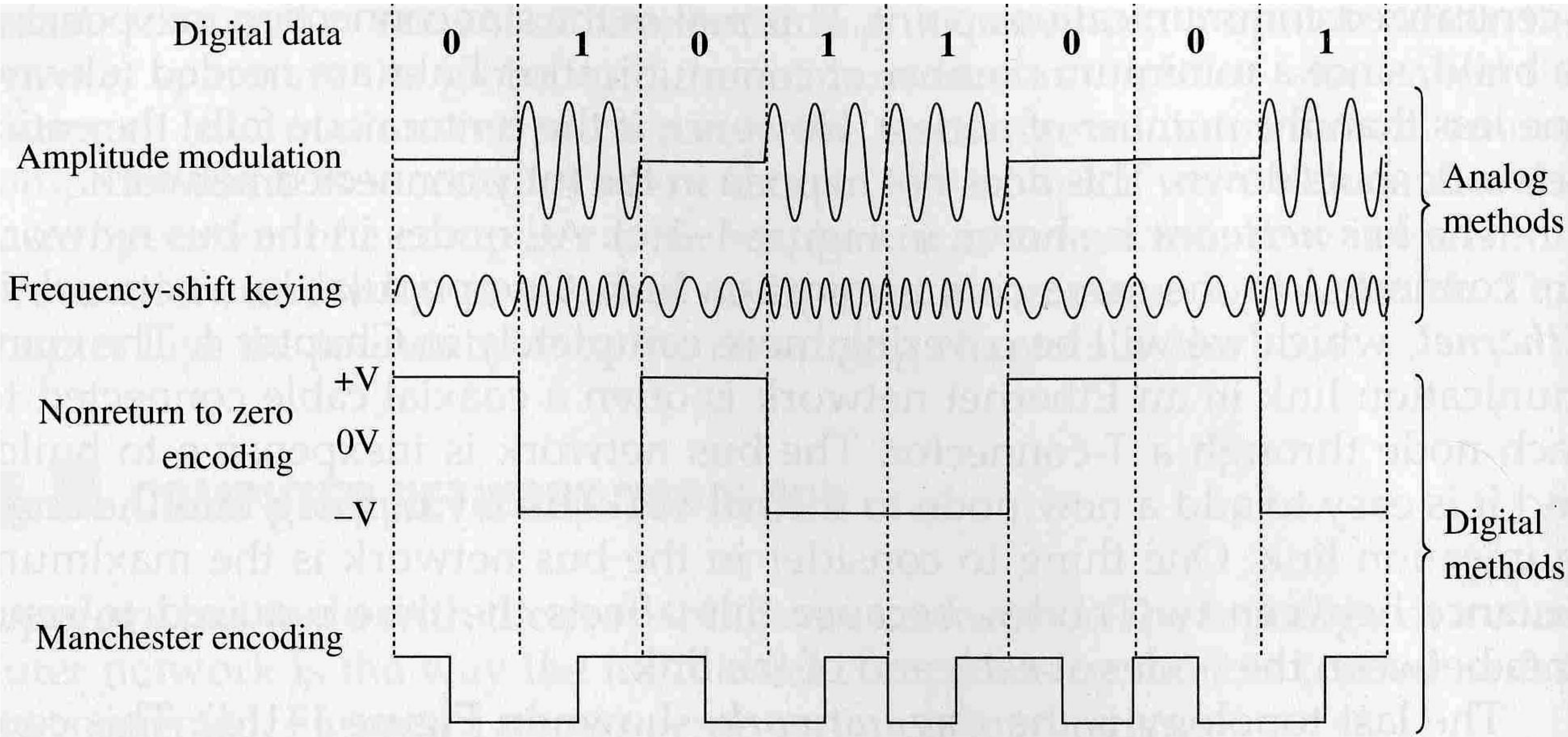
Encoding

- **Encoding:** map the data values onto signal (**Physical** layer)
- **Signals** travel over a link between two signaling components
- **Bits** flow between network adaptors



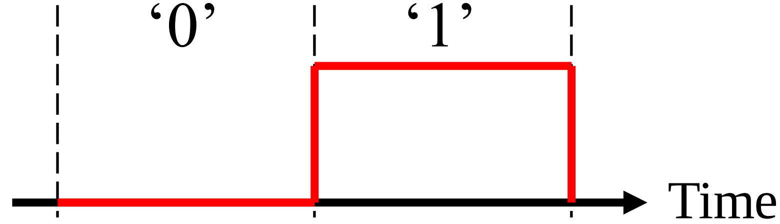
Methods of Representing Digital Data

- Physical layer signals

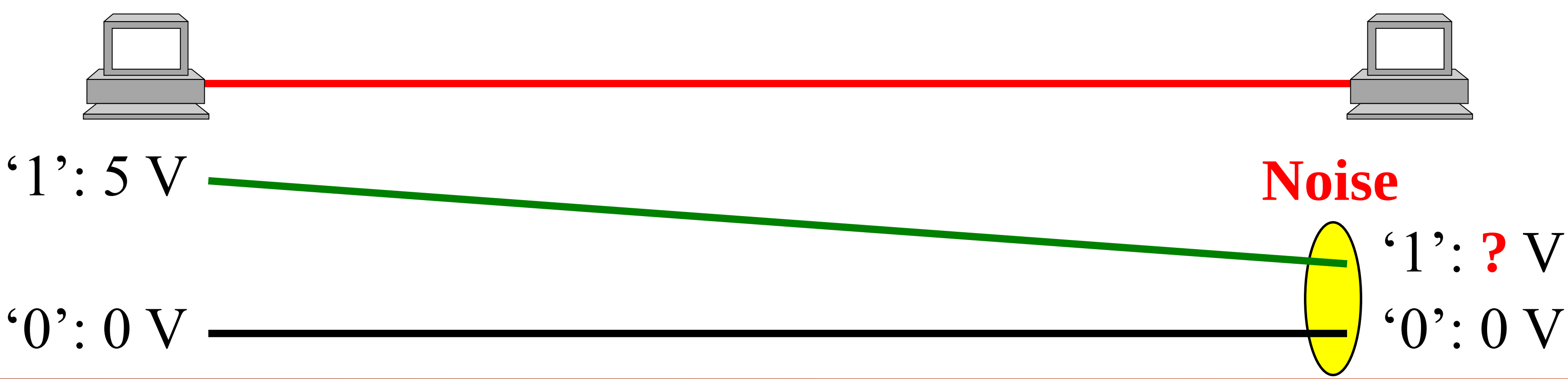


Non-Return to Zero

- **NRZ (Non-Return to Zero)**
  - Data value ‘1’: high signal
  - Data value ‘0’: low signal

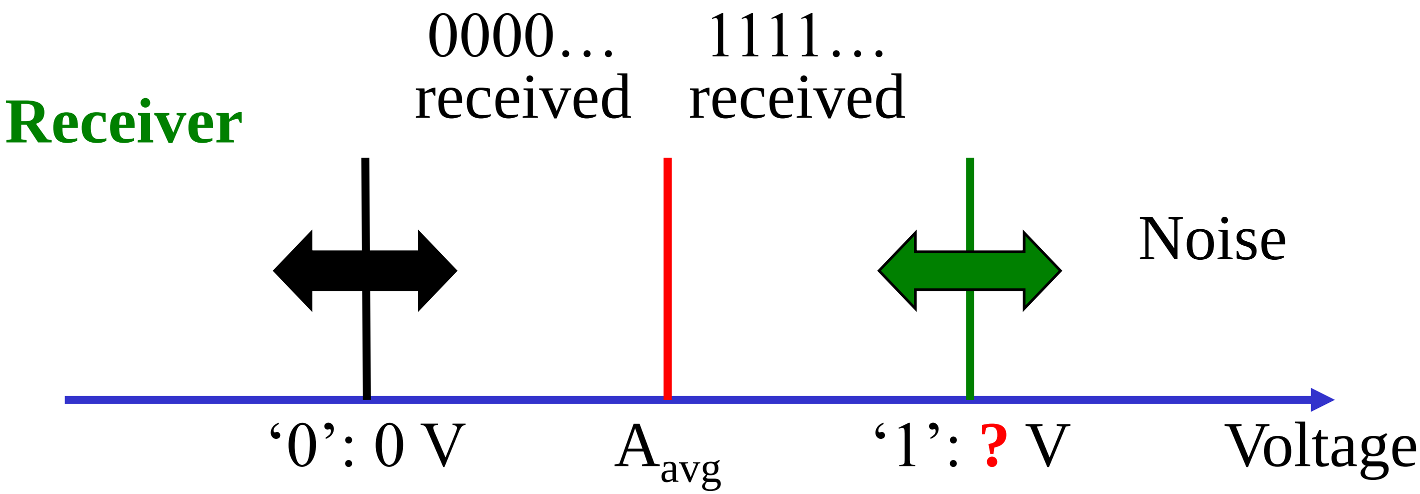


- Two problems are caused by **long strings** of ‘1’s or ‘0’s
  - **Baseline wander (Average amplitude level is variable)**
    - 1111… received: $A_{avg}\uparrow$; 0000… received: $A_{avg}\downarrow$
  - **Clock recovery (clock timing; bit interval?)**



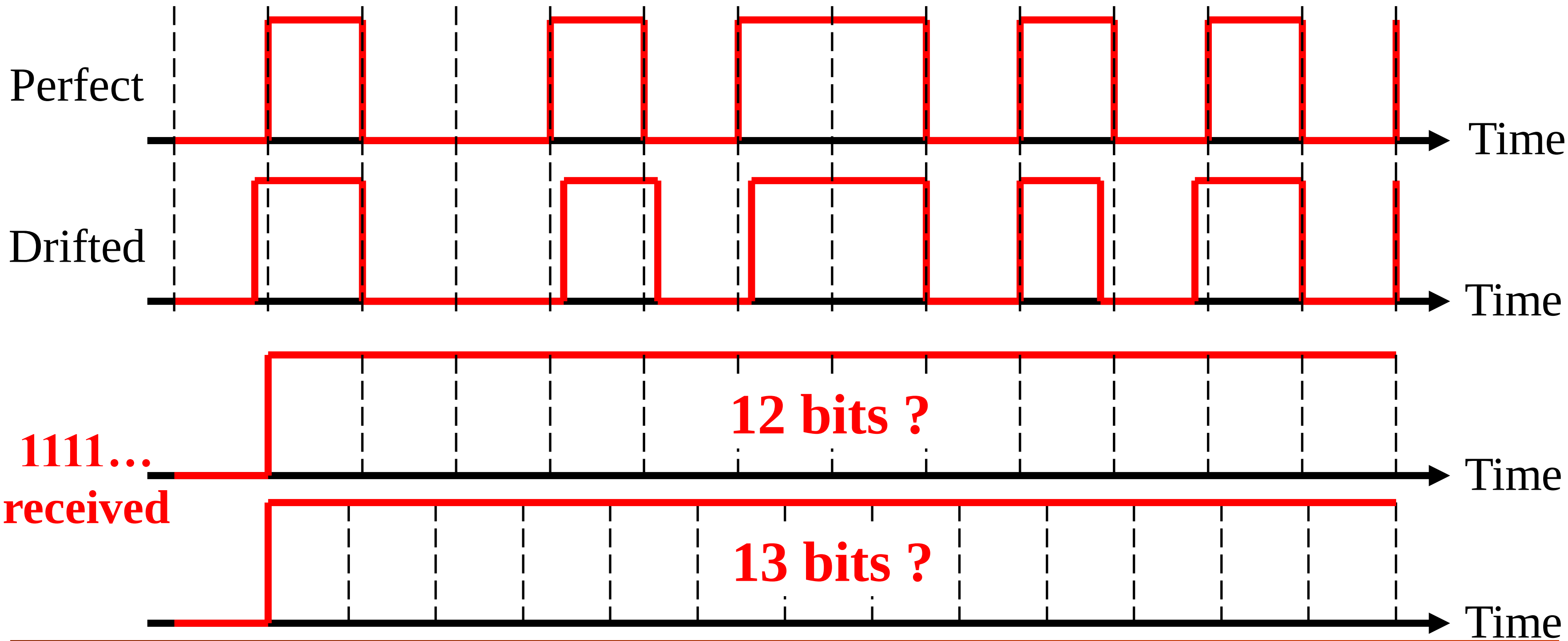
Non-Return to Zero

- To determine the value of a bit being '0' or '1'
  - If $A_r > A_{avg} \Rightarrow$ a '1' is received
  - If $A_r < A_{avg} \Rightarrow$ a '0' is received
- **Baseline wander problem**
  - 1111… received: $A_{avg} \uparrow$; 0000… received: $A_{avg} \downarrow$
  - Induce bit error



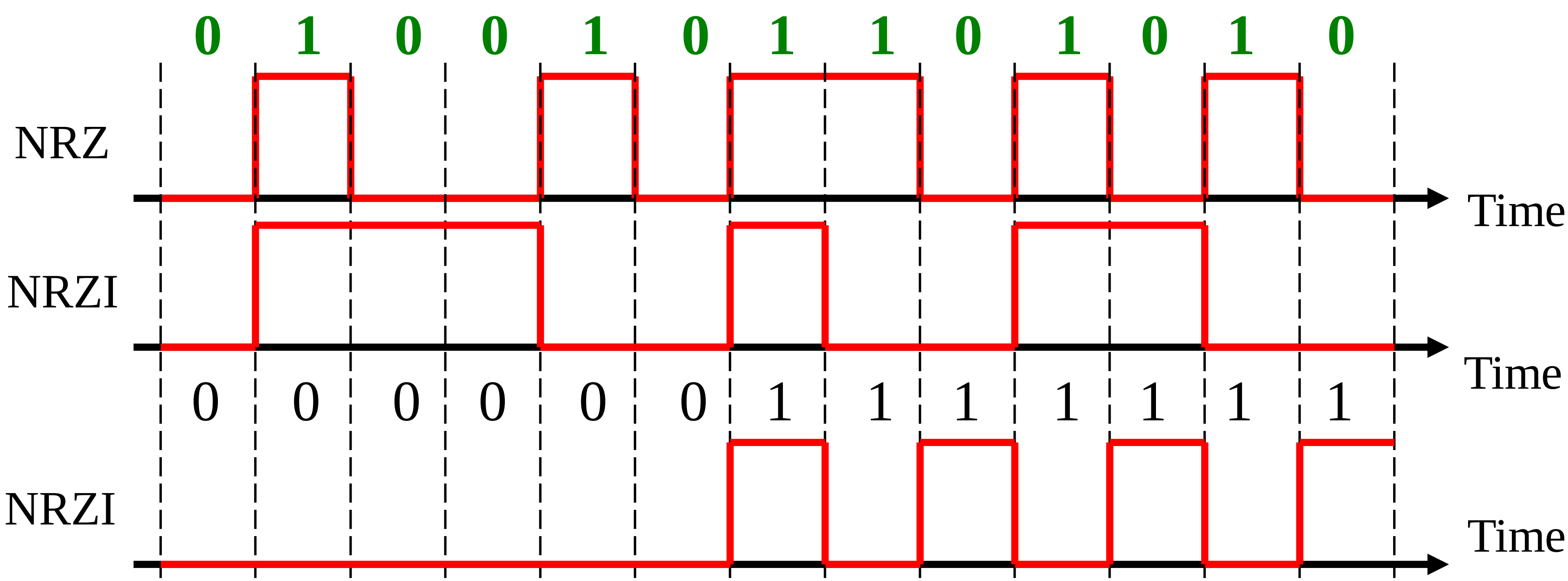
Non-Return to Zero

- The clock timing will **drift with time**
- The receiving clock is obtained according to the received signal
- **Clock recovery problem**



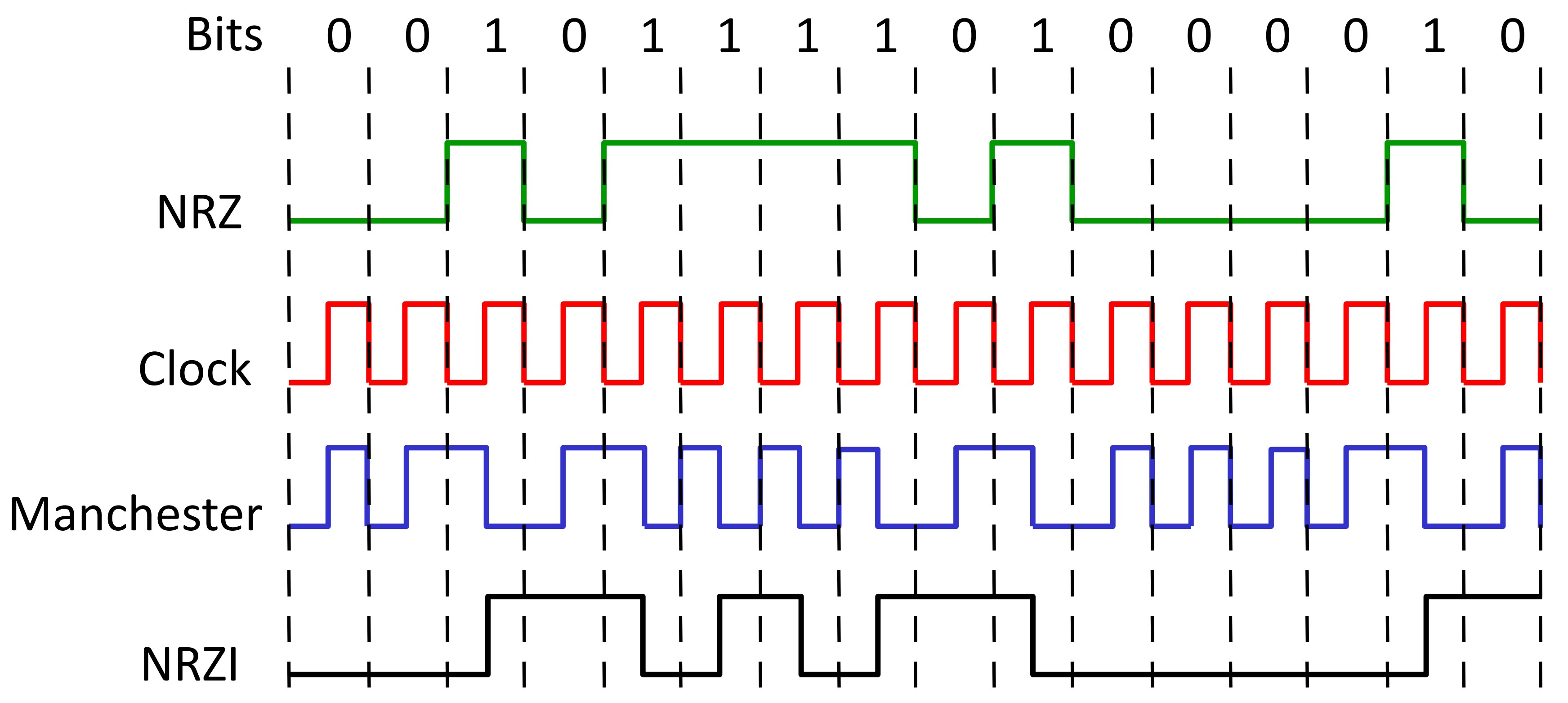
Non-Return to Zero Inverted

- **NRZI (Non-Return to Zero Inverted)**
  - Data value ‘1’: a transition from the current signal
  - Data value ‘0’: stay at the current signal
- NRZI solves the problem of consecutive ‘1’s, but **does nothing for consecutive ‘0’s**



Manchester

- **Manchester encoding**
  - Data value ‘1’: high signal to low signal
  - Data value ‘0’: low signal to high signal



4B/5B

- Manchester encoding suffers from the **transmission inefficiency** (**50% efficiency**)
- Rates for same bandwidth: NRZ & NRZI: *R*; Manchester: *R*/2
- **4B/5B:** inserts extra bits into the bit stream for NRZI
  - **80% efficiency**

No '00000' or '11111'

4-Bit Data Symbol	5-Bit Code	4-Bit Data Symbol	5-Bit Code
0000	11110	1000	10010
0001	01001	1001	10011
0010	10100	1010	10110
0011	10101	1011	10111
0100	01010	1100	11010
0101	01011	1101	11011
0110	01110	1110	11100
0111	01111	1111	11101

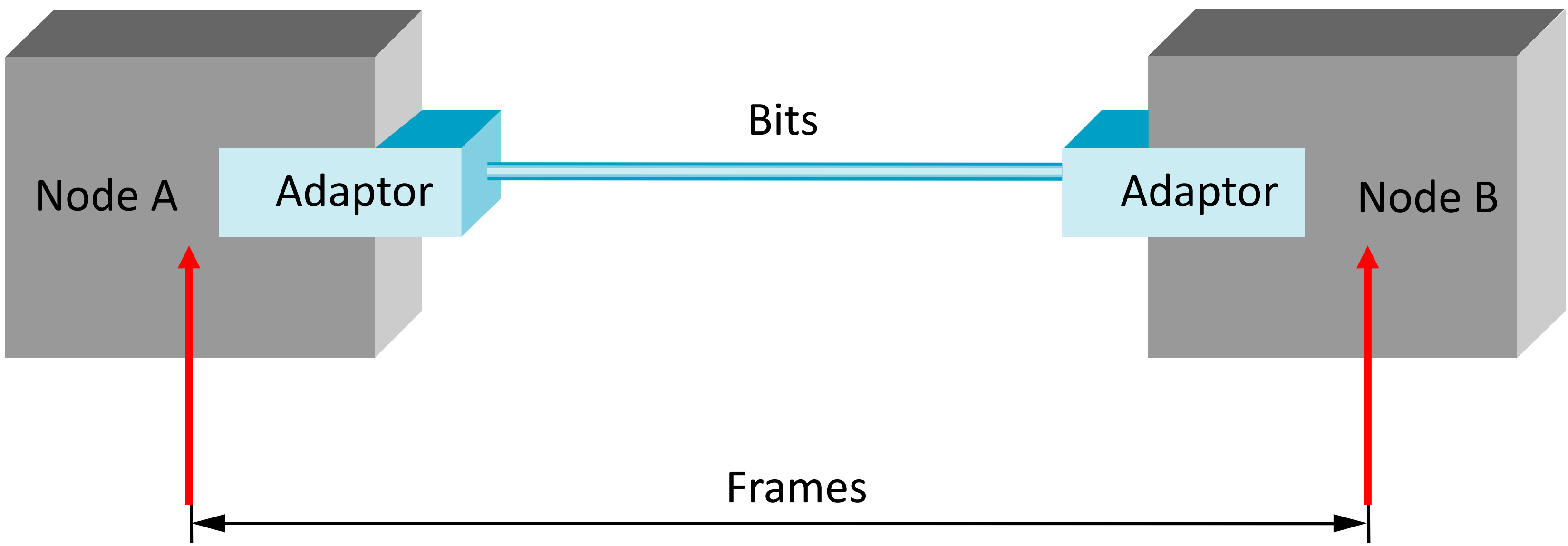
Encodings (cont)

- 4B/5B
  - every 4 bits of data encoded in a 5-bit code
  - 5-bit codes selected to have no more than one leading 0 and no more than two trailing 0s
  - thus, never get more than three consecutive 0s
  - resulting 5-bit codes are transmitted using NRZI
  - achieves 80% efficiency

Framing

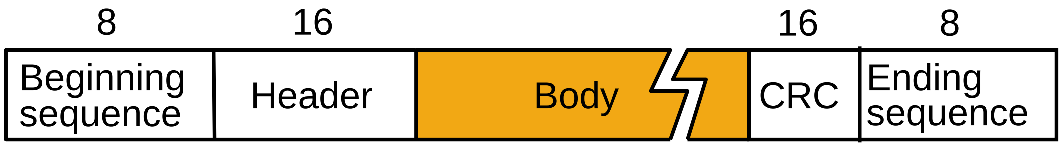
Framing

- A frame is a sequence of bits transmitted over a point-to-point link – from adaptor to adaptor (**Data link** layer)
  - **Frames** flow between nodes
- Recognize exactly what set of bits constitutes a frame
  - Determine where the frame **begins and ends**



Approaches

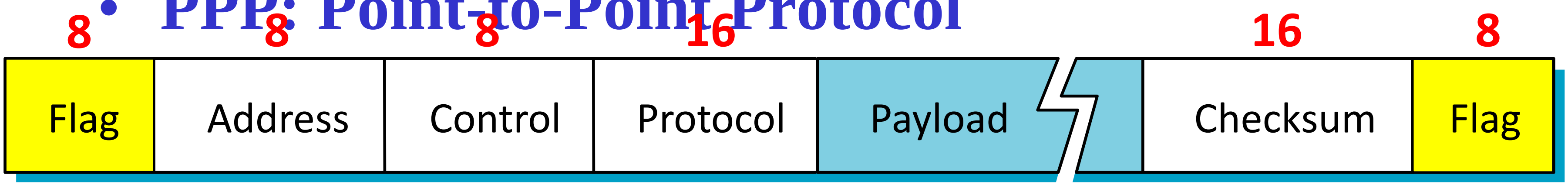
- Sentinel-based
  - delineate frame with special pattern: 01111110
  - e.g., HDLC, SDLC, PPP



  - problem: special pattern appears in the payload
  - solution: *bit stuffing*
    - sender: insert 0 after five consecutive 1s
    - receiver: delete 0 that follows five consecutive 1s

Byte-Oriented Protocols

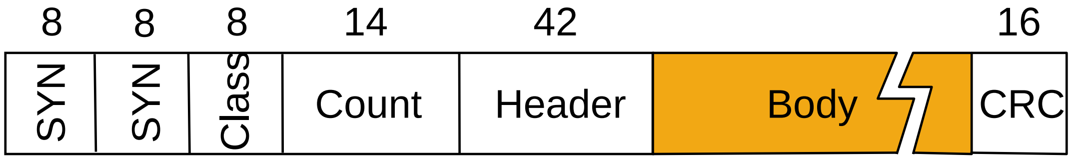
- **PPP: Point-to-Point Protocol**



  - **Flag field: 01111110** (sentinel character)
  - **Address and Control fields:** default values
  - **Protocol field:** used for high-level protocol
  - **Payload field:** can be negotiated, default: 1500 bytes
  - **Checksum field:** CRC

Approaches (cont)

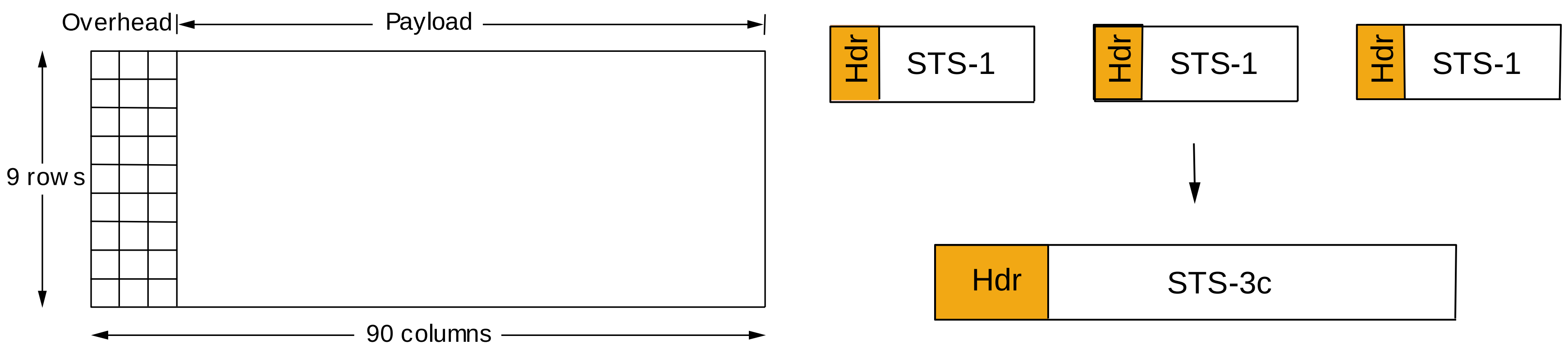
- Counter-based
  - include payload length in header
  - e.g., DDCMP



  - problem: count field corrupted
  - solution: catch when CRC fails

Approaches (cont)

- Clock-based
  - each frame is 125us long
  - e.g., SONET: Synchronous Optical Network
  - STS-*n* (STS-1 = 51.84 Mbps)



Clock-Based Framing Protocols

- **SONET (Synchronous Optical Network)**
  - First proposed by Bellcore
  - Developed under the American National Standards Institute (ANSI) for digital transmission over **optical fiber**
  - Adopted by the ITU-T
  - Some special information is offered to represent the frame starts and ends
  - **No bit stuffing is used**
  - A frame's length does not depend on the data being sent

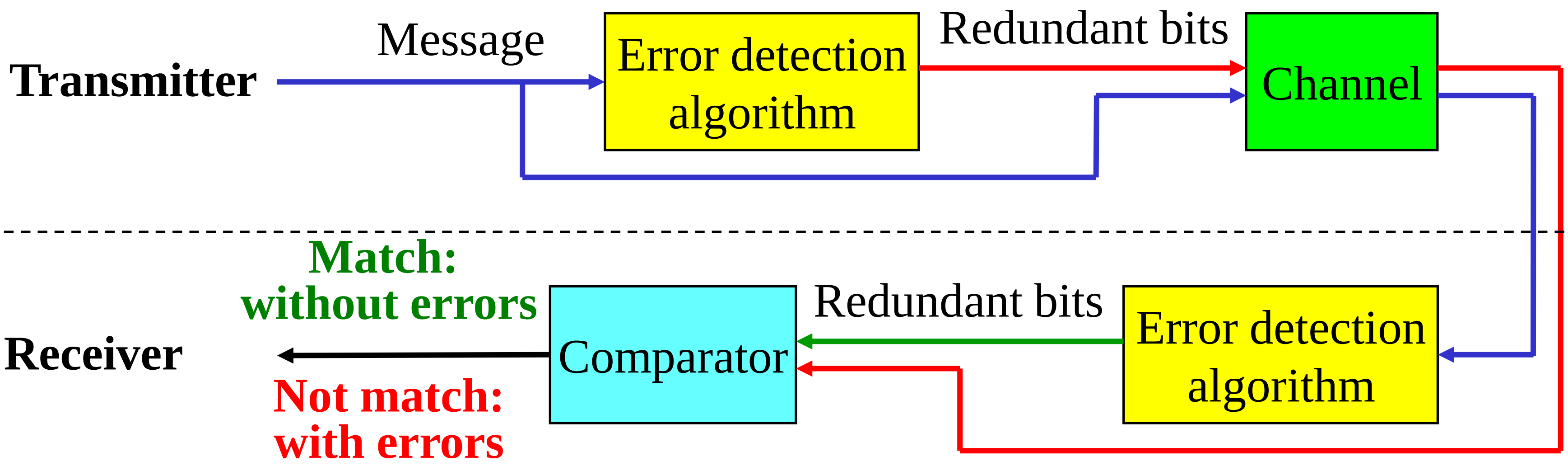
Error Detection

Error Detection

- Bit errors may be introduced into frames:
  - electrical interference
  - thermal noise
- Error detection mechanism adds some **redundant information** (redundant bits) to a frame
  - Redundant bits are used to determine if errors have been introduced
- A major goal in designing error detection algorithms is to
  - **Maximize** the probability of detecting errors using only a **small number** of redundant bits

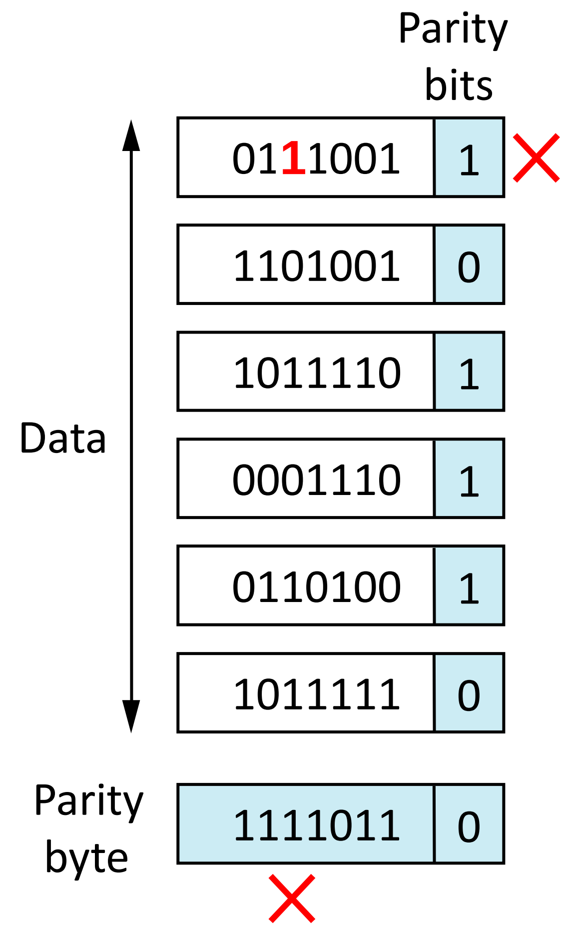
Error Detection

- **Error detection:**
  - Send ***k*** redundant bits for an ***n***-bit message, ***k*** **<<** ***n***
- For example:
  - A frame carries 12,000 bits of data (n = 12000)
  - A 32-bit CRC code is added (k = 32)
- Error detection:



Two-Dimensional Parity

- **Parity:**
  - Balance the number of '1's in a block
- **Add one extra bit** to a 7-bit code to balance the number of '1's in a byte
- Two-dimensional parity can catch all 1-, 2, and 3-bit errors, and most 4-bit errors
- Add k = 14 redundant bits for an n = 42-bit message



Internet Checksum

- **Checksum:**
  - Create the code based on **addition**
- Add up all the words that are transmitted and then **transmit the result of the sum**
  - Using **ones complement arithmetic**
- 16-bit ones complement arithmetic
  - A negative integer $-x$ is represented as the complement of x
  - +3: 0011 $\Rightarrow$ –3: 1100; +5: 0101 $\Rightarrow$ –5: 1010;
  - A **carryout** from the MSB needs to be added to the result
  - (–3)+(–5) = 0110+1 = 0111 = (–8); +8: 1000

Cyclic Redundancy Check

- Add k bits of redundant data to an n-bit message
  - want $k << n$
  - e.g., $k = 32$ and $n = 12{,}000$ (1500 bytes)
- Represent n-bit message as n-1 degree polynomial
  - e.g., MSG=10011010 as $M(x) = x^7 + x^4 + x^3 + x^1$
- Let k be the degree of some divisor polynomial
  - e.g., $C(x) = x^3 + x^2 + 1$

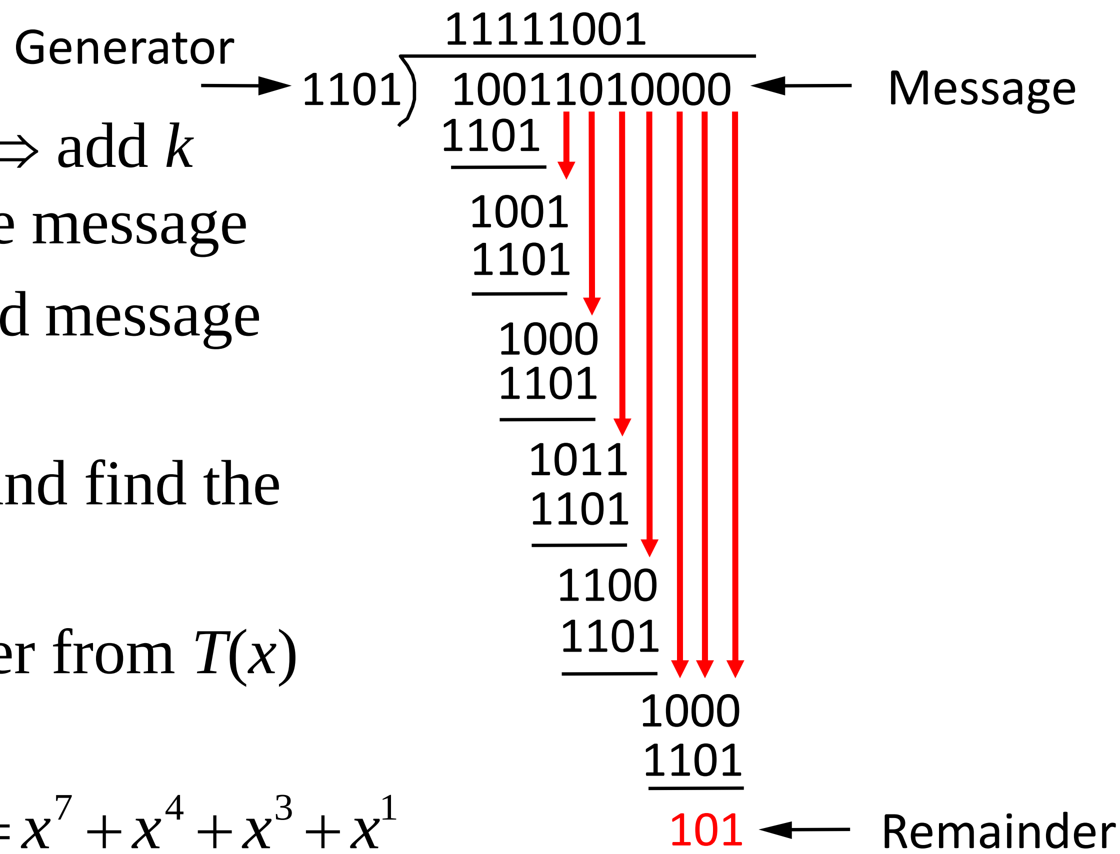
CRC (cont)

- Transmit polynomial $P(x)$ that is evenly divisible by $C(x)$
  - shift left k bits, i.e., $M(x)x^k$
  - subtract remainder of $M(x)x^k / C(x)$ from $M(x)x^k$
- Receiver polynomial $P(x) + E(x)$
  - $E(x) = 0$ implies no errors
- Divide $(P(x) + E(x))$ by $C(x)$; remainder zero if:
  - $E(x)$ was zero (no error), or
  - $E(x)$ is exactly divisible by $C(x)$

Cyclic Redundancy Check

- An **(*n*+1)-bit** message can be represented by a polynomial
  - $10011010 \Rightarrow M(x) = x^7 + x^4 + x^3 + x^1$
- The transmitter and receiver have to agree on a divisor polynomial $C(x)$ with degree k
- The $(n+1)$-bit message $M(x)$ plus k-bit redundant bits: $P(x)$
- If $P(x)$ is transmitted and there are **no errors** introduced
  - The receiver **can** divide $P(x)$ by $C(x)$ exactly, leaving a remainder of zero
- If $P(x)$ is transmitted and **some error** is introduced
  - The receiver **cannot** divide $P(x)$ by $C(x)$ exactly
- The subtraction is the **exclusive-OR (XOR)** operation

Cyclic Redundancy Check



- Multiply $M(x)$ by $x^k \Rightarrow$ add k zeros at the end of the message
  - This zero-extended message is called $T(x)$
- Divide $T(x)$ by $C(x)$ and find the remainder
- Subtract the remainder from $T(x)$ $\Rightarrow P(x)$
- For example: $M(x) = x^7 + x^4 + x^3 + x^1$

$$C(x) = x^3 + x^2 + 1$$

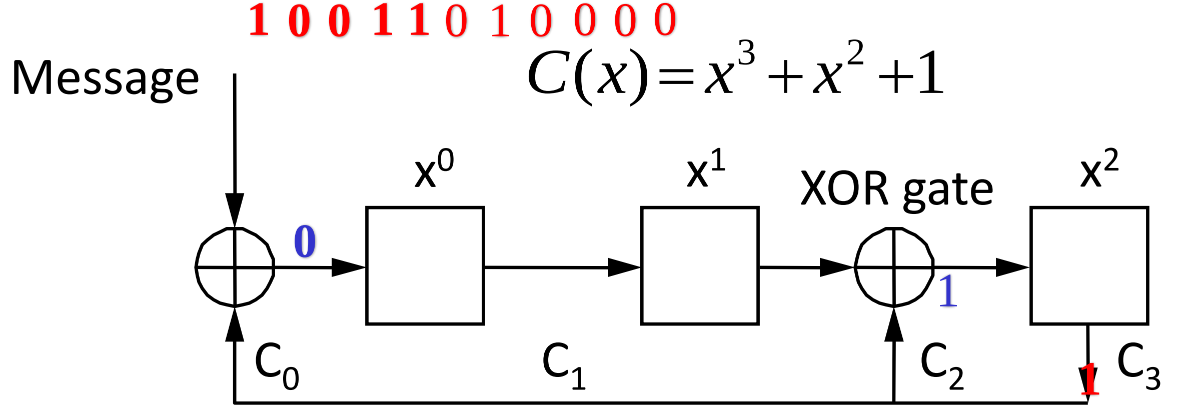
$$P(x) = x^{10} + x^7 + x^6 + x^4 + x^2 + 1 \Rightarrow 10011010101$$

Cyclic Redundancy Check

- Choose the polynomial $C(x)$:
  - If the transmitted message is $P(x)$ and the **error** is a polynomial $E(x) \Rightarrow$ the recipient sees $P(x) + E(x)$
  - If $E(x)$ can be divided by $C(x) \Rightarrow E(x)$ **cannot be detected**
  - Pick $C(x)$ so that this is very unlikely for common types of errors
- The CRC polynomial $C(x)$ has the following properties:
  - All single-bit errors: x^k and x^0 have nonzero coefficients
  - All single-bit errors: $C(x)$ has a factor with at least three terms
  - Any odd number of errors: $C(x)$ contains the factor $(x+1)$
  - Any burst errors with length less than k bits

Cyclic Redundancy Check

- CRC-8: $C(x)=x^8+x^2+x^1+1$
- CRC-10: $C(x)=x^{10}+x^9+x^5+x^4+x^1+1$
- CRC-12: $C(x)=x^{12}+x^{11}+x^3+x^2+1$
- CRC-16: $C(x)=x^{16}+x^{15}+x^2+1$
- CRC-CCITT: $C(x)=x^{16}+x^{12}+x^5+1$
- CRC-32: $C(x)=x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$



Mar 19, 2012

Brief Review Of Algebra

Correspond to Page 55,
Ch.2

Part 1: Algebra

or (+)	0	1
0	0	1
1	1	1

and (∘)	0	1
0	0	0
1	0	1

It is called Boolean algebra.(+ and ∘)

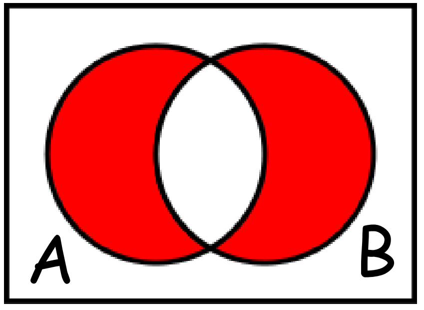
0 is additive unit element(1+___=1)

1 is multiplicative unit element(1*___=1)

GF(n) = Galois Field

- A finite field or Galois field is a field that contains a finite number of elements.
- Take GF(2) for example

$\oplus$	0	1
0	0	1
1	1	0



$A \oplus B$

$\otimes$	0	1
0	0	0
1	0	1

- 1 is an additive inverse of 1, because $1 \oplus 1 = 0$.

- Take GF(3) for example

$\oplus$	0	1	2
0	0	1	2
1	1	2	0
2	2	0	1

$\otimes$	0	1	2
0	0	0	0
1	0	1	2
2	0	2	1

$a \oplus b = (a+b) \bmod 3$ $a \otimes b = (a*b) \bmod 3$

mod(modulo operation): the remainder after division

- 2 is an additive inverse of 1, because $2 \oplus 1 = 0$.
- 2 is a multiplicative inverse of 2, because $2 \otimes 2 = 1$.

Part 2 : Matrix

- The usual extension to matrix operations : addition(⊕), multiplication(⊗)

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \oplus \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} = \begin{pmatrix} a_{11} \oplus b_{11} & a_{12} \oplus b_{12} \\ a_{21} \oplus b_{21} & a_{22} \oplus b_{22} \end{pmatrix}$$

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \otimes \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} = \begin{pmatrix} (a_{11} \otimes b_{11}) \oplus (a_{12} \otimes b_{21}) & (a_{11} \otimes b_{12}) \oplus (a_{12} \otimes b_{22}) \\ (a_{21} \otimes b_{11}) \oplus (a_{22} \otimes b_{21}) & (a_{21} \otimes b_{12}) \oplus (a_{22} \otimes b_{22}) \end{pmatrix}$$

Part 3 : Polynomial

- $x^2 = x \otimes x$ (commutative)

$$(a \otimes x \oplus b) \otimes (c \otimes x \oplus d)$$
$$= (a \otimes c) \otimes x^2 \oplus ((a \otimes d) \oplus (b \otimes c)) \otimes x \oplus (b \otimes d)$$

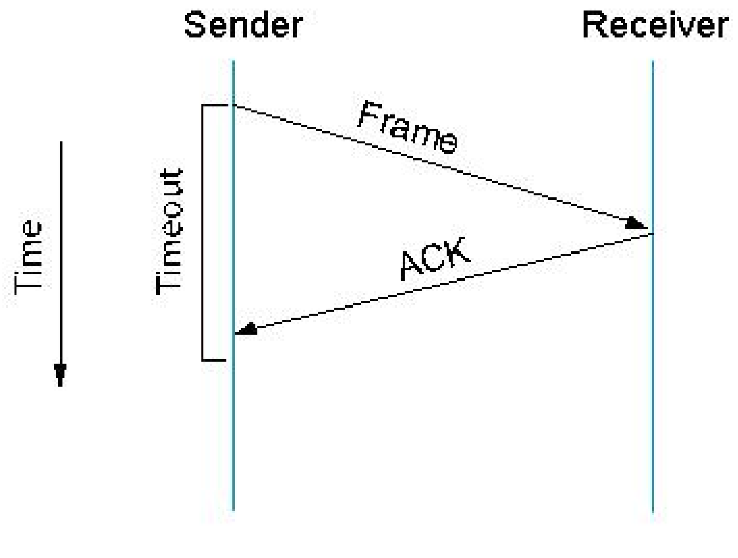
Reliable Transmission

Reliable Transmission

- Some corrupt frames must be **discarded** in the receiver
  - A link-level protocol must recover the discarded frames
- Use a combination of two fundamental mechanisms
  - **Acknowledgment** (ACK)
  - **Timeout**
- Acknowledgment: a small **control frame** that is sent back to its peer saying that it **has received** an earlier frame
  - Can **piggyback** an ACK on a data frame in the **opposite direction**
- Timeout: if the sender does not receive an ACK after a reasonable amount of time $\Rightarrow$ **Retransmit** the original frame
- This is known as **ARQ (Automatic Repeat reQuest)**

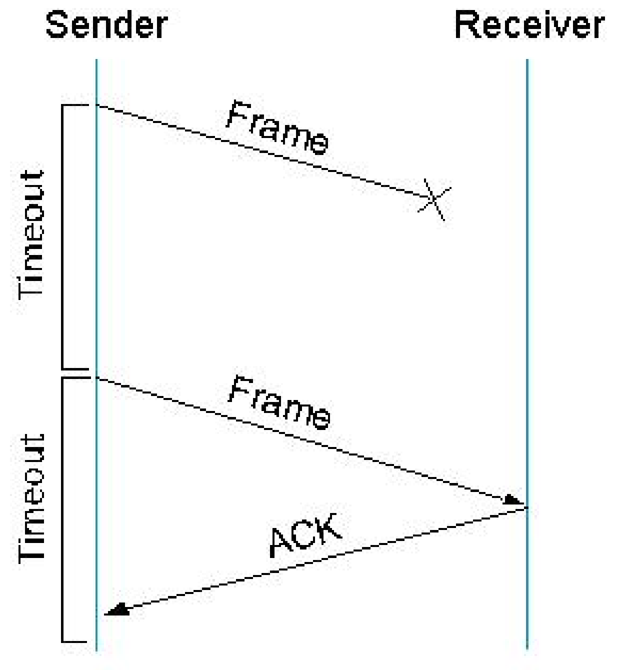
Stop-and-Wait

- The simplest ARQ scheme
- After transmitting one frame, the sender **waits** for an ACK before transmitting the next frame
- If timeout occurs, the sender retransmits the original frame



Normal

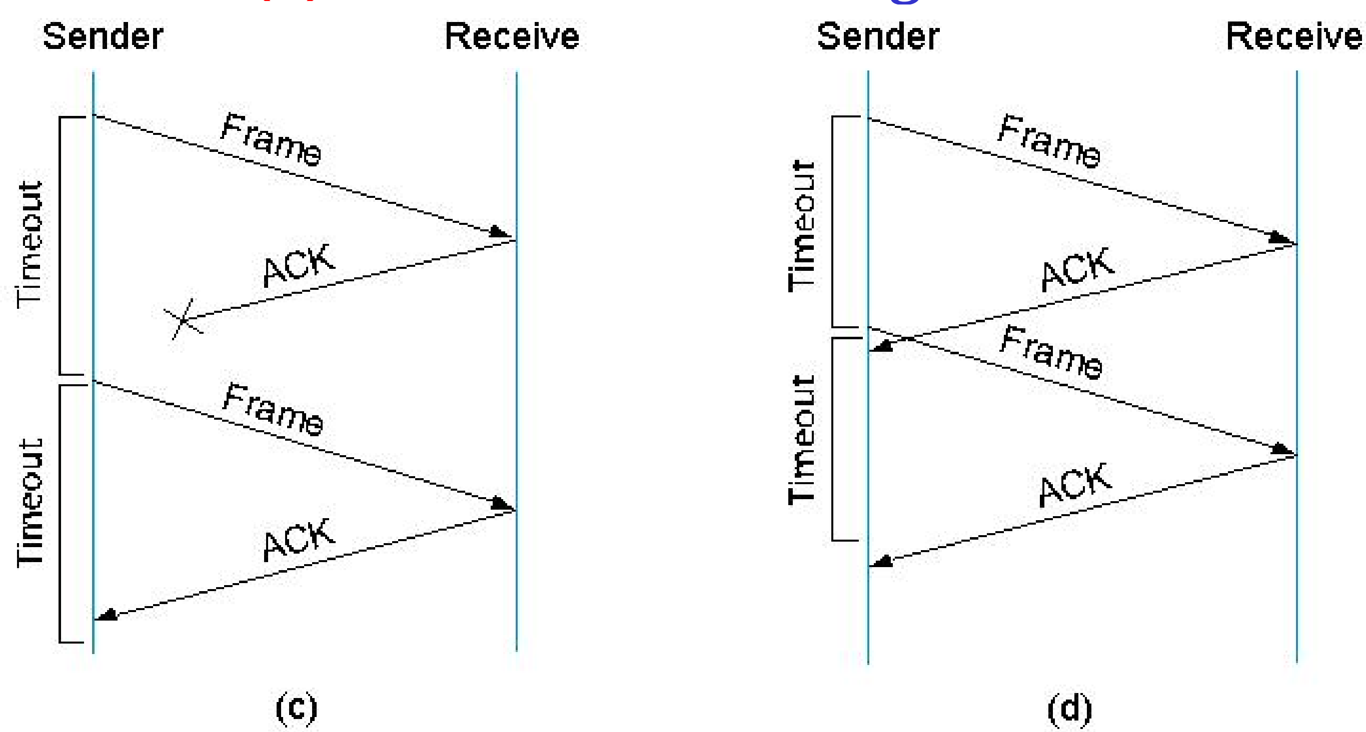
(a)



(b)

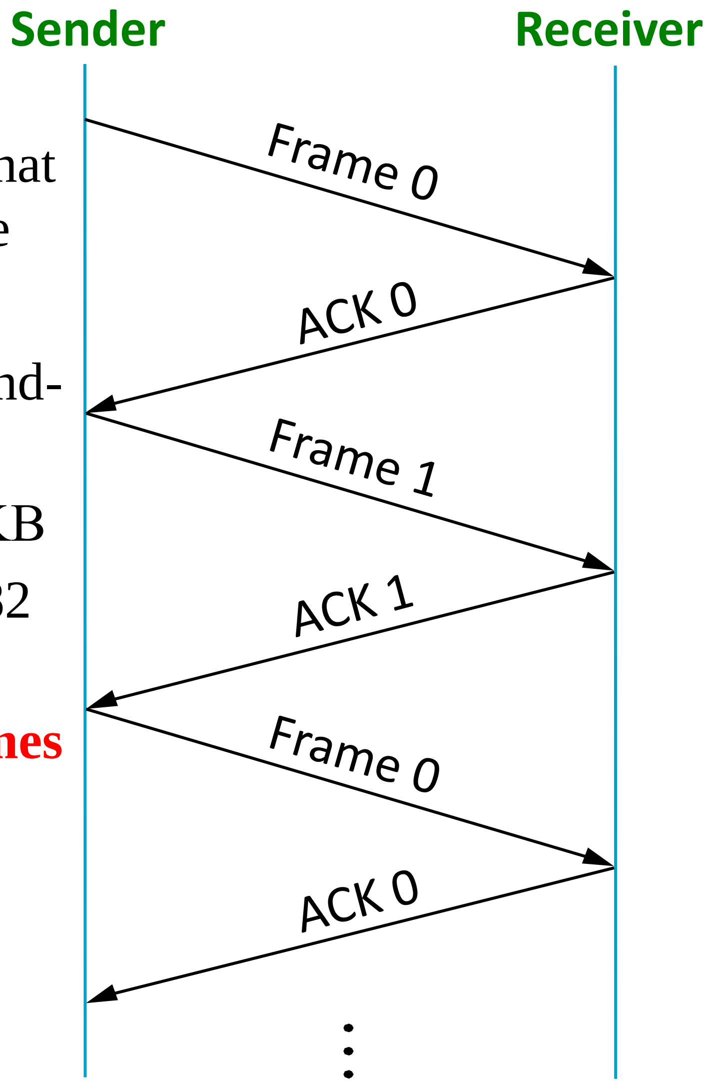
Stop-and-Wait

- (c) the ACK loses
- (d) the timeout fires too soon
- The receiver will think the retransmitted frame being the **next frame (?)** ⇒ **Frame numbering** is essential



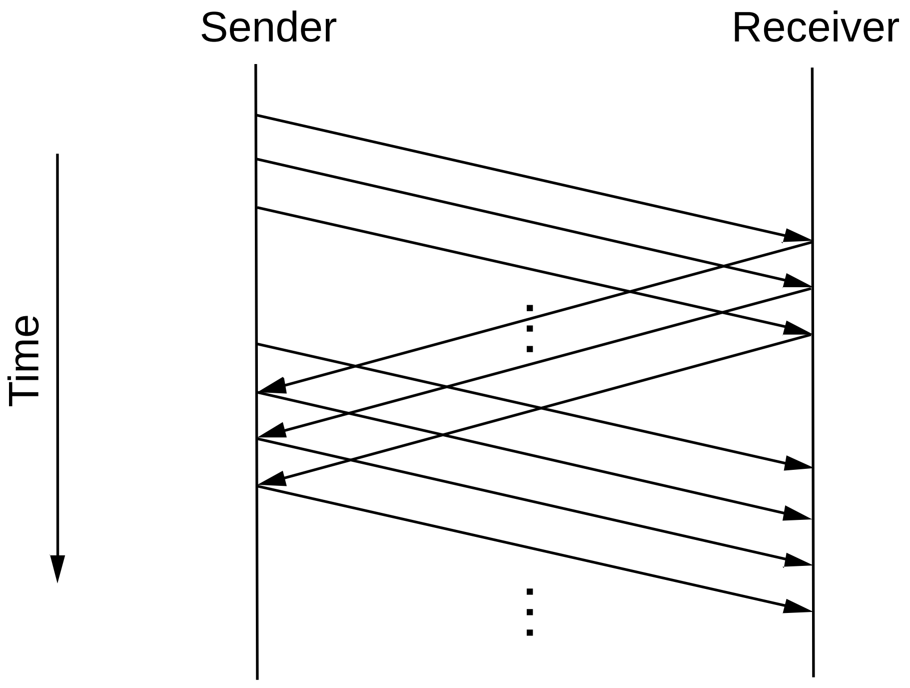
Stop-and-Wait

- Frame numbering: **1-bit** sequence number
- The main shortcoming of the stop-and-wait is that only one outstanding frame on the link at a time
  - **Bad bandwidth efficiency (fill the pipe)**
- For example: a 1.5 Mbps link with a 45 ms round-trip time
  - delay × bandwidth product = 67.5 kb ≈ 8 KB
  - frame size = 1 KB ⇒ 1024 × 8 / 0.045 = 182 kbps **(actual average data rate)**
  - The link is able to transmit up to **eight frames**



Sliding Window

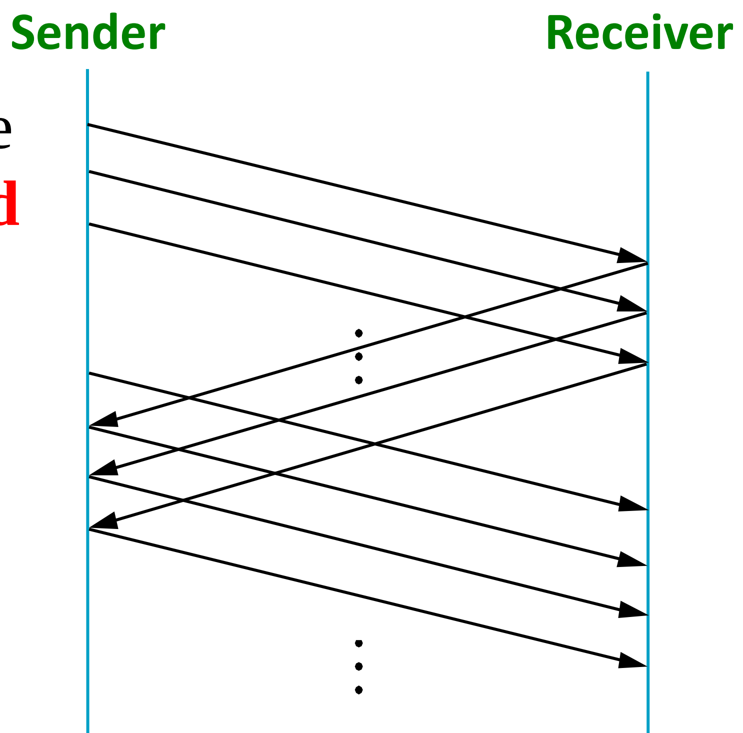
- Allow multiple outstanding (un-ACKed) frames
- Upper bound on un-ACKed frames, called *window*



Sliding Window (Sender)

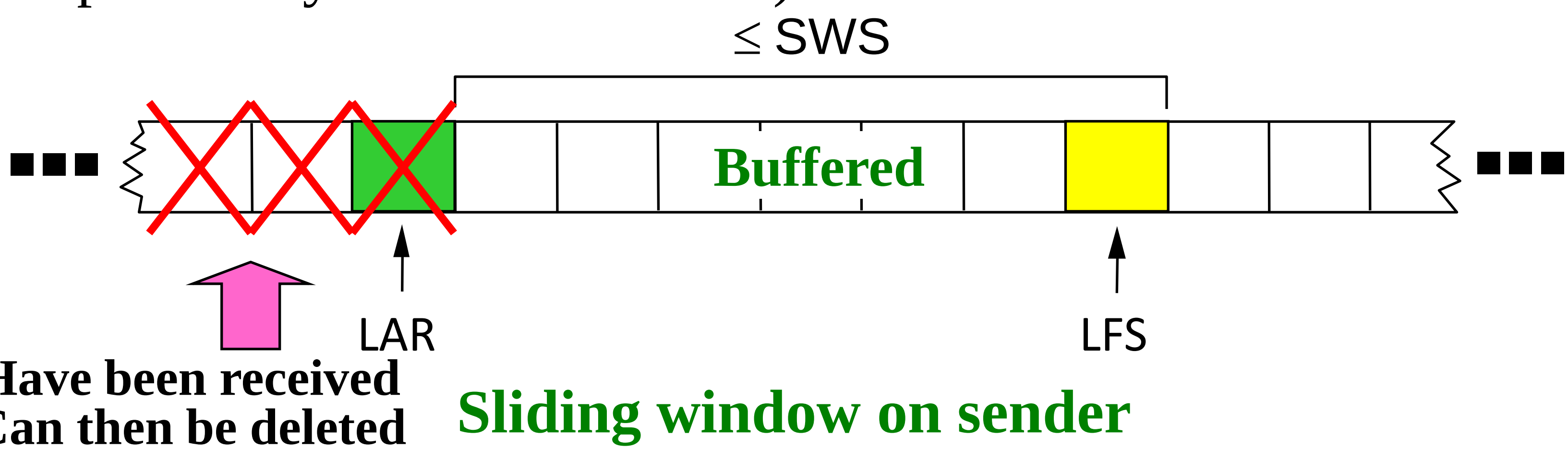
- The sender assigns a sequence number, denoted as **SeqNum**, to each frame
- The **sender** maintains three variables:
  - **SWS** (send window size): the **upper bound** on the number of outstanding frames
  - **LAR**: the sequence number of the **Last Acknowledgment Received**
  - **LFS**: the sequence number of the **Last Frame Sent**
- The **sender** maintains the following invariant:

LFS – LAR ≤ SWS



Sliding Window (Sender)

- When an ACK arrives, the sender moves LAR to the **right**
  - Allow the sender to transmit another frame
- The sender associates a timer with **each transmitted frame**
  - The sender retransmit the frame when the timer expires and no ACK is received
- The sender should **buffer** up to SWS frames (for the possibility of retransmission)

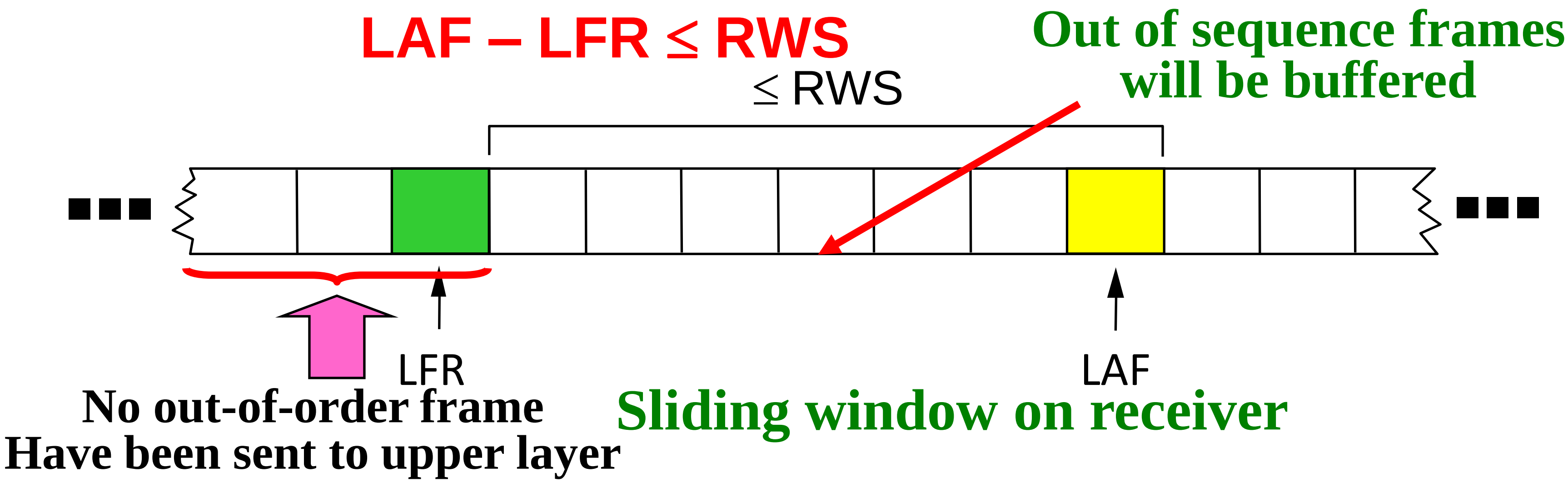


Sliding window on sender

Sliding Window (Receiver)

- The receiver maintains three variables:
  - **RWS** (receive window size): the **upper bound** on the number of **out-of-order** frames
  - **LAF**: the sequence number of the **Largest Acceptable Frame**
  - **LFR**: the sequence number of the **Last Frame Received**
- The receiver maintains the following invariant:

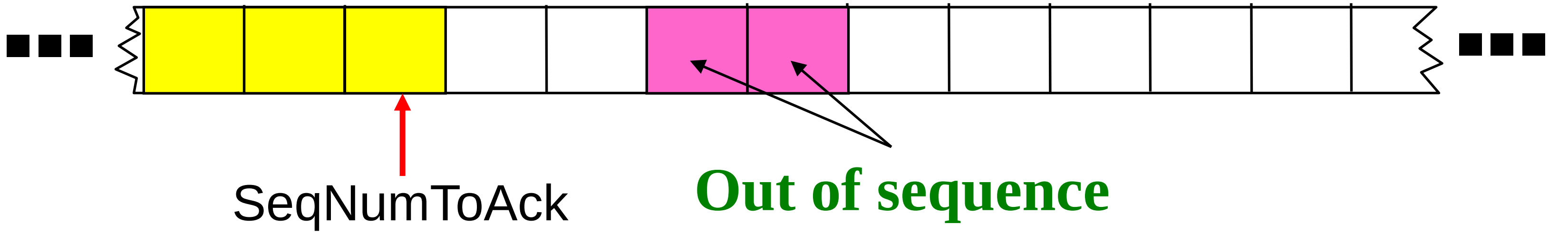
LAF – LFR ≤ RWS



Sliding window on receiver

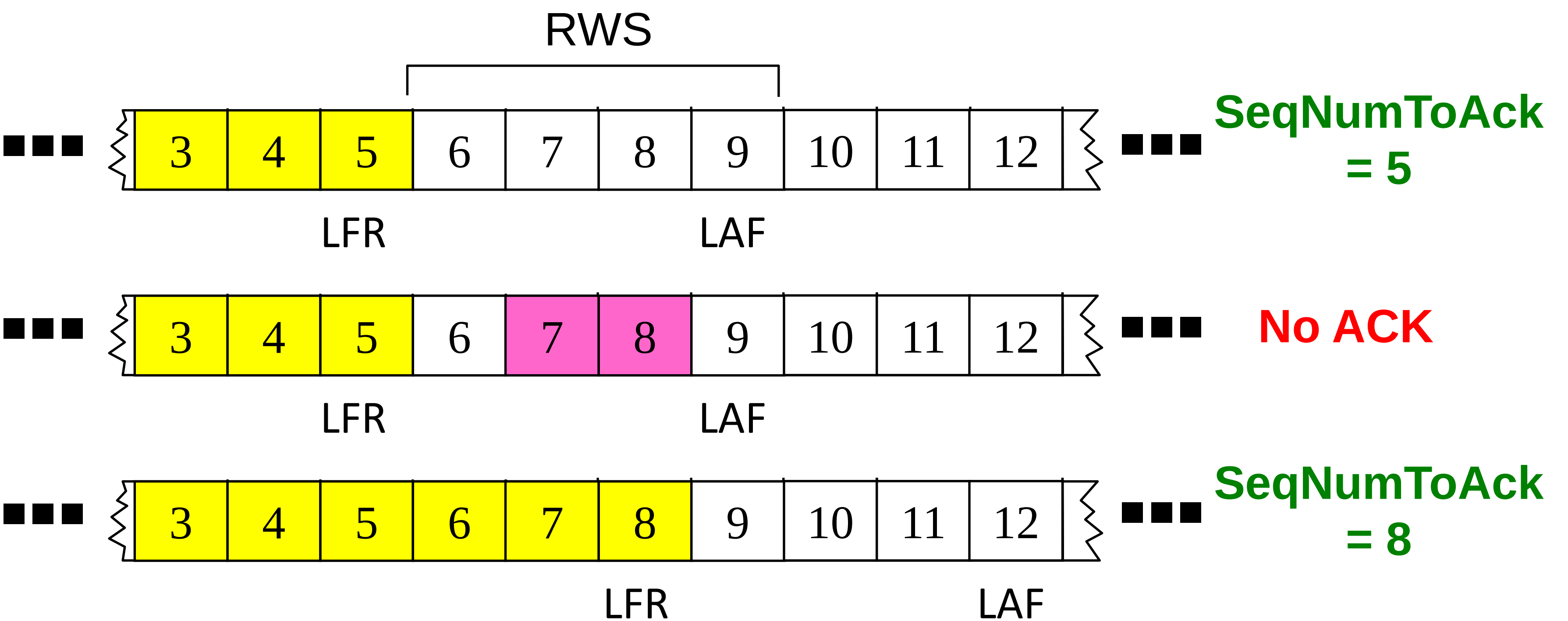
Sliding Window

- When a frame with sequence number **SeqNum** arrives
  - If **SeqNum ≤ LFR** or **SeqNum > LAF** ⇒ the frame is outside the receiver's window ⇒ **discarded**
  - If **LFR < SeqNum ≤ LAF** ⇒ the frame is within the receiver's window ⇒ **accepted**
- **SeqNumToAck**: the acknowledged sequence number
  - The **largest** sequence number not yet acknowledged, and
  - All frames with number ≤ SeqNumToAck have been received **(No out-of-order frame)**



Sliding Window

- Then the receiver sets **LFR = SeqNumToAck** and adjusts **LAF = LFR + RWS**
- For example:
  - LFR = 5, RWS = 4, and LAF = 9



Sliding Window

- **Early detection of packet losses:**
  - **Negative acknowledgment (NAK):** send a NAK for frame 6 as soon as frame 7 arrived, or
  - **Duplicate ACK:** resend an ACK for frame 5 when frame 7 arrived
- **Selective acknowledgments:** the receiver acknowledges those frames it has received
- Window size selection:
  - SWS: according to the delay × bandwidth product
  - RWS =1: the receiver will not buffer out of order frames
  - RWS = SWS: the receiver will buffer any frames
  - RWS > SWS: no more than SWS out of order frames

Sequence Number Space

- **`SeqNum`** field is finite; sequence numbers wrap around
- Sequence number space must be larger then number of outstanding frames
- **`SWS`** <= **`MaxSeqNum-1`** is not sufficient
  - suppose 3-bit **`SeqNum`** field (0..7)
  - **`SWS=RWS=7`**
  - sender transmit frames 0..6
  - arrive successfully, but ACKs lost
  - sender retransmits 0..6
  - receiver expecting 7, 0..5, but receives second incarnation of 0..5
- **`SWS`** < **`(MaxSeqNum+1)/2`** is correct rule
- Intuitively, **`SeqNum`** “slides” between two halves of sequence number space

Sequence Numbers

- The number of possible sequence numbers must be larger than the number of outstanding frames allowed
- If RWS = 1, MaxSeqNum ≥ SWS+1 is sufficient
- If RWS = SWS, SWS < (MaxSeqNum+1)/2
- **MaxSeqNum ≥ SWS+RWS**

Sequence Numbers (Example)

- Suppose that we run the sliding window algorithm with SWS = 5 and RWS = 3, and no out-of-order arrivals
- (a) Find the smallest value for MaxSeqNum.
- It must be satisfied that "When DATA[MaxSeqNum] is in the receive window, DATA[0] is out of the sender window"

  $\Rightarrow$ The data in RWS are DATA[MaxSeqNum-2], DATA[MaxSeqNum-1] and DATA[MaxSeqNum]

  $\Rightarrow$ DATA[MaxSeqNum-3] must have been delivered

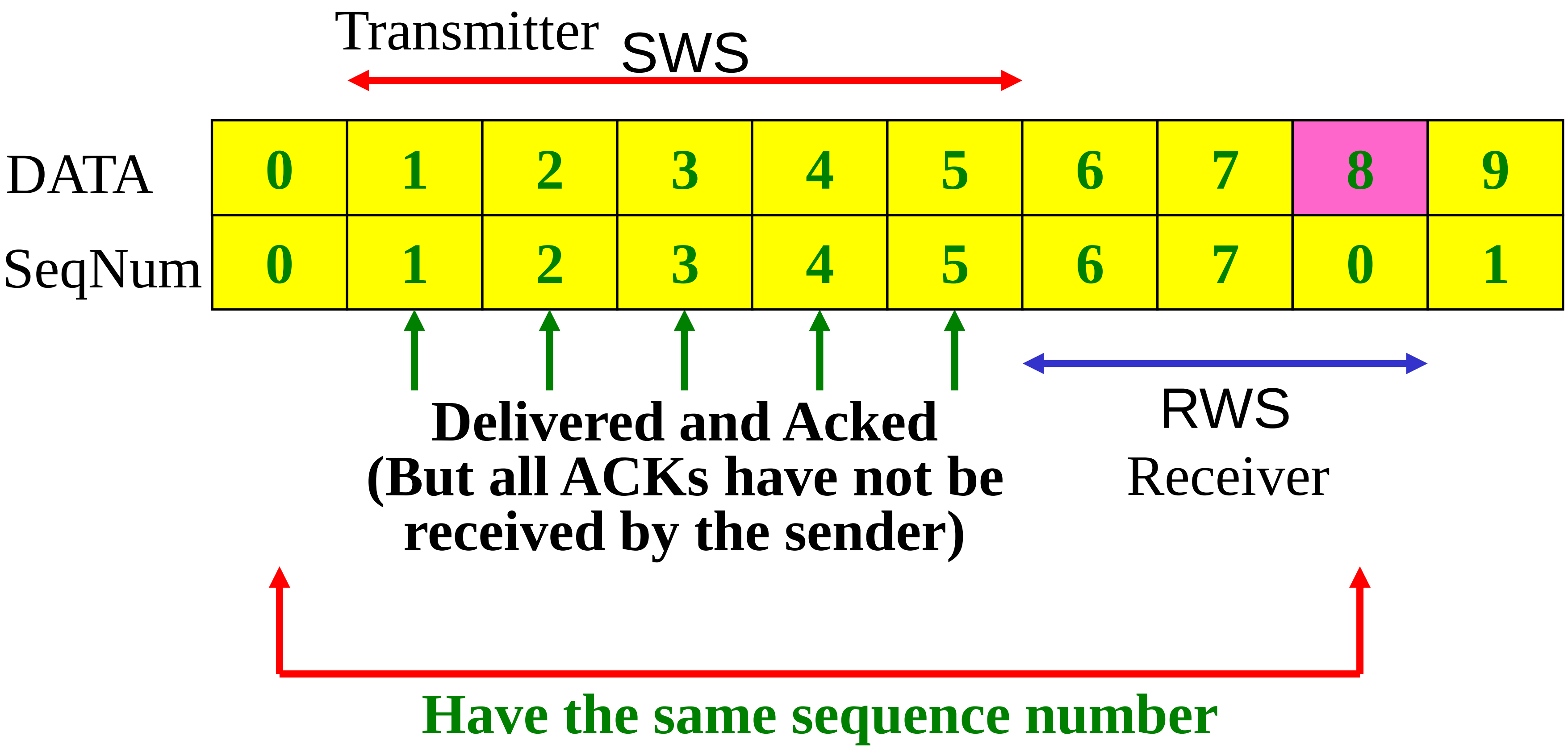
  $\Rightarrow$ The data in SWS are DATA[MaxSeqNum-7] … DATA[MaxSeqNum-3]

  $\Rightarrow$ DATA[0] cannot be in SWS, otherwise DATA[0] and DATA[MaxSeqNum] have the same sequence number **0**

  $\Rightarrow$ MaxSeqNum-7 > 0 $\Rightarrow$ **MaxSeqNum ≥ 8**

Sequence Numbers (Example)

- If **MaxSeqNum** = 8 = SWS + RWS; SeqNum = 0, 1, …, 7



Sequence Numbers (Example)

- (b) Given an example showing that MaxSeqNum is not sufficient.
  - MaxSeqNum = 7

  $\Rightarrow$ The data in RWS: DATA[5], DATA[6], DATA[7]

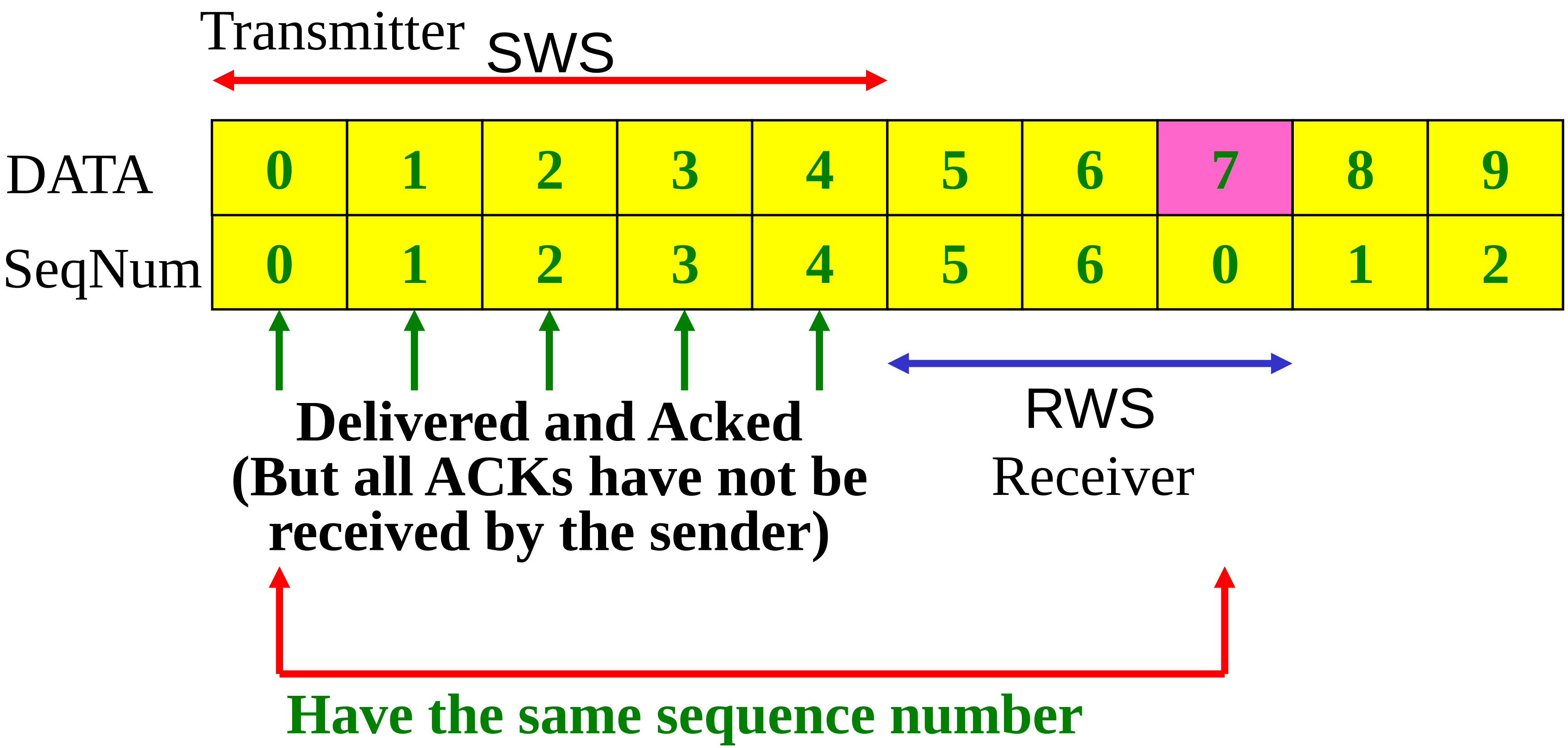
  $\Rightarrow$ The data in SWS: DATA[0], DATA[1], DATA[2], DATA[3], DATA[4]
  - If the arrival of the ACK to the sender is delayed

  $\Rightarrow$ The sender may retransmit DATA[0] after timeout

  $\Rightarrow$ The receiver may receive DATA[0] as DATA[7] (with the same sequence number)
- A general rule for the minimum MaxSeqNum
  - **MaxSeqNum $\geq$ SWS+RWS**

Sequence Numbers (Example)

- If **MaxSeqNum** = 7 < SWS + RWS; SeqNum = 0, 1, …, 6



Concurrent Logical Channels

- Multiplex 8 logical channels over a single link
- Run stop-and-wait on each logical channel
- Maintain three state bits per channel
  - channel busy
  - current sequence number out
  - next sequence number in
- Header: 3-bit channel num, 1-bit sequence num
  - 4-bits total
  - same as sliding window protocol
- Separates *reliability* from *order*

Random Access Protocols

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

Slotted ALOHA

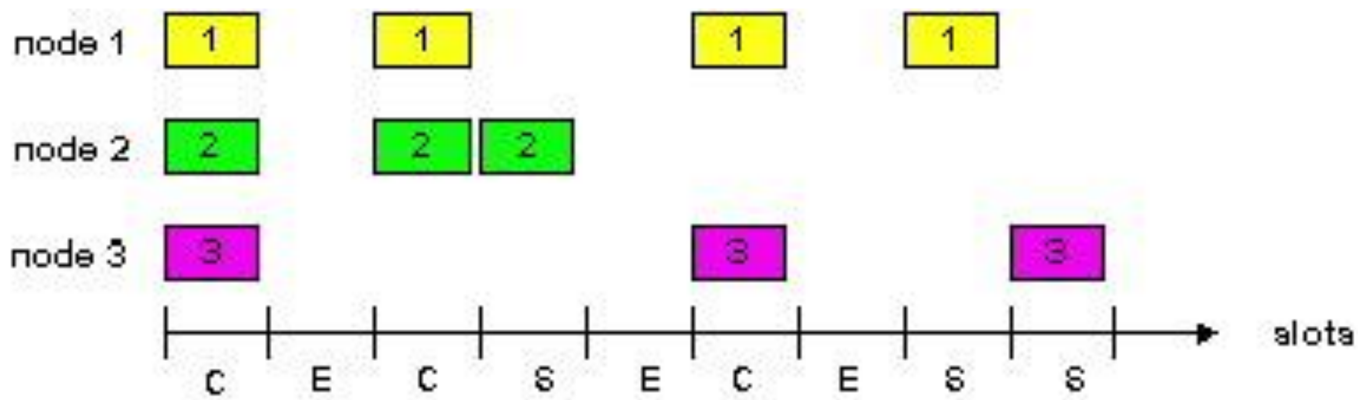
Assumptions:

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

Operation:

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

Slotted ALOHA



Pros

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

Cons

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

Slotted Aloha efficiency

Efficiency : long-run fraction of successful slots (many nodes, all with many frames to send)

- *suppose:* N nodes with many frames to send, each transmits in slot with probability p
- prob that given node has success in a slot = $p(1-p)^{N-1}$
- prob that *any* node has a success = $Np(1-p)^{N-1}$

- max efficiency: find p^* that maximizes $Np(1-p)^{N-1}$
- for many nodes, take limit of $Np^*(1-p^*)^{N-1}$ as N goes to infinity, gives:

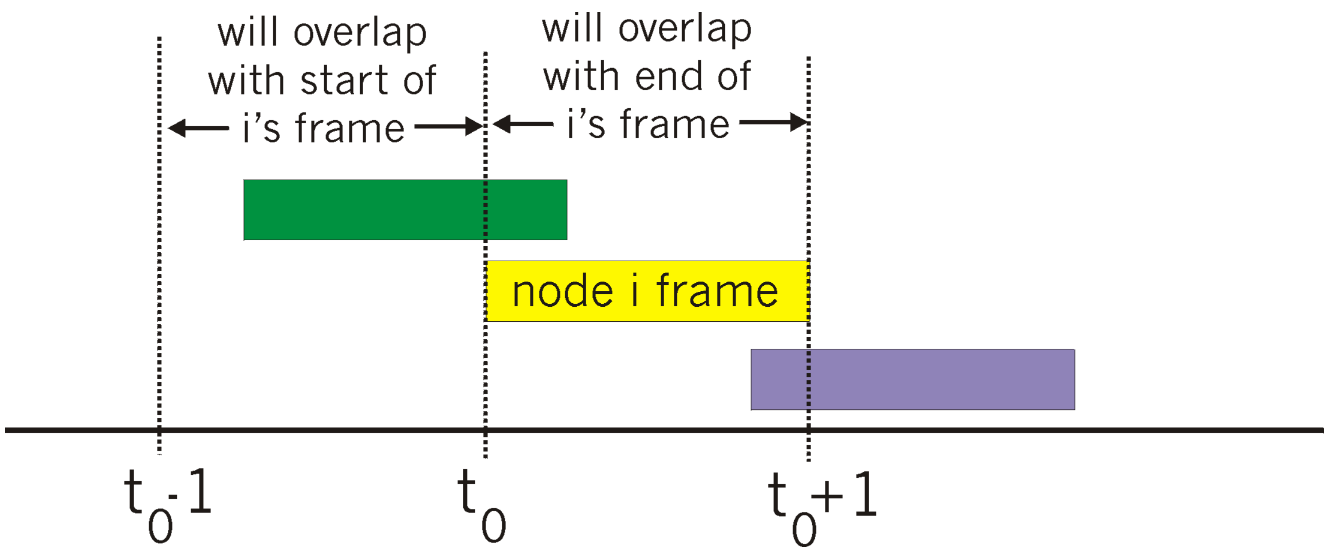
Max efficiency = $1/e$ = .37

At best: channel used for useful transmissions 37% of time!

!

Pure (unslotted) ALOHA

- unslotted Aloha: simpler, no synchronization
- when frame first arrives
  - transmit immediately
- collision probability increases:
  - frame sent at t_0 collides with other frames sent in $[t_0-1, t_0+1]$



Pure Aloha efficiency

P(success by given node) = P(node transmits) $\cdot$

P(no other node transmits in $[t_0-1,t_0]$ $\cdot$

P(no other node transmits in $[t_0, t_0+1]$

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

$= p \cdot (1-p)^{2(N-1)}$

… choosing optimum p and then letting n -> infty ...

= 1/(2e) = .18

even *worse* than slotted Aloha!

CSMA (Carrier Sense Multiple Access)

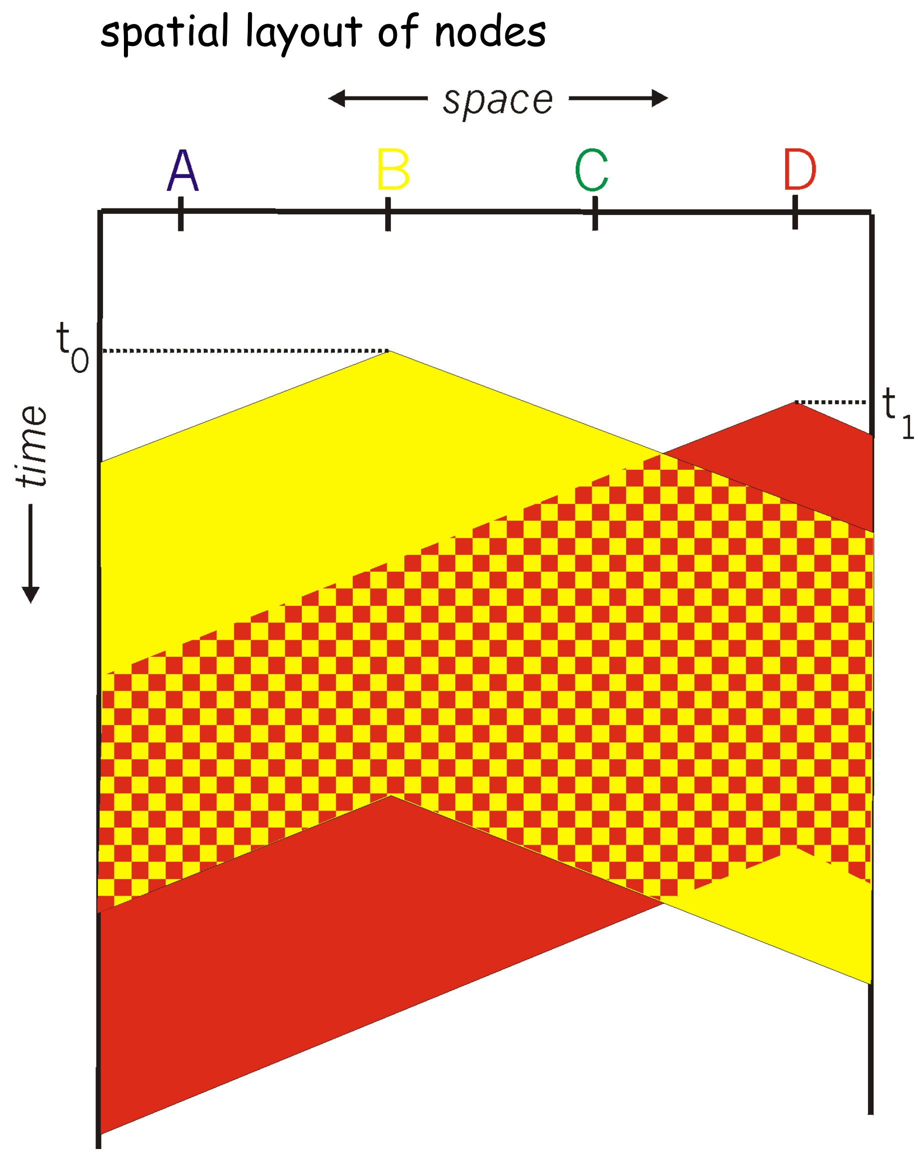
CSMA: listen before transmit:

If channel sensed idle: transmit entire frame

- If channel sensed busy, defer transmission

- human analogy: don't interrupt others!

CSMA collisions



collisions can still occur:
propagation delay means two nodes may not hear each other's transmission

collision:
entire packet transmission time wasted

note:
role of distance & propagation delay in determining collision probability

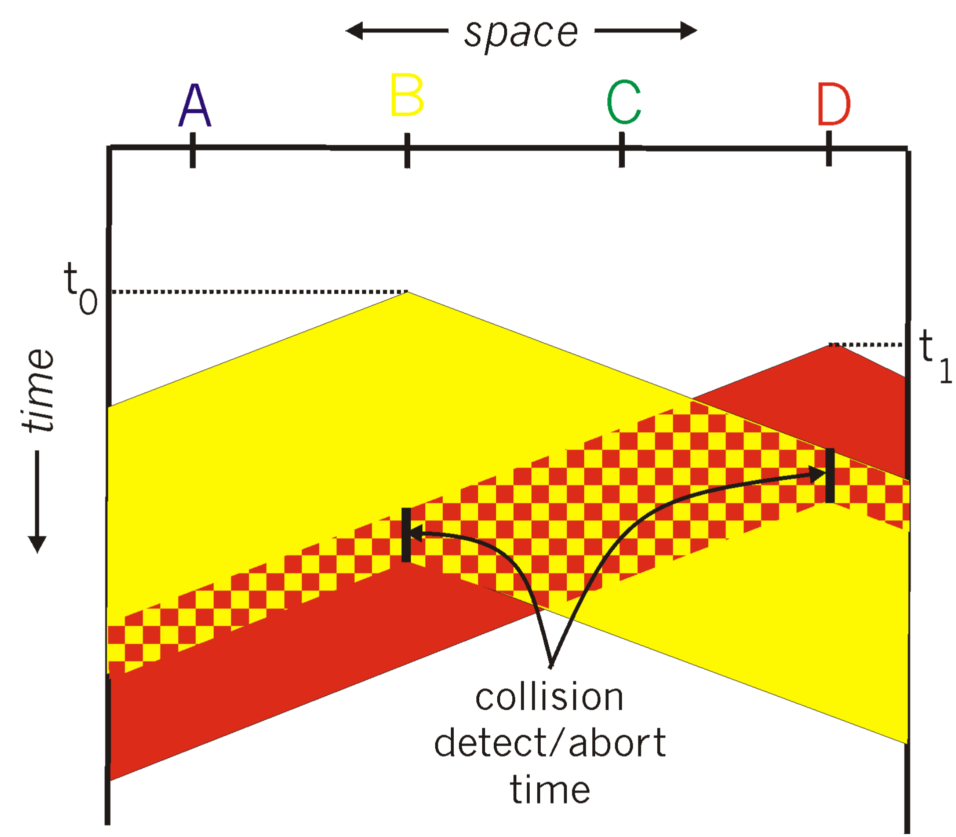
CSMA/CD (Collision Detection)

CSMA/CD: carrier sensing, deferral as in CSMA

- collisions *detected* within short time
- colliding transmissions aborted, reducing channel wastage

- collision detection:
  - easy in wired LANs: measure signal strengths, compare transmitted, received signals
  - difficult in wireless LANs: received signal strength overwhelmed by local transmission strength
- human analogy: the polite conversationalist

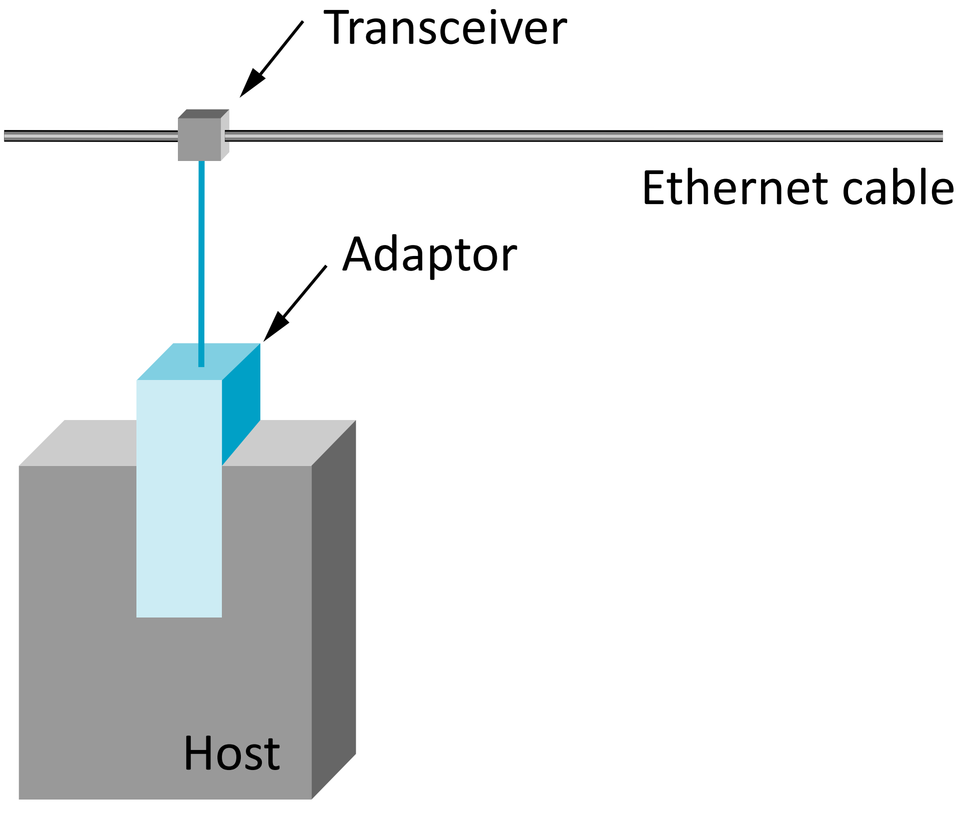
CSMA/CD collision detection



Ethernet (802.3)

Ethernet

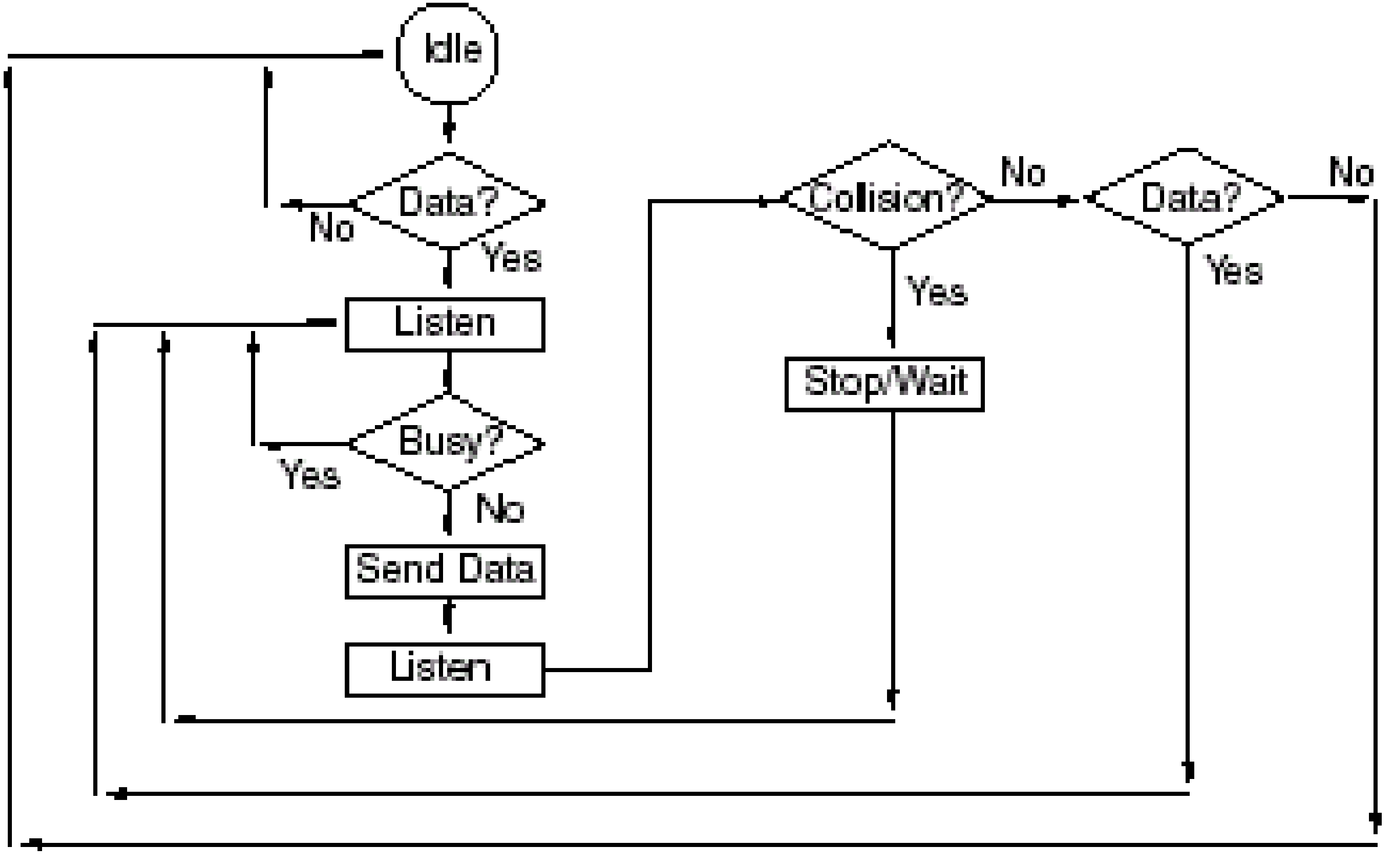
- Ethernet is a **multiple-access network**
  - A set of nodes send and receive frames over a **shared link**



Ethernet

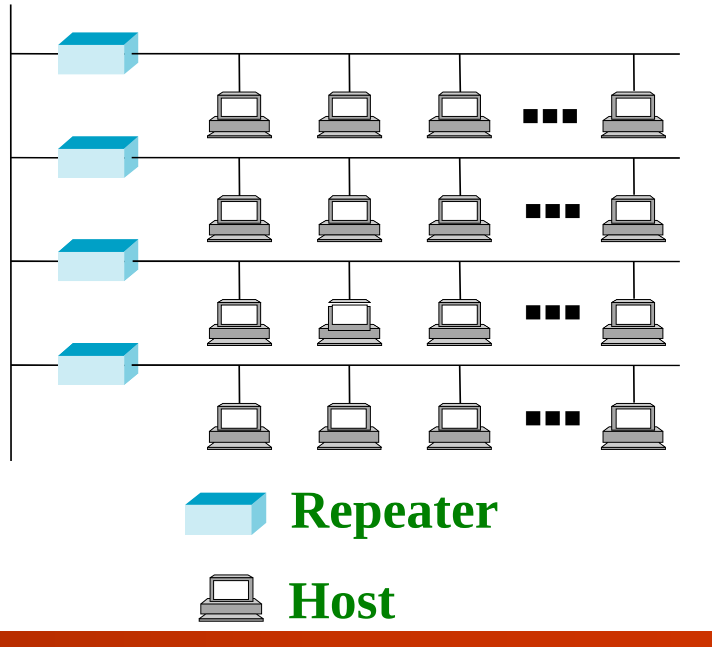
- **CSMA/CD:** Carrier Sense Multiple Access with Collision Detect is the applied multiple access technology
  - **Carrier Sense:** all the nodes can distinguish between an **idle** link and a **busy** link
  - **Collision Detect:** a node listens as it transmits and can therefore **detect the occurrence of a collision**
- 10-Mbps Ethernet standard: typically used in **multiple-access mode**
- 100-Mbps **Fast Ethernet** standard and 1000-Mbps **Gigabit Ethernet** standard: generally used in **full-duplex, point-to-point** configurations $\Rightarrow$ used in **switched networks**

CSMA/CD


Idle
Data?
No
Yes
Listen
Busy?
Yes
No
Send Data
Listen
Collision?
No
Yes
Stop/Wait
Data?
No
Yes

Physical Properties

- Ethernet implemented on a **coaxial cable** (cable TV) can support a transmission distance of up to **500 m**
- Multiple Ethernet segments are jointed together by **repeaters**
  - No more than **4 repeaters** may be positioned between any pair of nodes
  - A total reach of 2500 m
    - $(4+1) \times 500 = 2500$ m
- Support a maximum of **1024 hosts**

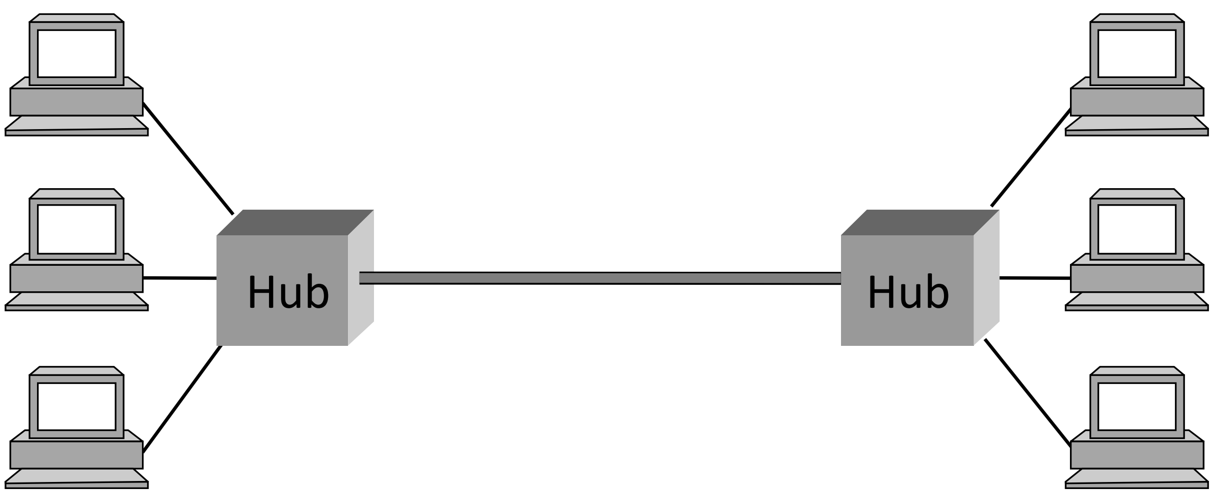


Physical Properties

- Any signal placed on the Ethernet by a host is **broadcast** over the entire network
- **10Base5** (thick-net): original cable
  - 10: the network operates at 10 Mbps
  - Base: a baseband system **(no carrier)**
  - 5: a segment can be no longer than 500 m
- **10Base2** (thin-net): thinner cable
  - 2: a segment can be no longer than 200 m
- **10BaseT**: new cable, Category 5 **twisted pair**
  - T: twisted pair, a segment can be no longer than 100 m
- 100-Mbps and 1000-Mbps Ethernets also run over Category 5 twisted pair

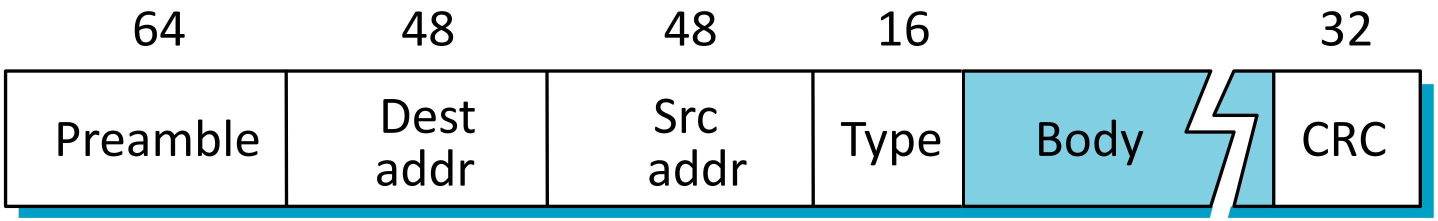
Physical Properties

- With 10BaseT:
  - Several point-to-point segments can be connected by a multiway repeater
  - **Hub:** the multiway repeater
- Multiple 100-Mbps Ethernet segments can also be connected by a hub



Access Protocol

- Frame format:



  - **64-bit preamble:** allows the receiver to synchronize with the signal (010101…010101)
  - **48-bit address** (Dest addr & Src addr): identify the **source** and **destination** hosts
  - **16-bit packet type:** identifies higher-level protocols
  - **Body:** up to 1500 bytes of data, at least **46 byte** of data (for the detection of a collision)
  - **32-bit CRC**

Addresses

- Every Ethernet host in the world has a **unique** **Ethernet address**
  - The address belongs to the adaptor (not the computer)
- Ethernet address
  - A sequence of six numbers separated by colons
  - Leading 0s are dropped
- For example: **8:0:2b:e4:b1:2**

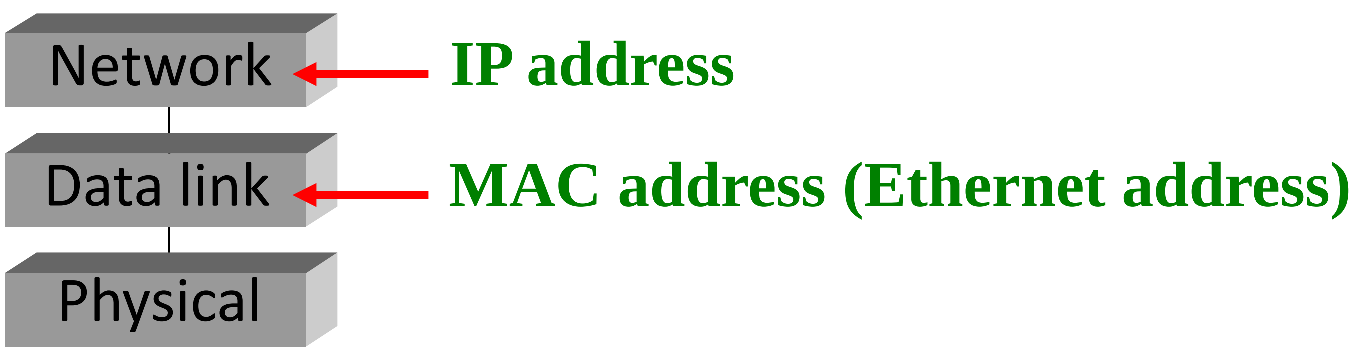
8 : 0 : 2 b : e 4 : b 1 : 2

00001000 00000000 00101011 11100100 10110001 00000010

- Each manufacturer of Ethernet devices is allocated a different prefix

MAC Address VS IP Address

- MAC address (such as the Ethernet address) and IP address are used for different layers
  - MAC address: Data link layer
  - IP address: Network layer
- **MAC address** is valid only **in a network**
  - A physical network
- **IP address** is valid for the **whole internetwork**
  - Multiple physical networks



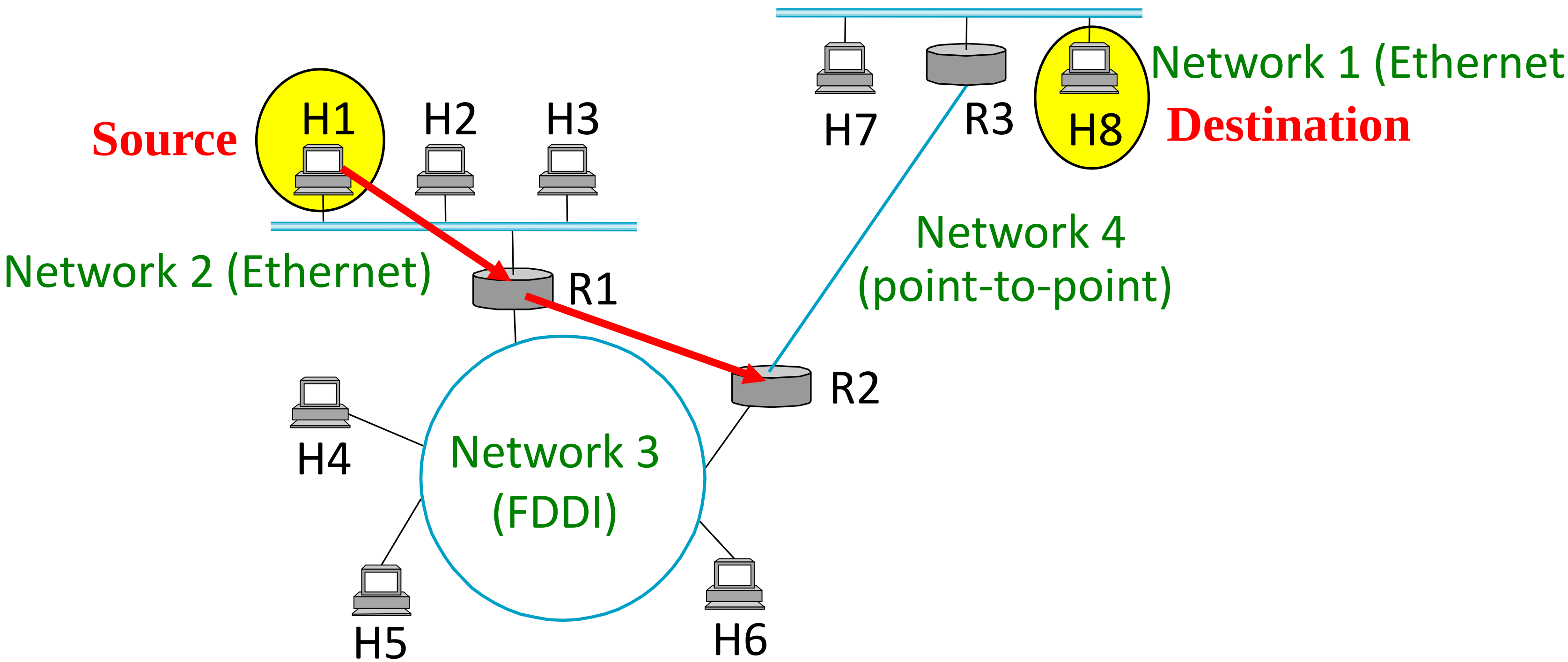
MAC Address VS IP Address

- **H1:** destination IP address is not in the same network

⇒ H1→ R1 in network 2 (Ethernet address)

- **R1:** destination IP address should go to R2

⇒ R1→ R2 in network 3 (FDDI address)



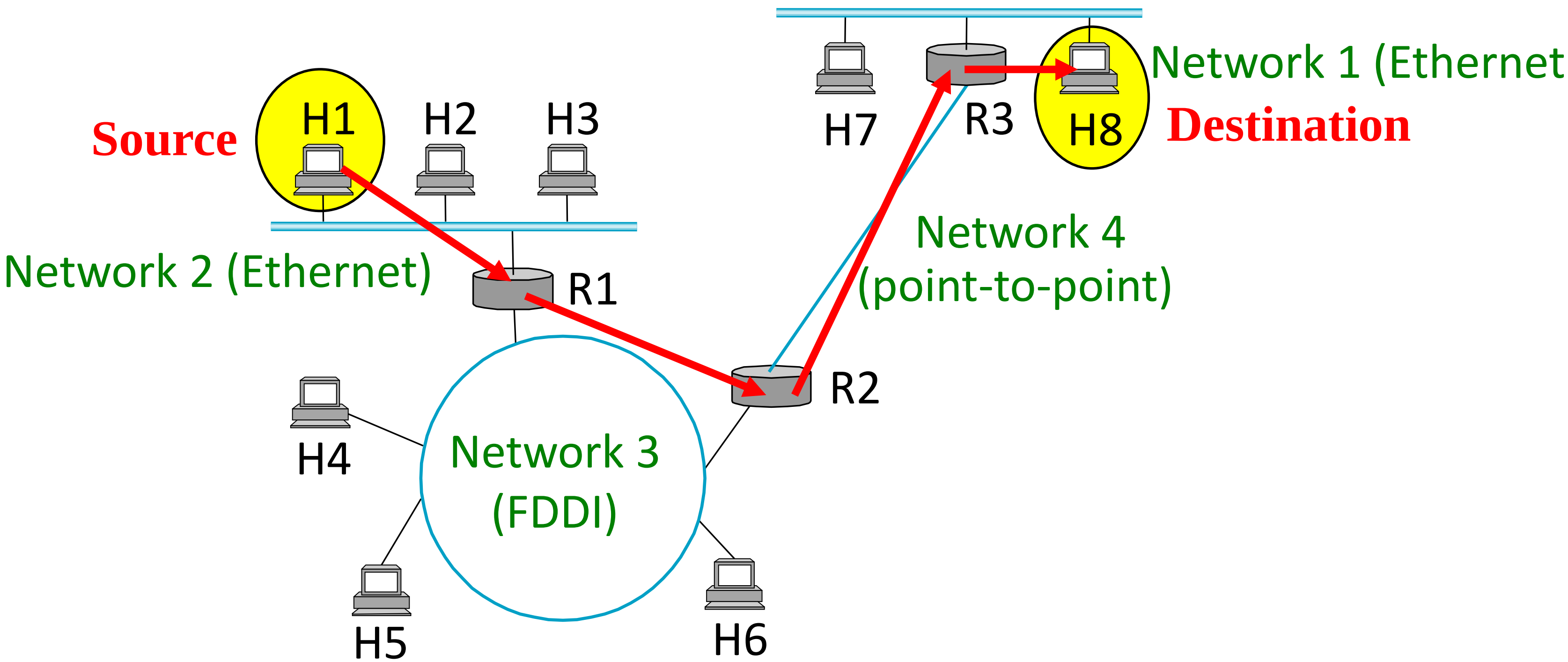
MAC Address VS IP Address

- **R2:** destination IP address should go to R3

$\Rightarrow$ R2$\rightarrow$ R3 in network 4 (point-to-point transmission)

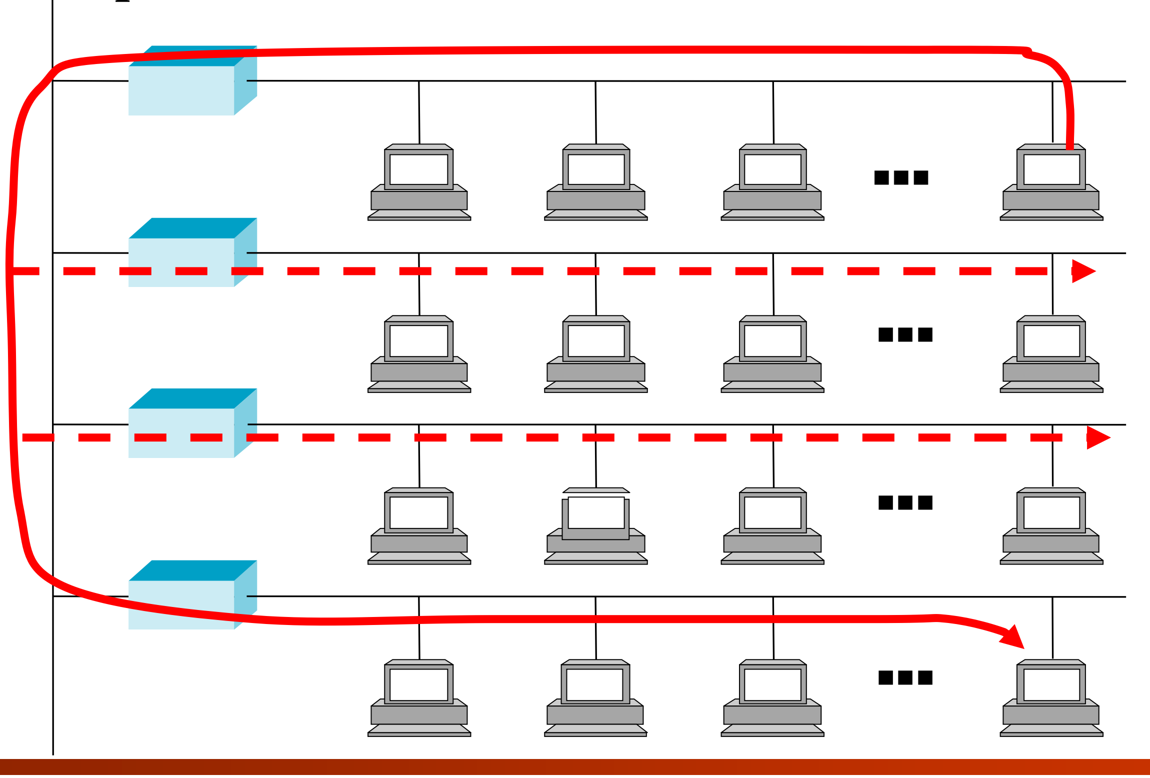
- **R3:** destination IP address is in the same network

$\Rightarrow$ R3$\rightarrow$ H8 in network 1 (Ethernet address)



Receiving

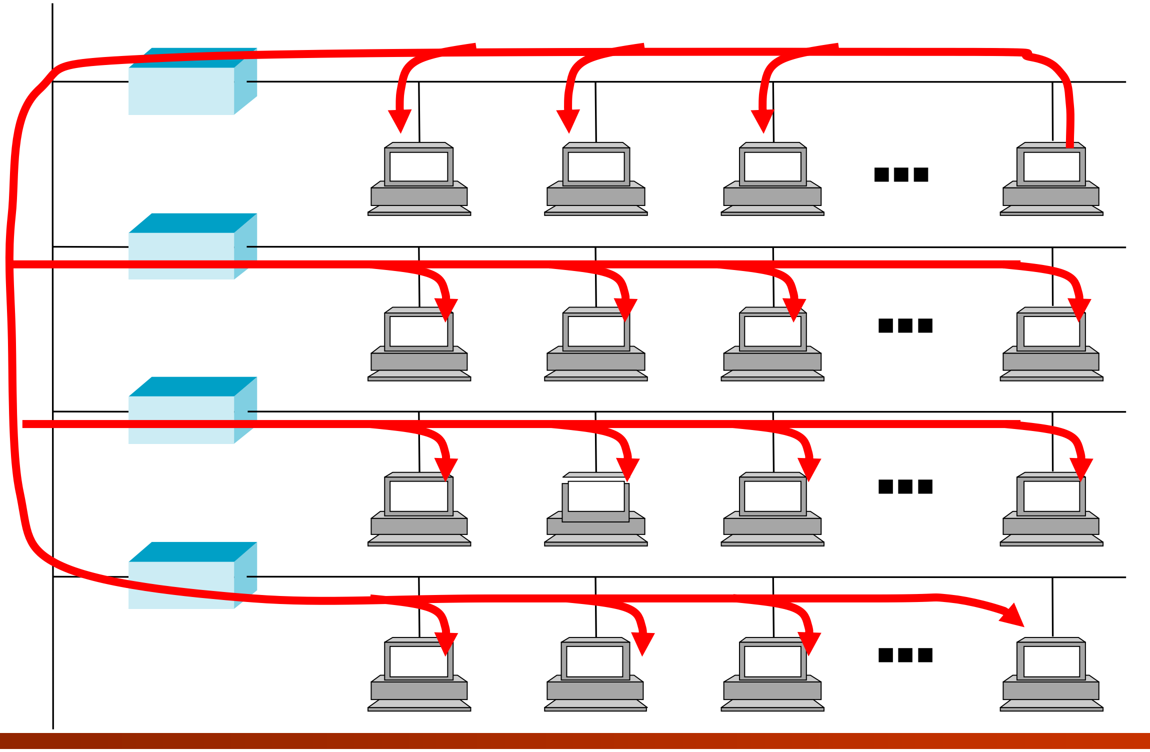
- Each transmitted frame is received by **every** adaptor connected to the Ethernet
- **Unicast:** each adaptor passed those frames addressed to **its address** up to the host



Receiving

- **Broadcast:** each adaptor passed those frames addressed to the **broadcast address** up to the host
  - An Ethernet address consisting of **all '1's**

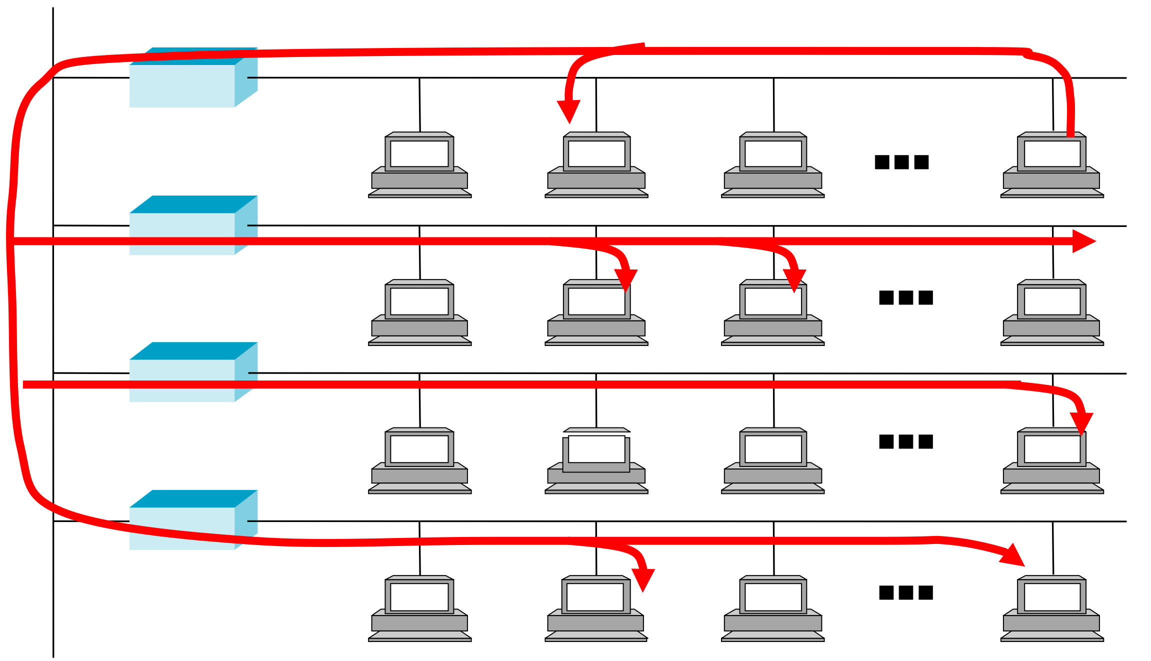
11111111 11111111 11111111 11111111 11111111 11111111



Receiving

- **Multicast:** a host can program its adaptor to accept some set of **multicast addresses**
  - An Ethernet address that has an **odd** number in the second hexadecimal digital but is not the broadcast address

<u>xxxxxxx1</u> <u>xxxxxxxx</u> <u>xxxxxxxx</u> <u>xxxxxxxx</u> <u>xxxxxxxx</u> <u>xxxxxxxx</u>



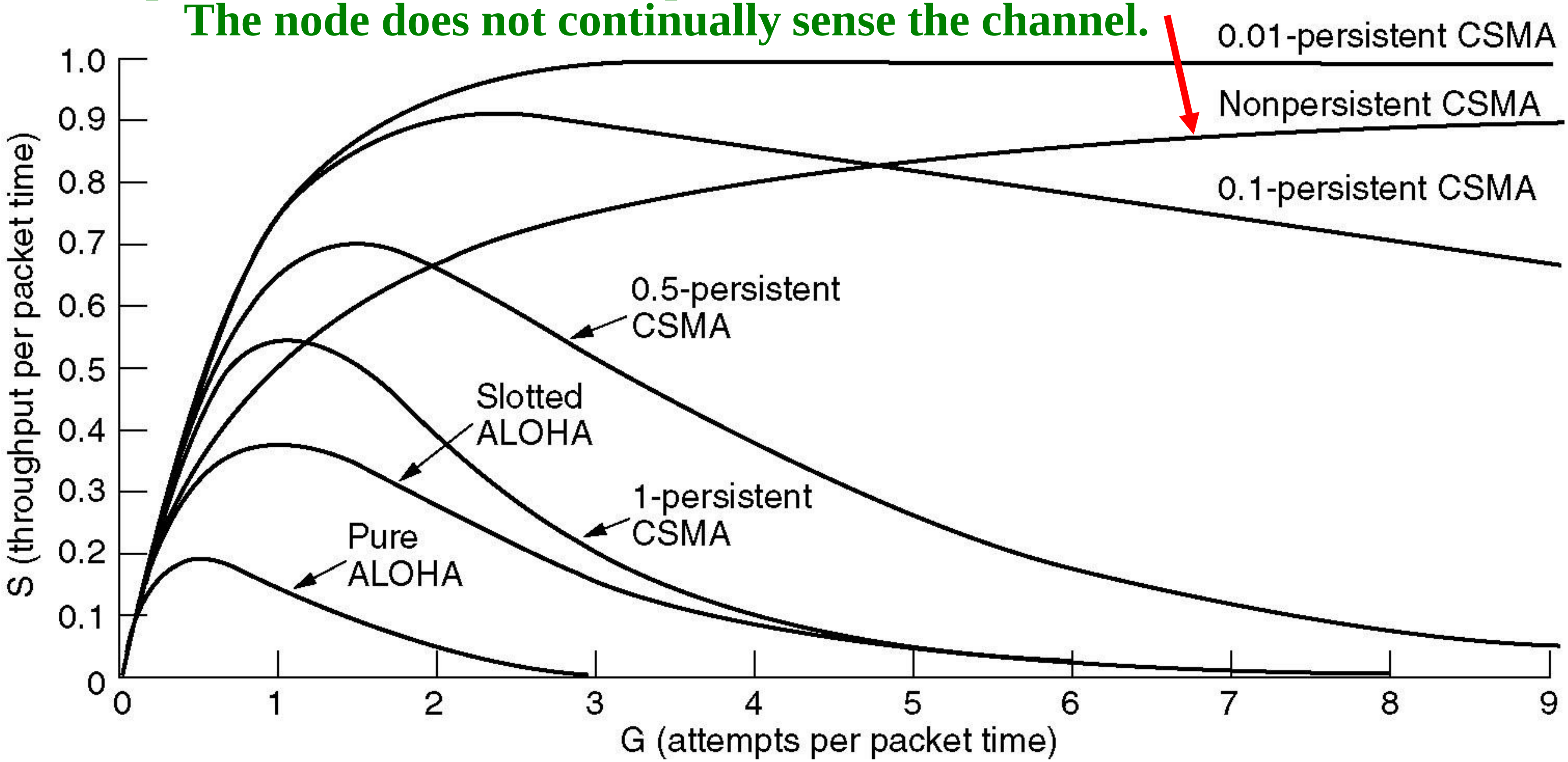
Transmission Algorithm

- When an adapter has a frame to send and the line is **busy**
  - **1-persistent:** it waits for the line to go idle and then transmits **immediately**
  - ***p*-persistent:** it waits for the line to go idle and then transmits **with probability $0 \le p \le 1$**, and defers with probability $q = 1 - p$
- ***p*-persistent:**
  - If the next slot is also **empty**, the node again decides to transmit or defer with probability p and q, respectively
  - If the next slot is **not empty** (some other node has decided to transmit), the node waits for the line to go idle
- There might be multiple adaptors waiting for the busy line to become idle $\Rightarrow$ 1-persistent may induce collision

Transmitter Algorithm

- Throughput versus Traffic load

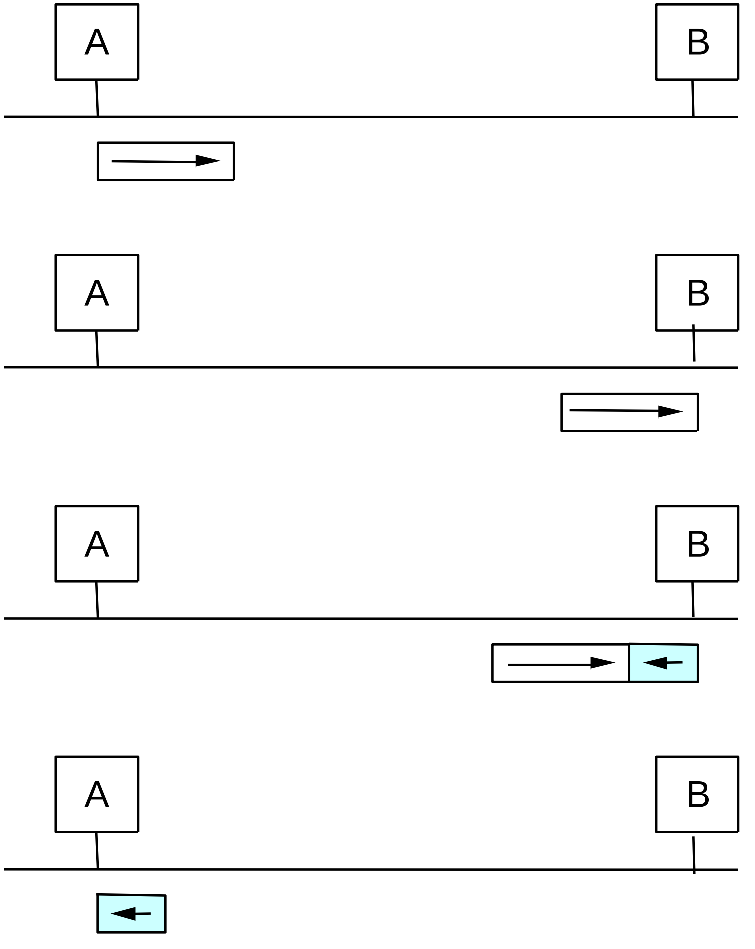
Nonpersistent: If the channel is busy, a node waits a random period of time and then repeats to sense the channel. The node does not continually sense the channel.



Algorithm (cont)

- If collision…
  - jam for 32 bits, then stop transmitting frame
  - minimum frame is 64 bytes (header + 46 bytes of data)
  - delay and try again
    - 1st time: 0 or 51.2us
    - 2nd time: 0, 51.2, or 102.4us,153.6us
    - *nth* time: k x 51.2us, for randomly selected k=0..2^n - 1
    - give up after several tries (usually 16)
    - exponential backoff

Collisions



"Taking Turns" MAC protocols

channel partitioning MAC protocols:

- share channel *efficiently* and *fairly* at high load
- inefficient at low load: delay in channel access, 1/N bandwidth allocated even if only 1 active node!

Random access MAC protocols

- efficient at low load: single node can fully utilize channel
- high load: collision overhead

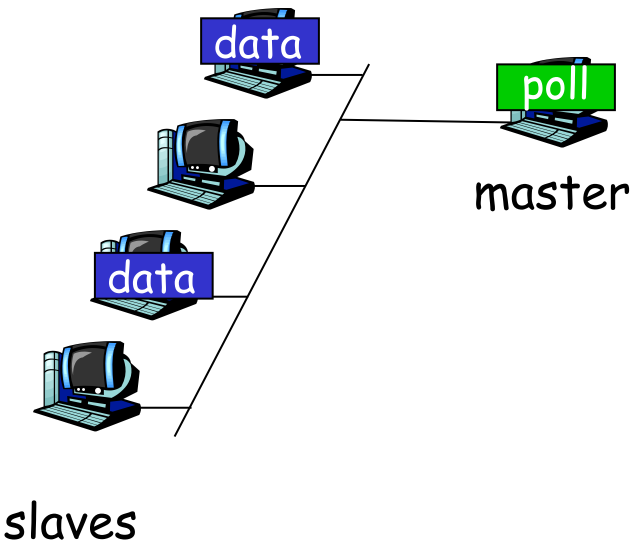
"taking turns" protocols

look for best of both worlds!

"Taking Turns" MAC protocols

Polling:

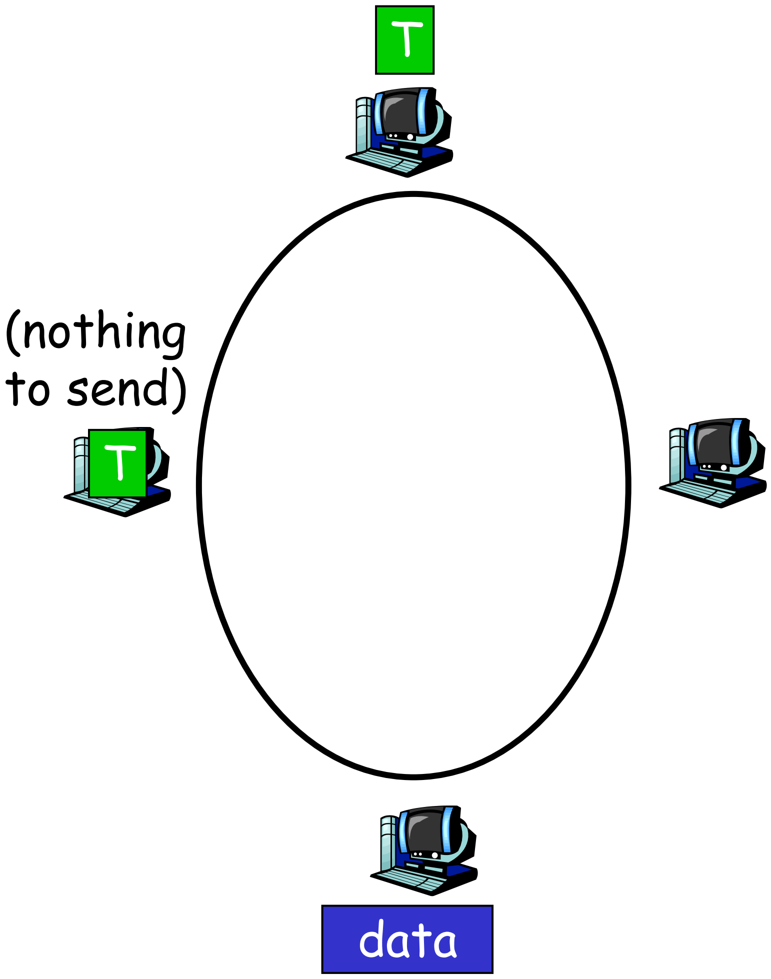
- master node "invites" slave nodes to transmit in turn
- typically used with "dumb" slave devices
- concerns:
  - polling overhead
  - latency
  - single point of failure (master)



"Taking Turns" MAC protocols

Token passing:

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

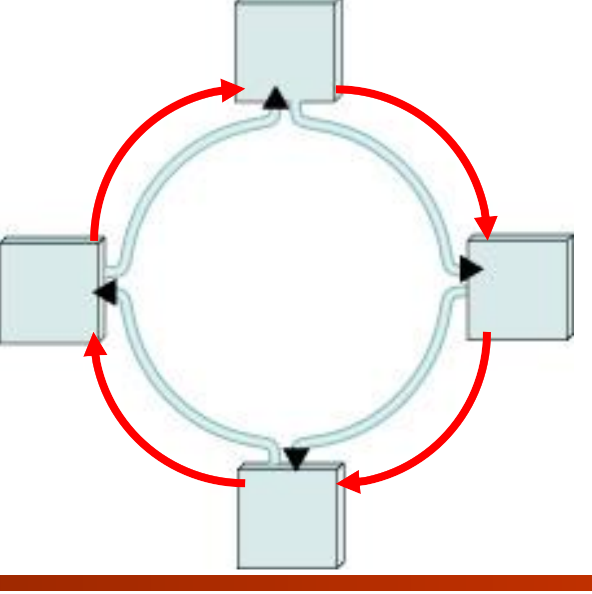


Token Rings (802.5, FDDI)

802.5

Token Rings

- 802.5 standards: is the traditional token ring
- **FDDI (Fiber Distributed Data Interface)** standard: is a newer, faster type of token ring
- A token ring network consists of a set of nodes connected in a ring
- Data always **flows in a particular direction** around the ring

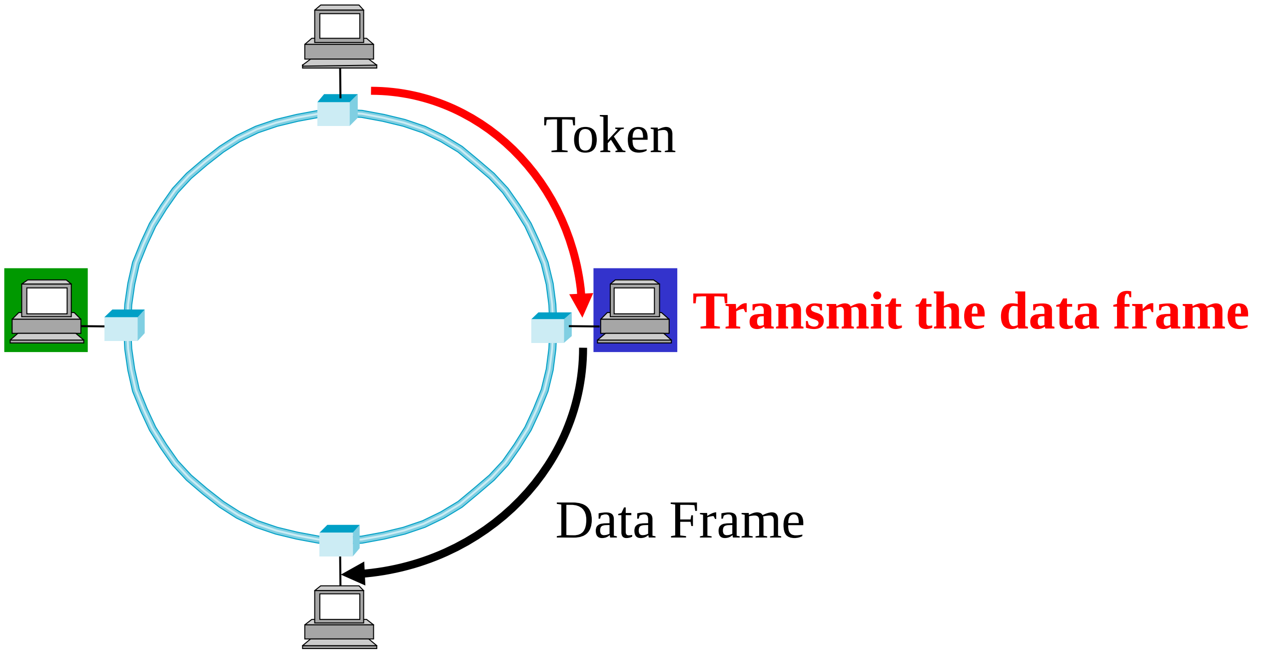


Token Rings

- Each node receives frames from its **upstream** neighbor and then forwards them to its **downstream** neighbor
- The ring is viewed as **a single shared medium**
  - Not behave as a collection of independent point-to-point links
- **Token:** a certificate that a node has the right to access the shared ring
  - A special sequence of bits circulates around the ring
  - Each node receives and then forwards the token

Token Rings

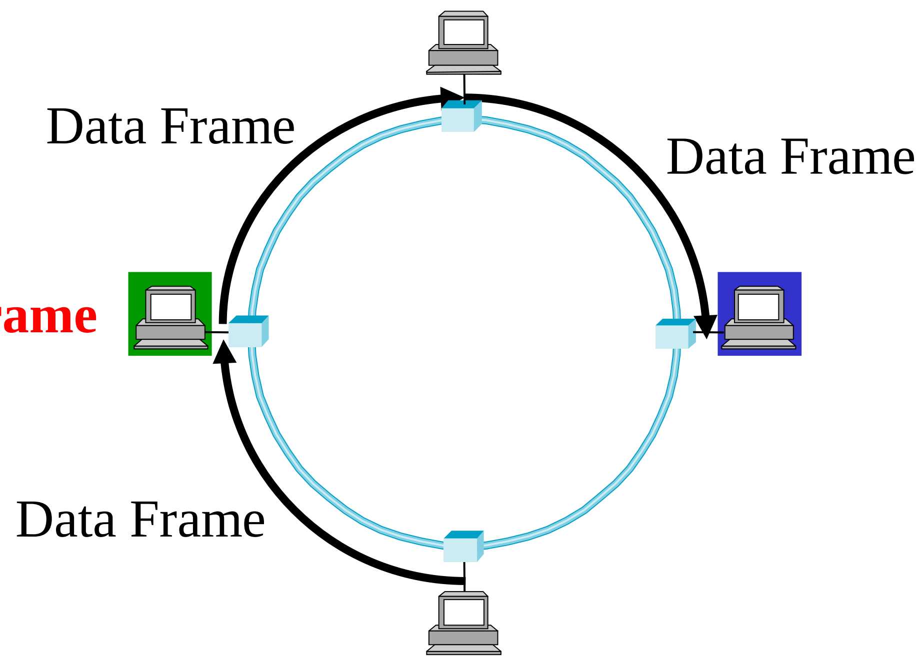
- When a node, that has a frame to transmit, **sees the token**
  - It takes the token **off the ring** (does not forward the token)
  - It inserts its frame into the ring



Token Rings

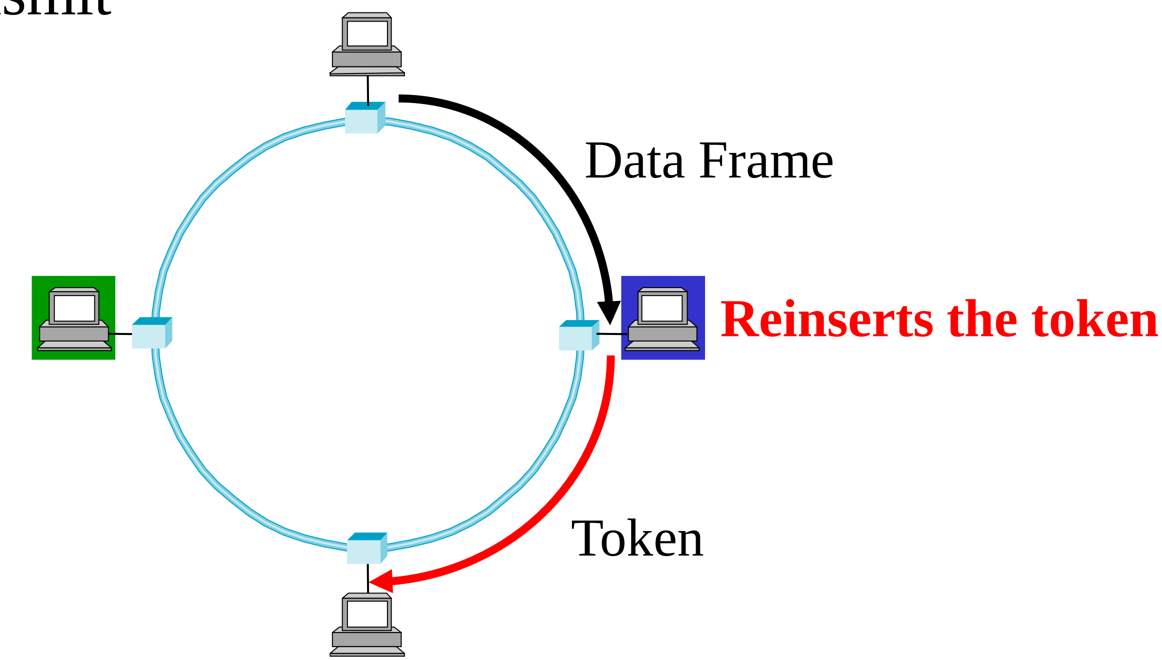
- Each node along the way simply **forwards** the frame
- The destination node **saves a copy and forwards** the frame to the next node on the ring

Save the data frame



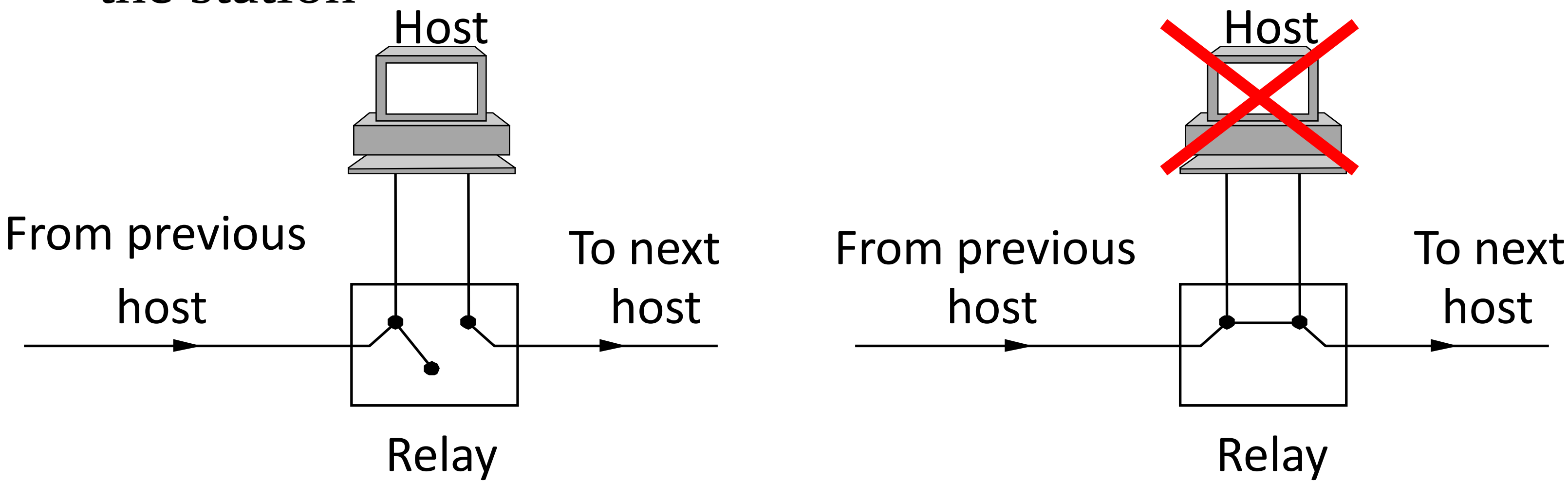
Token Rings

- When the frame makes its way back around to the **sender**
  - It strips its frame off the ring
  - It reinserts the token into the ring
- Since the token **circulates around** the ring, each node gets a chance to transmit



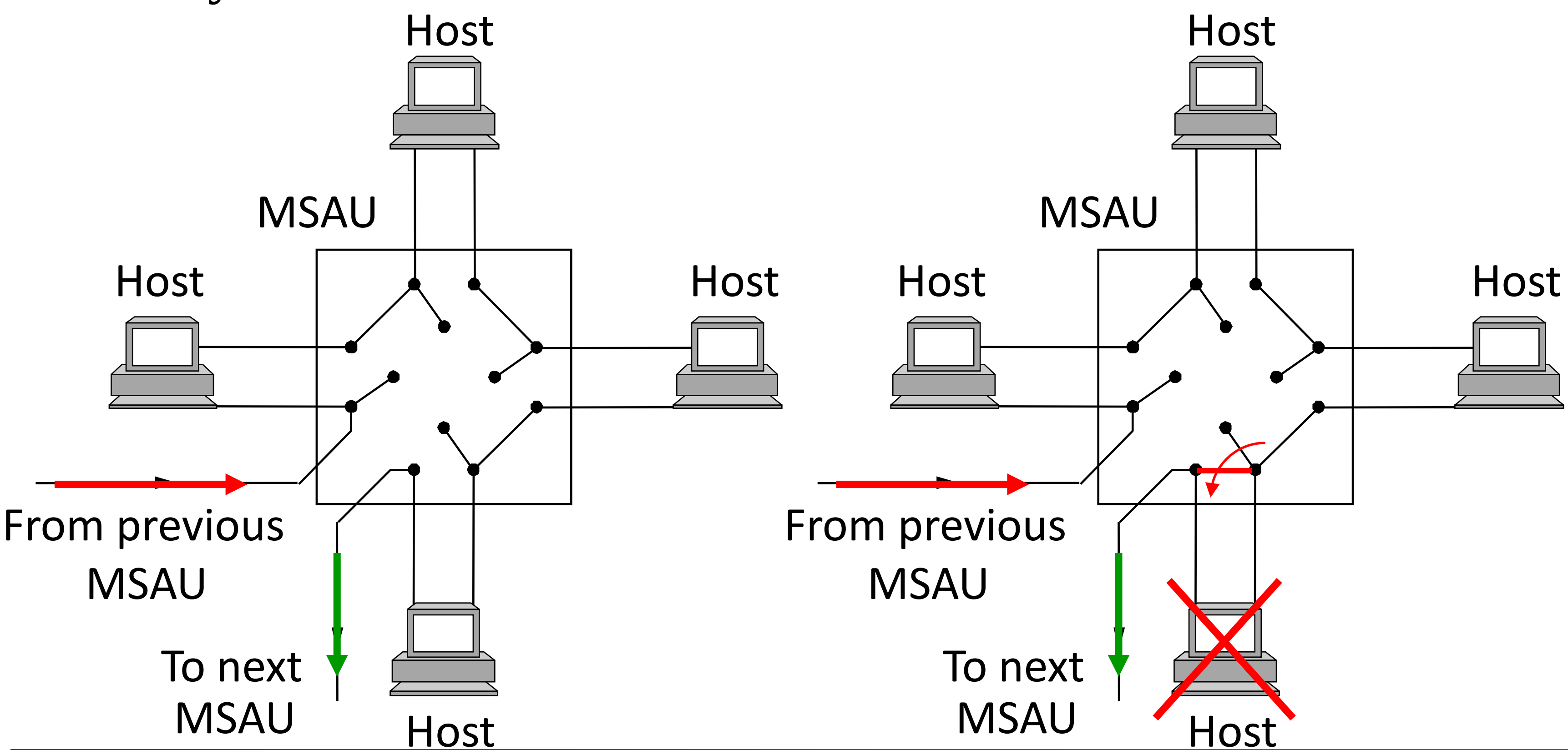
Physical Properties

- Each station is connected into the ring using an **electromechanical rely**
- If the station is **healthy**
  - the relay is **open** and the station is **included** in the ring
- If the station stops providing power or is **failed**
  - the relay is **closed** and the ring automatically **bypasses** the station



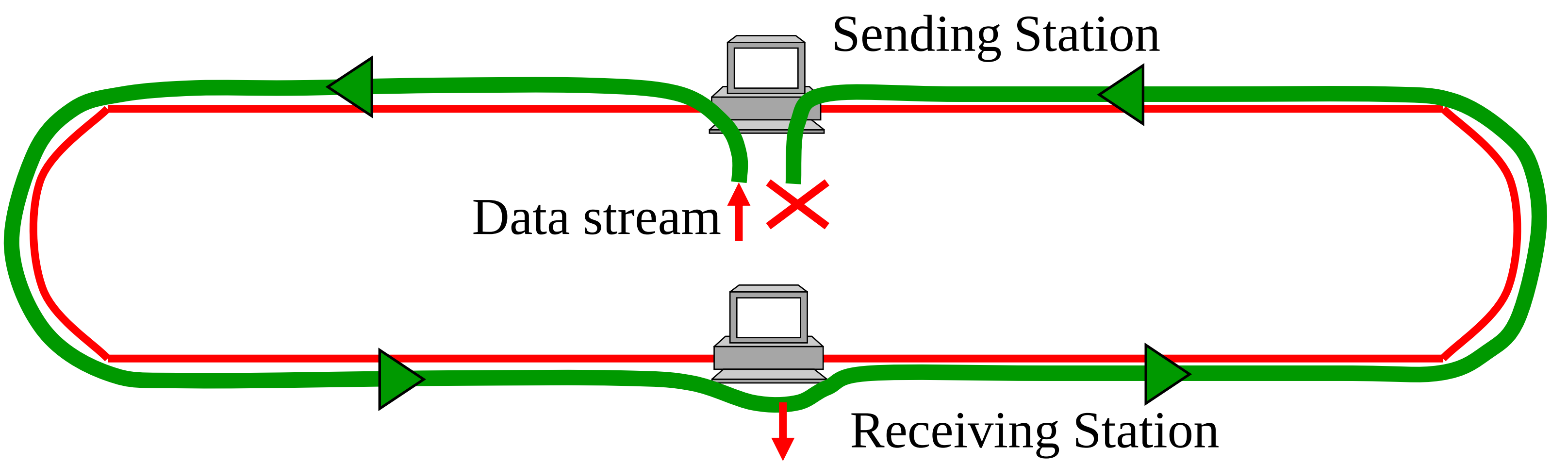
Physical Properties

- Several relays are usually packed into a single box
  - **MSAU: multi-station access unit**
- Easy to add stations to and remove stations from the network



Media Access Control

- The **sending station** has the responsibility of **removing** the packet from the ring
- For any packet **longer** than the number of bits that can be stored in the ring
  - The sending station will be draining the first part of the packet from the ring, while still transmitting the latter part



Media Access Control

- **Token holding time (THT):**
  - How much data a given node is allowed to transmit each time it possesses the token
  - If **larger** THT is used: it has better utilization of the ring, but favors nodes that have a lot of data to send
  - In 802.5 networks, the default THT is **10 ms**
- **The token rotation time (TRT):**
  - The amount of time for a token traversing the ring

$$\mathbf{TRT \leq ActiveNodes \times THT + RingLatency}$$

- **RingLatency:** the time for the token to circulate around the ring when no one has data to send
- **ActiveNodes:** number of nodes that have data to transmit

Media Access Control

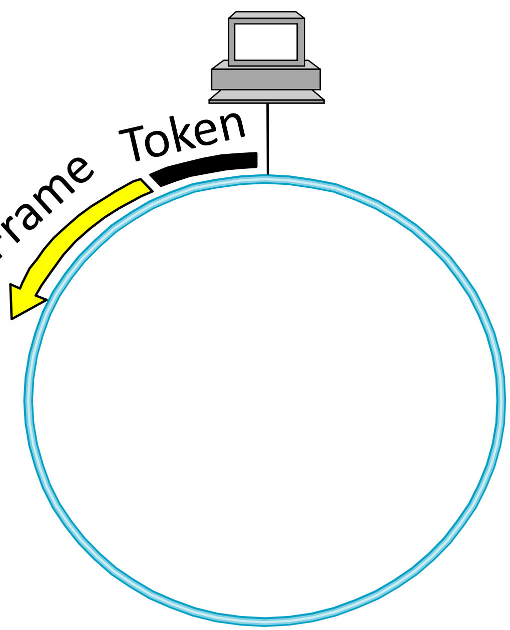
- **Reliable delivery:**
  - Using 2 bits in the packet trailer
    - **A and C bits – '0,0'**
- When a station **sees** a frame for which it is the **intended recipient** – **sets the A bit in the frame** (i.e. '0' → '1')
- When it **copies** the frame into its adaptor – **sets the C bit**
- The sending station sees the frame come back over the ring
  - **A bit still '0':** the intended recipient is not functioning
  - **A and C bits '1,0':** for some reason (lack of buffer), the destination could not accept the frame (retransmit again)
  - **A and C bits '1,1':** the frame has been successfully received by the recipient

Media Access Control

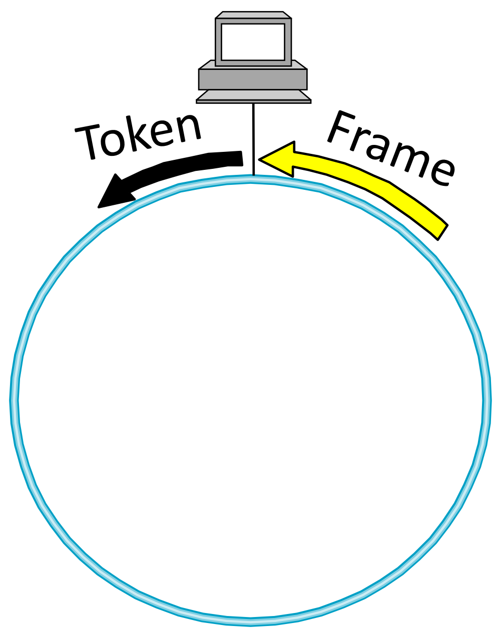
- **Priority:** to support different levels of transmission priority
  - The token contains a **3-bit priority field**
  - The token has a certain priority *n* at a time
- The device can only seize the token to transmit a packet
  - If the packet's priority is **at least as great as the token**
- For example: the token priority is $4 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow 0$
  - For a node has a priority 4, it can transmit a packet for each token circulating cycle
  - For a node has a priority 0, it can transmit a packet only for one token circulating cycle

Media Access Control

- **Token release:**
  - **Early:** inserts the token immediately following its frame
  - **Delayed:** inserts the token after the transmitted frame has been removed



Early release



Delayed release

Frame Format - 802.5

- 802.5 uses **differential Manchester** encoding

    ‘0’: maintain the code (01 or 10);
    ‘1’: change the code (01→10 or 10→01)

- Use **“illegal”** Manchester codes in the **start** and **end** delimiters (‘0’→01; ‘1’→10; illegal code: **‘00 00 11 11’**)

8	8	8	48	48	Variable	32	8	8
Start delimiter	Access control	Frame control	Dest addr	Src addr	Body	Checksum	End delimiter	Frame status

  - **Access control:** frame priority, reservation priority, …
  - **Frame control:** de-multiplex key (higher-layer protocol)
  - **48-bit address** (Dest addr & Src addr)
  - **Body:** variable length
  - **Checksum:** 32-bit CRC
  - **Frame status:** includes the A and C bits

Maintenance

- Monitoring for a Valid Token
  - should periodically see valid transmission (frame or token)
  - maximum gap = ring latency + max frame < = 25ms
  - set timer at 25ms and send claim frame if it fires

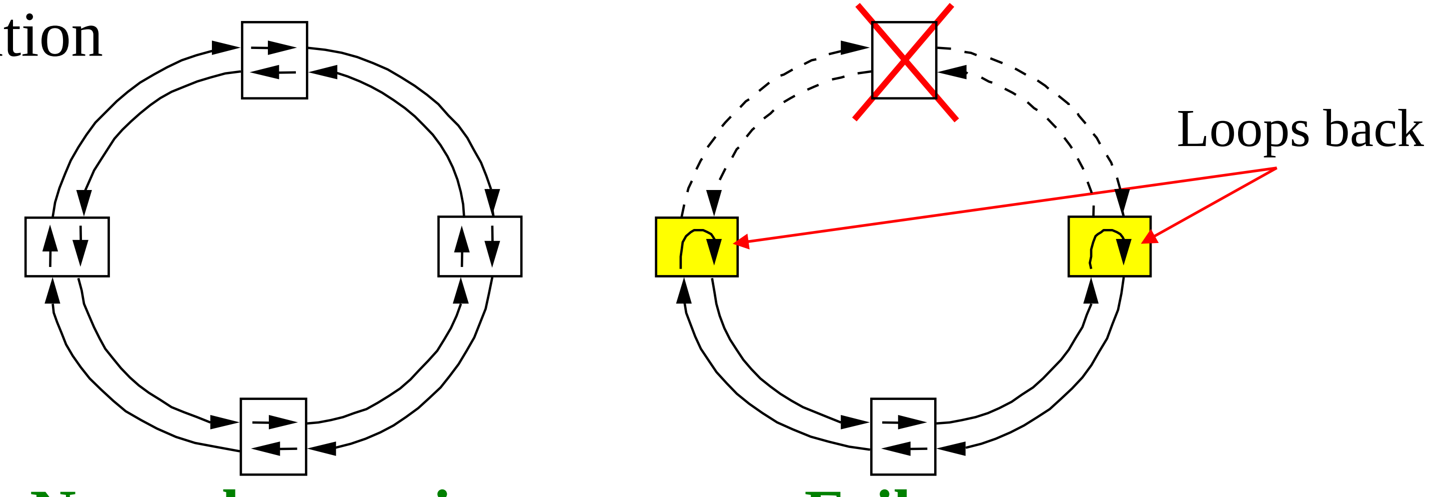
Token Maintenance

- Lost Token
  - no token when initializing ring
  - bit error corrupts token pattern
  - node holding token crashes
- Generating a Token
  - execute when join ring or suspect a failure
  - send a *claim frame* that includes the node's *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
    - you insert new token

FDDI

Physical Properties

- An **FDDI** network consists of a **dual ring**
  - Two independent rings that transmit data in **opposite directions** – the **primary** and **secondary** rings
- The secondary ring is not used during normal operation
  - It comes into play only if the primary ring fails
- FDDI can tolerate **a single break** in cable or failure of one station



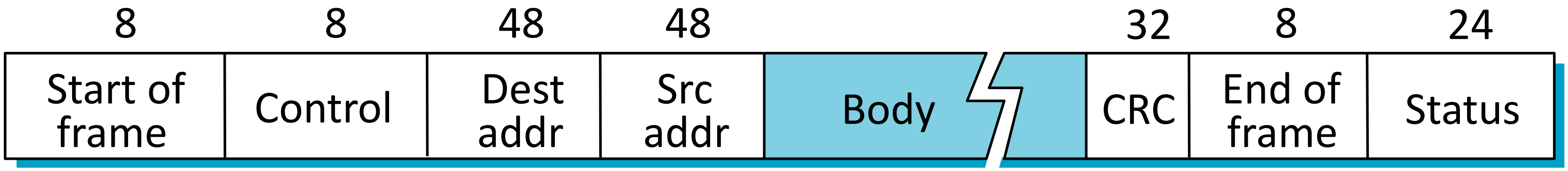
Normal operation **Failure**

Physical Properties

- The standard limits a single FDDI network to at most **500 stations**
  - With a maximum distance of **2 km** between any pair of stations
- The network is limited to a total of **200 km** of fiber
  - The total amount of cable connecting all stations is limited to **100 km**
- FDDI is defined to run over a number of different physical medium
  - Including **coax and twisted pair**
- FDDI uses **4B/5B** encoding

Frame Format -FDDI

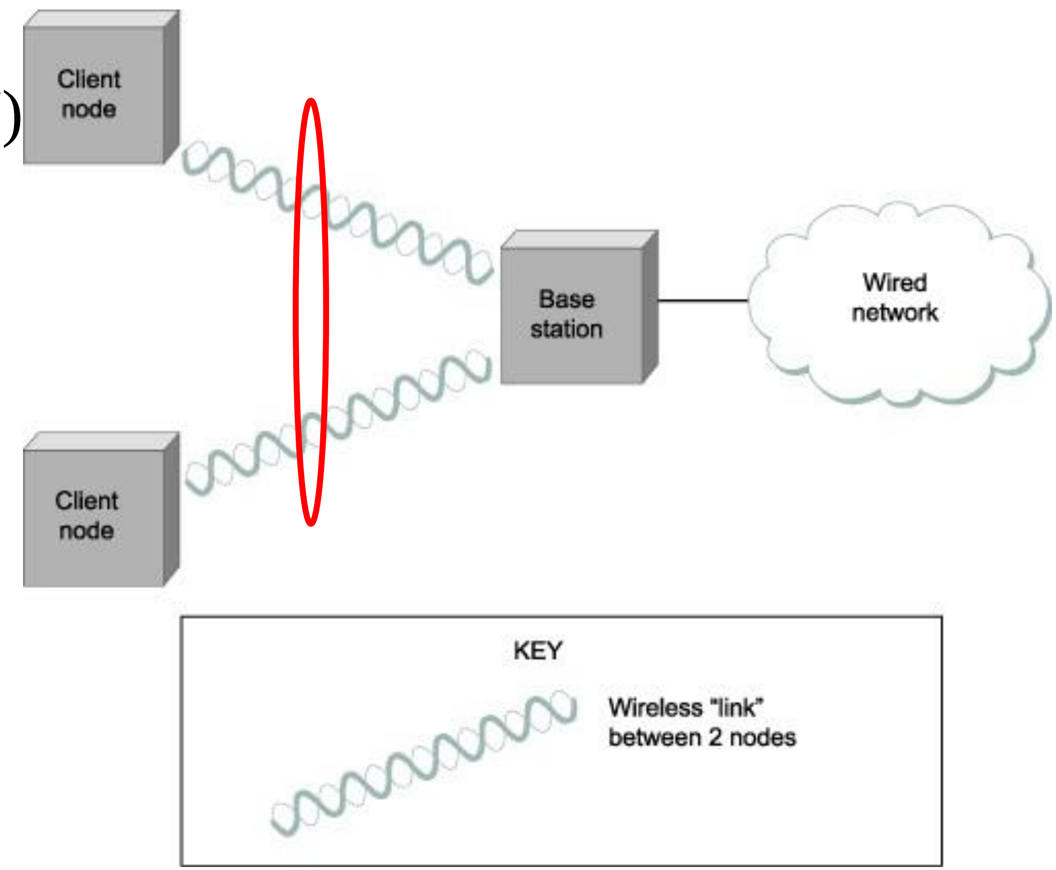
- Use **4B/5B control symbols**, rather than illegal Manchester symbols, in the start- and end-of-frame markers



Wireless
(802.11/802.15.1/802.16/3G Cellular)

Wireless Network

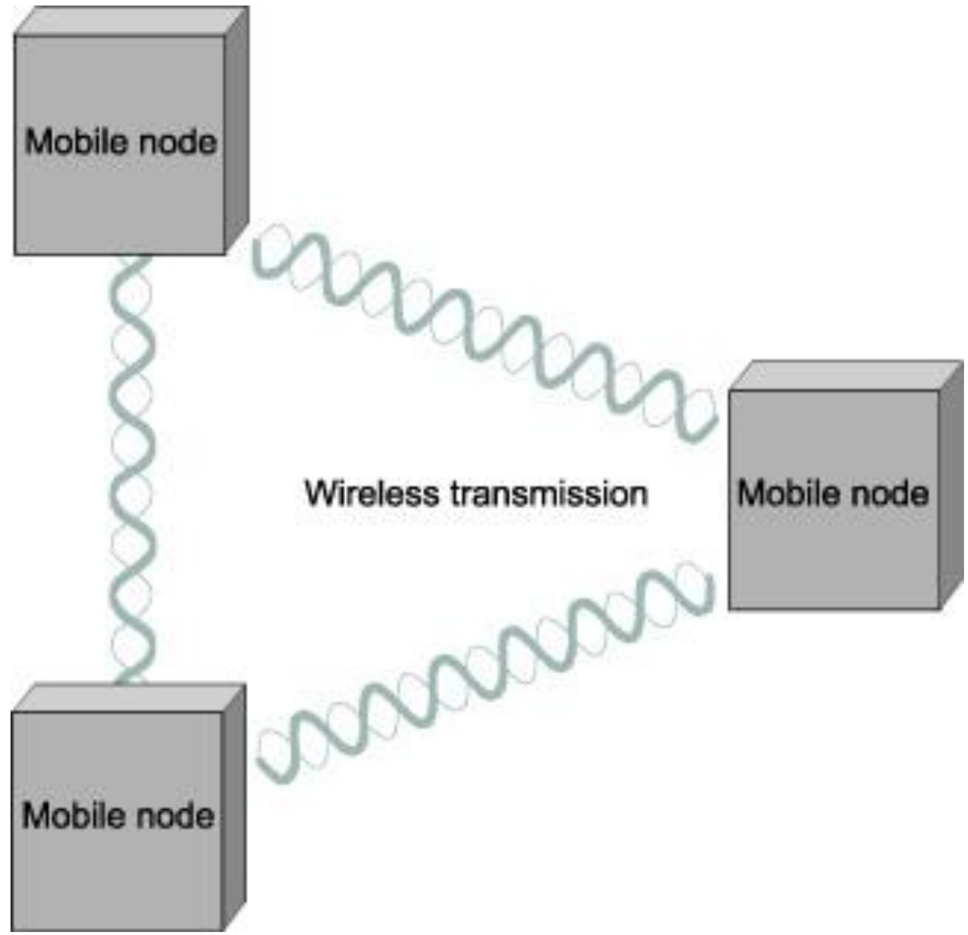
- Four prominent wireless technologies are discussed:
  - **Bluetooth**
    (IEEE 802.15.1 WPAN)
  - **Wi-Fi**
    (IEEE 802.11 WLAN)
  - **Wi-MAX**
    (IEEE 802.16 WMAN)
  - **3G cellular systems**



A wireless network using a base station

Wireless Network

- For **infrastructure** architecture, all client nodes are routed via **base stations**.
- For **ad hoc** or **mesh** architecture, nodes are **peers**, and messages may be forwarded via a chain of peer nodes.



A wireless ad hoc or mesh network

Wireless Network

	Bluetooth (802.15.1)	Wi-Fi (802.11)	Wi-MAX (802.16)	3G cellular
Typical range	10 m	100 m	~ Km	~ Km
Typical bandwidth	2.1 Mbps (shared)	54 Mbps (shared)	70 Mbps (shared)	384+ Kbps (per link)
Typical use	Link a peripheral to a computer (cell phone…)	Link a computer to a wired base	Link a building to a wired tower	Link a cell phone to a wired tower
Wired technology	USB	Ethernet	Cable/DSL	DSL

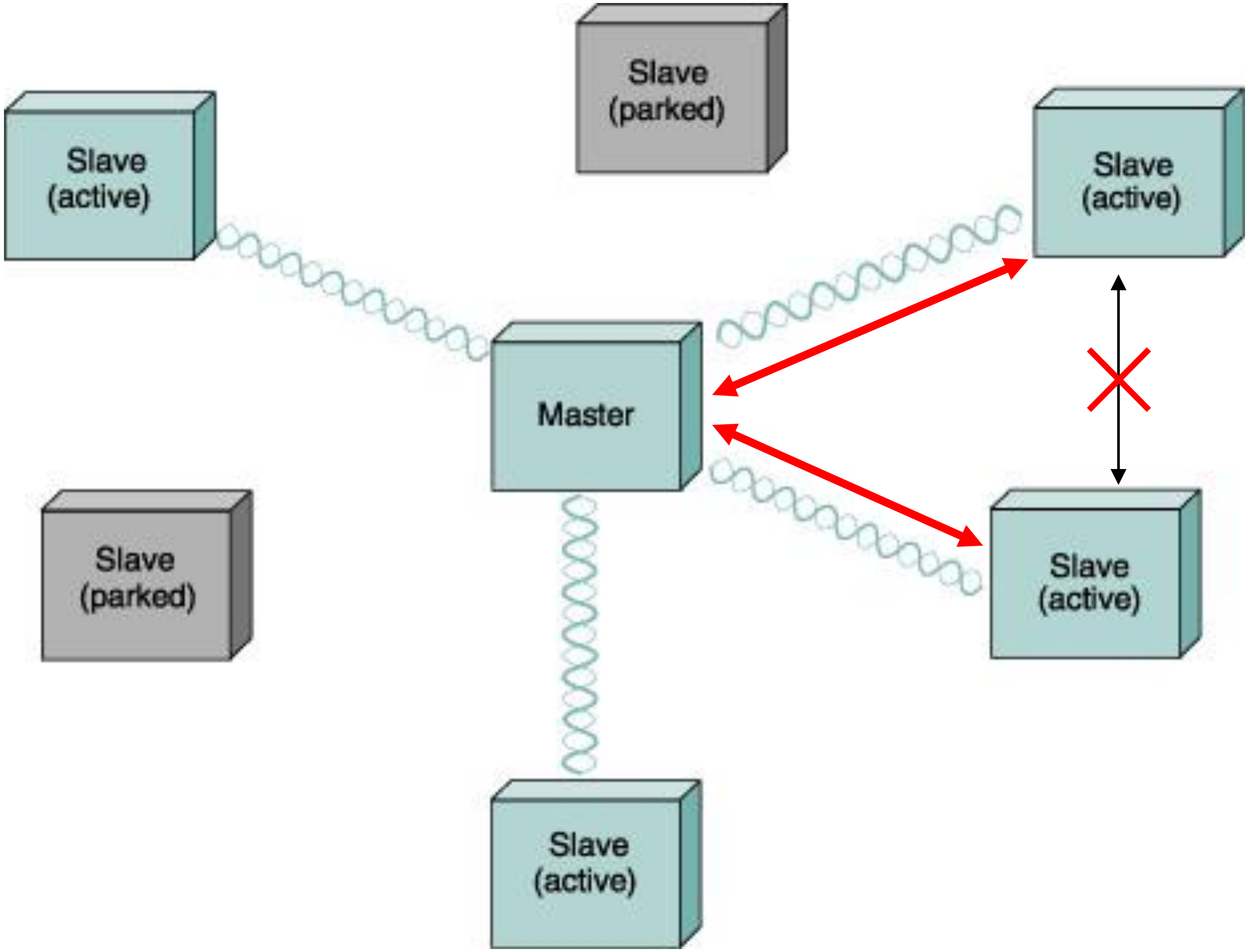
Bluetooth
(IEEE 802.15.1)

Bluetooth

- **Bluetooth** targets on very short-range communication between mobile phones, PDAs, computers and other personal or peripheral devices
- Bluetooth operates in the un-licensed band at 2.45 GHz
- The basic Bluetooth network configuration is called a **piconet**, which consists of
  – A **master** device and
  – Up to **7 slave** devices
  – A slave device can be parked (set to an **inactive, low-power** state): cannot communicate on the piconet
- Any communication is **between the master and a slave**
  – The slavers do not communicate directly with each other
  – Make slavers simpler and cheaper

Bluetooth

- **ZigBee** is a new technology that competes with Bluetooth



WLAN
IEEE 802.11

Physical Properties

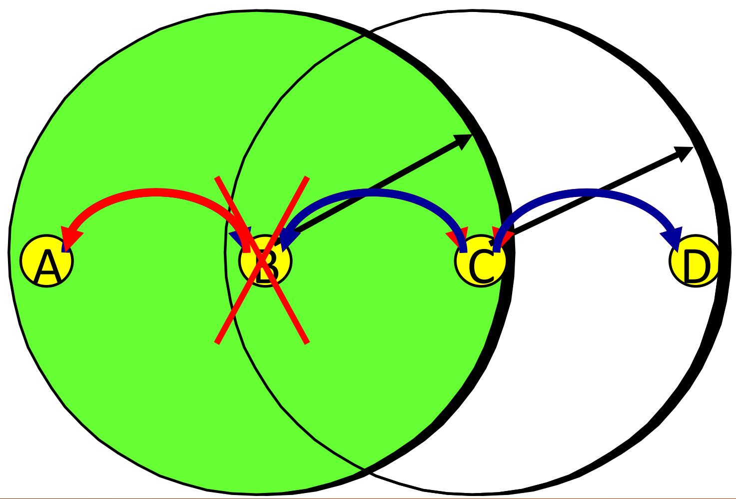
- **LANs (local area networks):** extend less than 1 km
- **IEEE 802.11** standard is designed for use in a limited geographical area, also known as ***Wi-Fi***
  - The challenge: to **share a common communication medium** – signals propagating through space
- 802.11 was designed to run over six different techniques:
  - One based on diffused **infrared** (no longer being used)
  - Two based on **spread spectrum** (**2.4** and **5.7 GHz**)
    - **DS (Direct Sequence)** and **FH (Frequency Hopping)**
    - **11 Mbps** (802.11b)
  - One based on **OFDM (Orthogonal Frequency Division multiplexing)**
    - **54 Mbps** (802.11a (**5.7 GHz**), 802.11g, (**2.4 GHz**))

Issues for Wireless Networks

- **Wireline Ethernet:** each node waits until the link becomes idle before transmitting and backs off should a collision occur
- However, this problem is **more complicated** in a **wireless network**
  - **The wireless coverage of a node is limited**
  - **Not all nodes are always within reach of each other**

Collision Avoidance

- **Hidden nodes problem:** A → B are C → B: collide with each other at B
  - Neither A nor C is aware of this collision
- **Exposed node problem:** Suppose that B is sending to A
  - Since C can hear B's transmission, it would be a mistake for C to conclude that it cannot transmit to D
  - C's transmission will not interfere with A's receiving



Collision Avoidance

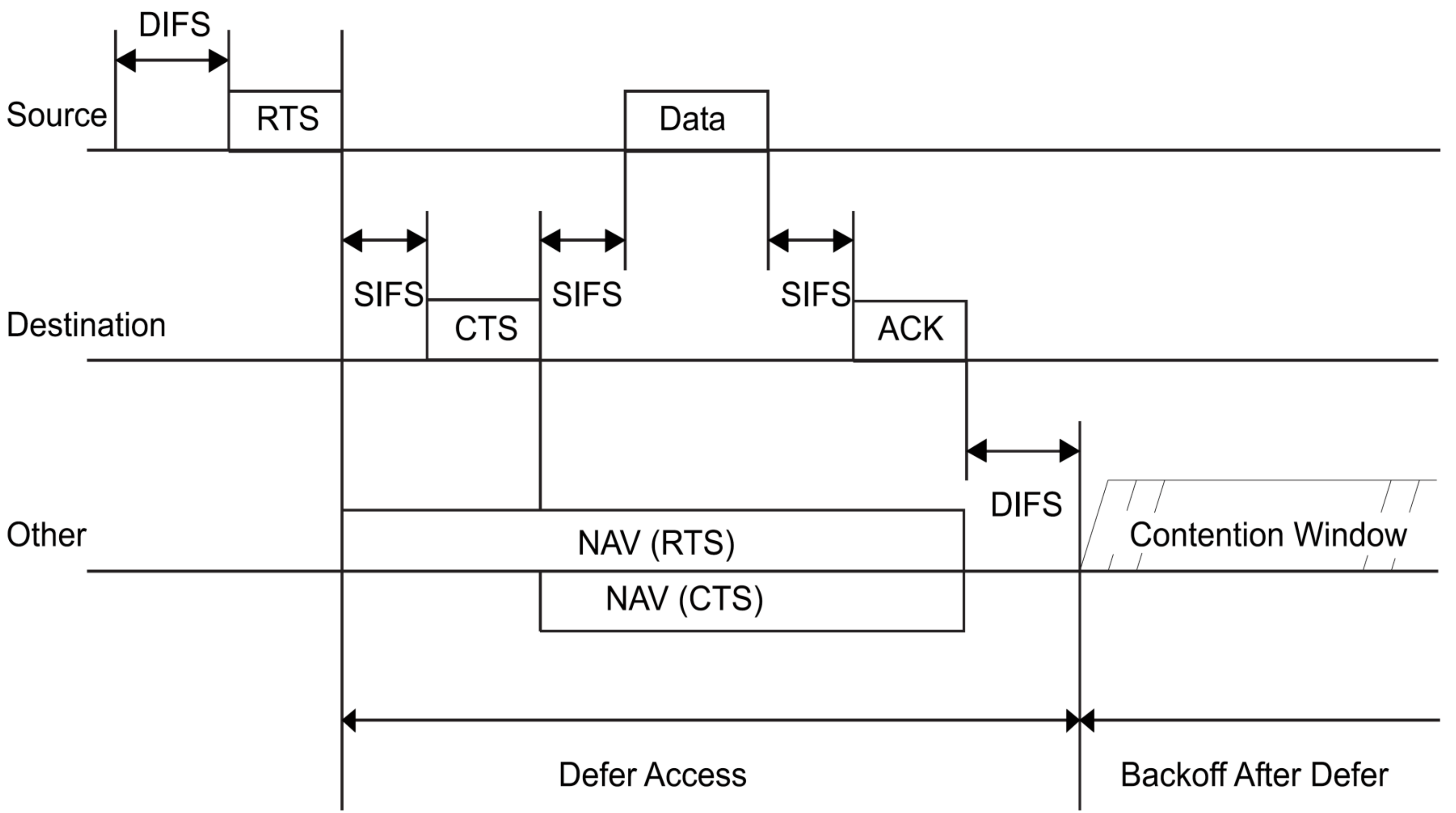
- 802.11 addresses these problems with an algorithm called **Multiple Access with Collision Avoidance (MACA)**
  - Before the sender actually transmits any data, the sender and receiver exchange **control frames** with each other
- The sender transmits a **Request to Send (RTS)** frame to the receiver
  - Includes a field that indicates **how long** the sender wants to hold the medium
- The receiver replies with a **Clear to Send (CTS)** frame
  - Echoes the **length field** back to the sender
- Any node that sees the CTS frame knows that it is close to the receiver and **cannot transmit** for the period of time

Collision Avoidance

- There are two more details for MACA
- The receiver sends an **ACK** to the sender after successfully receiving a frame
  - All nodes must **wait for this ACK** before trying to transmit
- Should two or more nodes detect an idle link and try to transmit an RTS frame at the same time
  - The RTS frames will collide with each other
  - The senders realize the collision has happened when they do not receive the CTS frame after a period of time
  - They wait a random amount of time before trying again
    - **Exponential backoff algorithm** used on the Ethernet

Collision Avoidance

- RTS/CTS/data/ACK

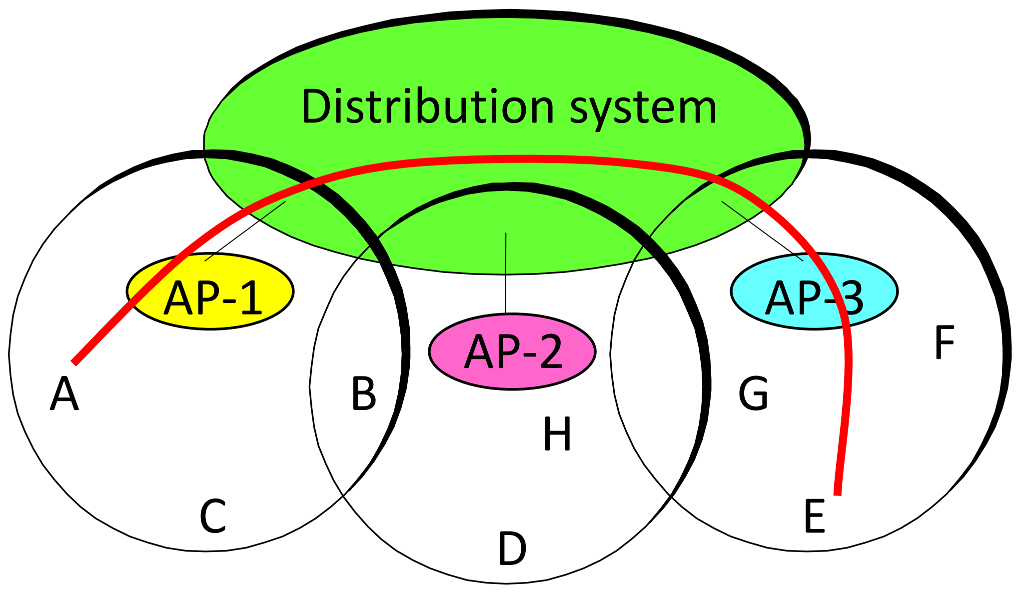


Mobile Ad Hoc Network (MANET)

- **Ad Hoc** configuration:
  - All nodes are free to move around
  - Nodes are free to **directly communicate** with each other, but in practice, they operate within the same structure
  - The connectivity depends on how far apart they are
- Nodes are also allowed to connect to a wired network
  - The connected points are called **access points (AP)**
  - APs are connected to each other by a so-called **distribution system**
  - An AP plays the same role as a **base station**
  - The distribution system could be an Ethernet or a token ring

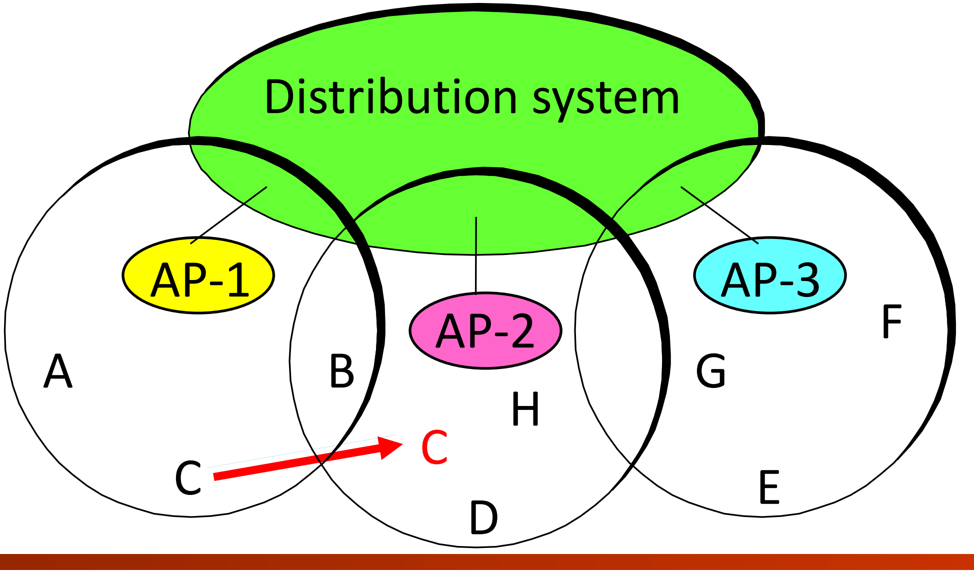
Mobile Ad Hoc Network (MANET)

- Two nodes can communicate directly with each other if they are within reach of each other
- Each node associates itself with one AP
  - A node can communicate with any node within the system
- A $\rightarrow$ E: A $\rightarrow$ AP-1 $\rightarrow$ Distribution system $\rightarrow$ AP-3 $\rightarrow$ E



Mobile Ad Hoc Network (MANET)

- A node may acquires a new AP due to **mobility**
  - The new AP notifies the old AP of the change
- **Active scanning:** the node actively searches for an new AP
- **Passive scanning:** each AP periodically sends a **Beacon** frame that advertises the capabilities of the AP
  - A node can change to a new AP based on the Beacon frame



Frame Format

- The control field contains:
  - **Type and Subtype fields (6-bit):** indicates whether it is a data frame, a RTS or CTS frame, or for other applications
  - **ToDS and FromDS:** indicate the address types
- It contains **four**, rather than two, addresses: Source Address, Destination Address, Transmitter Address, Receiver Address
- $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E$

RA	TA	DA	SA

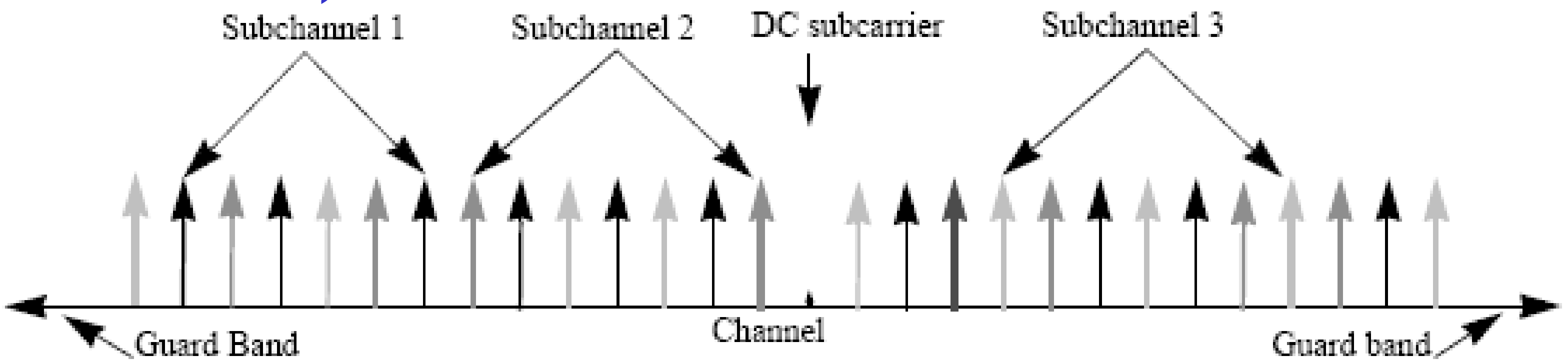
	RA	TA	DA	SA
$A \rightarrow B$	B	A	E	A
$B \rightarrow C$	C	B	E	A
$C \rightarrow D$	D	C	E	A
$D \rightarrow E$	E	D	E	A

16	16	48	48	48	16	48	0-18,496	32
Control	Duration	Addr1	Addr2	Addr3	SeqCtrl	Addr4	Payload	CRC

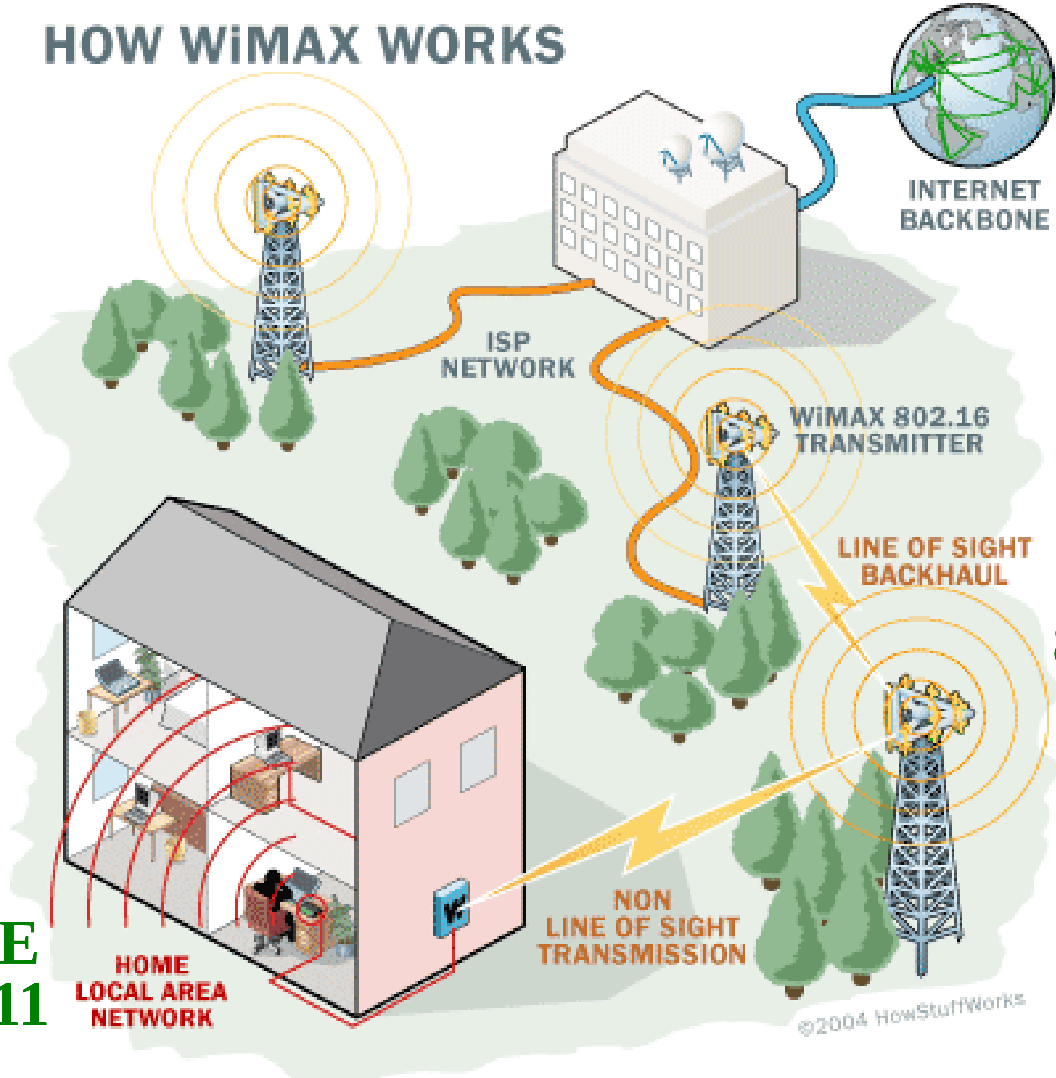
WMAN
IEEE 802.16

Physical Properties

- **MANs (metropolitan area networks):** tens of kilometers
- **WiMAX** (Worldwide Interoperability for Microwave Access)
  - **IEEE 802.16, IEEE 802.16a, IEEE 802.16e** standards
- 802.16 was designed to run over three different techniques:
  - **Single Carrier (QPSK, 16-QAM, 64-QAM)**
  - **OFDM (Orthogonal Frequency Division Multiplexing)**
  - **OFDMA (Orthogonal Frequency Division Multiple Access)**



WiMAX Systems

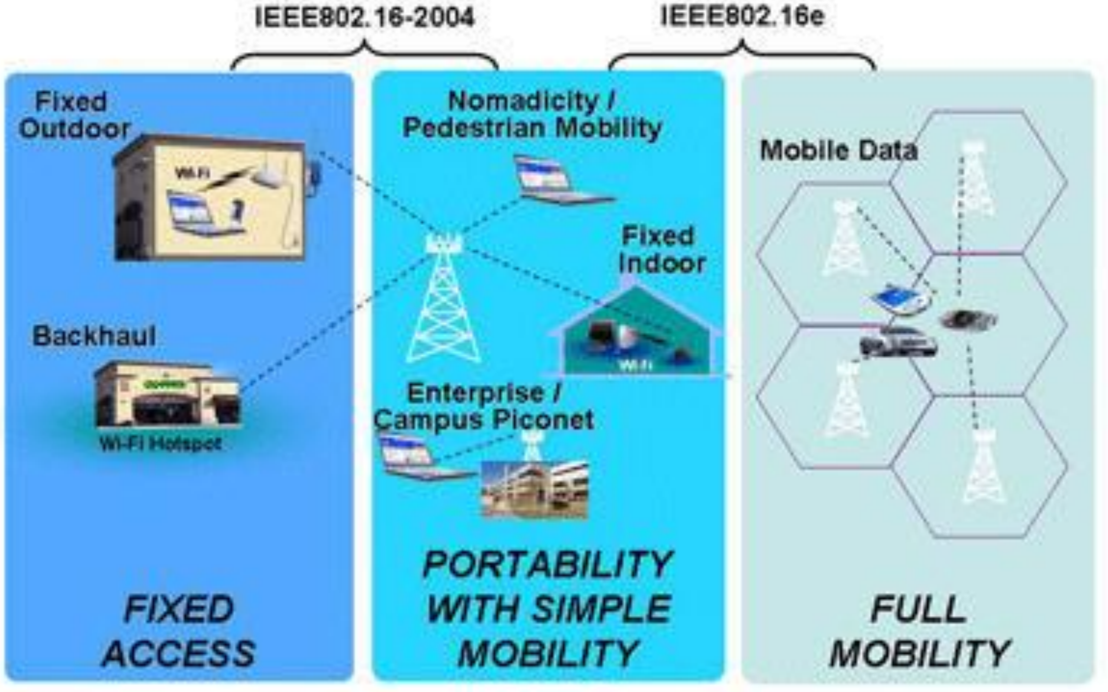


IEEE 802.16

IEEE 802.11

IEEE 802.16 Services

- **IEEE 802.16, IEEE 802.16a (IEEE 802.16-2004):**
  - Support for **fixed** services
- **IEEE 802.16e**
  - Support for **fixed** and **mobile** services



3G Cellular Systems

Evolution of Cellular Systems

- **First generation** cellular systems are all analog standards applying **FM modulation**
  - **AMPS** – US
  - **TACS, NMT** – Europe
  - **JTACS** – Japan
- **Second generation** cellular systems are all digital standards applying either **TDMA** (most) or **CDMA** (only one) technology
  - **IS-136, IS-95, PACS** – US
  - **GSM (Global System for Mobile Communications), DECT** – Europe
  - **PDC, PHS** – Japan

Evolution of Cellular Systems

- **Third generation** cellular systems are all digital standards applying **CDMA** technology
  - **cdma2000** – US
  - **W-CDMA** – Europe

Services for 2G System

- **2G Wireless**
  - **The technology of most current digital mobile phones**
  - **Features includes:**
    - Phone calls
    - Voice mail
    - Receive simple email messages
  - **Speed:** $<$ 10kb/sec
    - **Time to download a 3min MP3 song:** 31-41 min

Services for 2.5G System

- **2.5G Wireless**
  - **The best technology now widely available**
  - **Features includes:**
    - Phone calls/fax
    - Voice mail
    - Send/receive large email messages
    - Web browsing
    - Navigation/maps
  - **Speed:** 64-144kb/sec
    - **Time to download a 3min MP3 song:** 6-9min

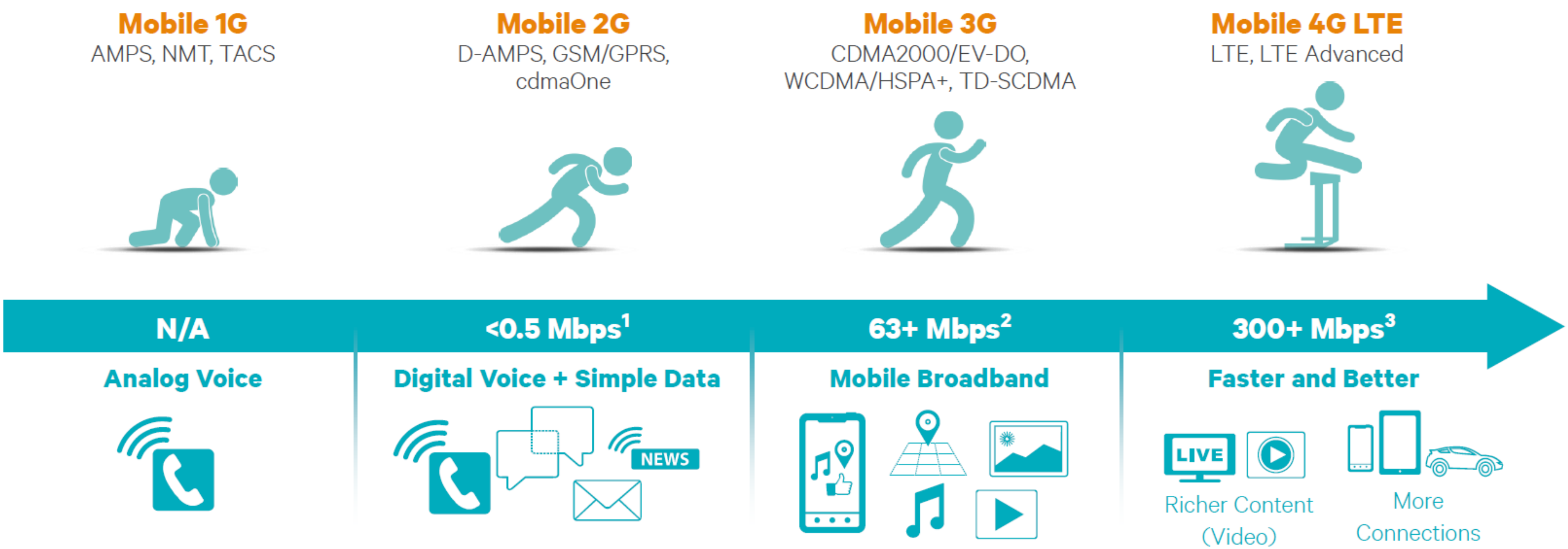
Services for 3G System

- **3G Wireless**
  - **Combines a mobile phone, laptop PC and TV**
  - **Features includes:**
    - Phone calls/fax
    - Global roaming
    - Send/receive large email messages
    - High-speed Web
    - Navigation/maps
    - Videoconferencing
    - TV streaming
  - **Speed**: 144kb/sec-2mb/sec
    - **Time to download a 3min MP3 song:** 11sec-1.5min

Radio Frequency Spectrum

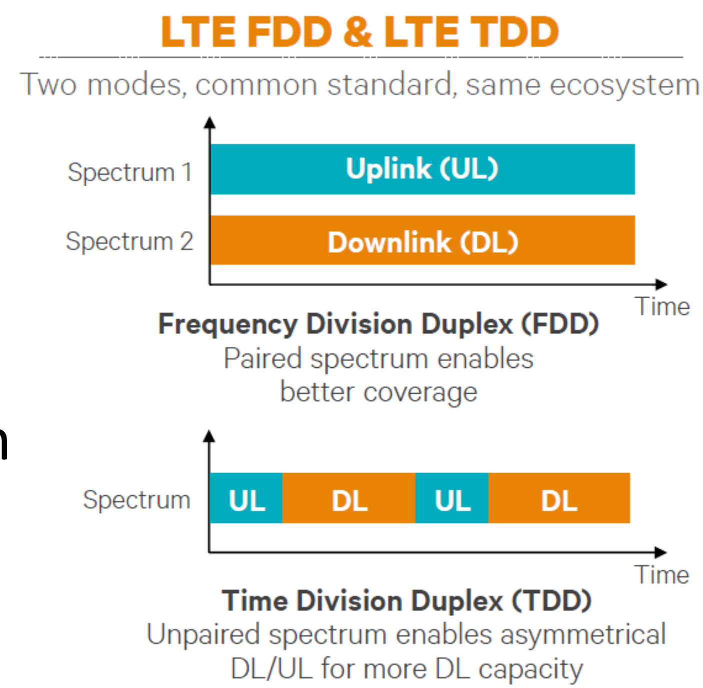
- **300MHz - 600MHz**
  - **NMT 450** Nordic Mobile Telephone System
  - **TV** Terrestrial Television, analog and digital
- **600MHz - 1.5GHz**
  - **GSM900** Global System for Mobile Communications
  - **GRRS** General Packet Radio Service
  - **CT-1** Cordless Telephone
  - **GPS** Global Positioning System
- **1.5GHz - 3GHz**
  - **GSM1800** Global System for Mobile Communications
  - **DECT** Digital Enhanced Cordless Telephone System
  - **3G UMTS** Univeral Mobile Telecommunications System
  - **WLAN** Wireless Local Area Network
  - **Bluetooth**
  - **Microwave Oven**
- **3GHz - 6GHz**
  - **Hiperlan** High Performance Local Area Network
  - **FWA** Fixed Wireless Access (Microwave communication systems)

Mobile 4G LTE is evolving to provide more data capacity



4G LTE

- Long Term Evolution (LTE)
- Advantages of LTE
    - High throughput
    - Low latency
    - FDD and TDD in the same platform
    - Superior end-user experience
    - Seamless Connection
    - Plug and play
    - Simple architecture



LTE Basic Parameters (1)

- Frequency range: UMTS FDD bands and TDD bands defined in 36.101(v860) Table, given below
- Duplexing : FDD, TDD, half-duplex FDD
- Channel coding : Turbo code
- Mobility: 350 km/h
- Channel Bandwidth (MHz): 1.4, 3, 5, 10, 15, 20
- Transmission Bandwidth Configuration NRB : (1 resource block = 180kHz in 1ms TTI): 6, 15, 25, 50, 75, 100
- Modulation Schemes
  - UL: QPSK, 16QAM, 64QAM(optional)
  - DL: QPSK, 16QAM, 64QAM
- Multiple Access Schemes
  - UL: SC-FDMA (Single Carrier Frequency Division Multiple Access) supports 50Mbps+ (20MHz spectrum)
  - DL: OFDM (Orthogonal Frequency Division Multiple Access) supports 100Mbps+ (20MHz spectrum)

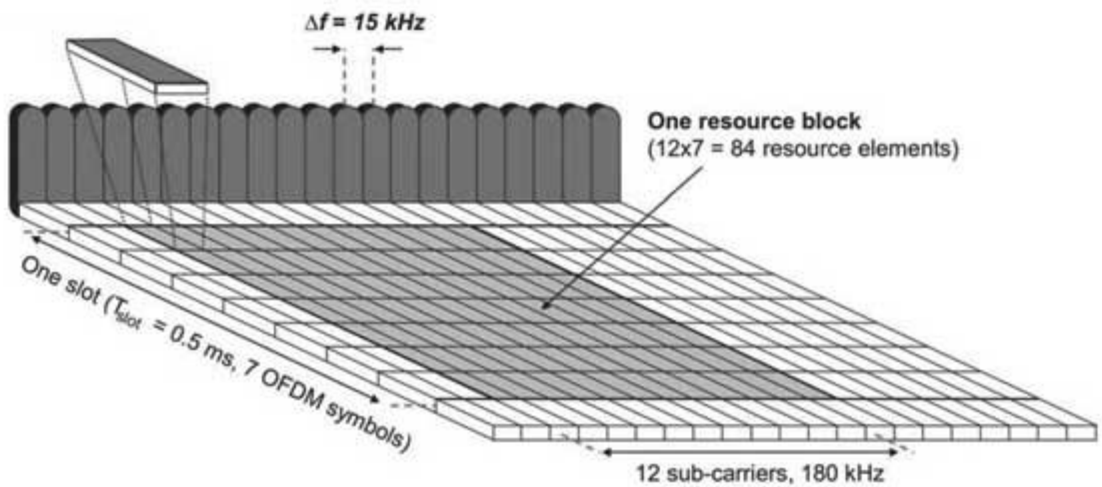
LTE Basic Parameters (2)

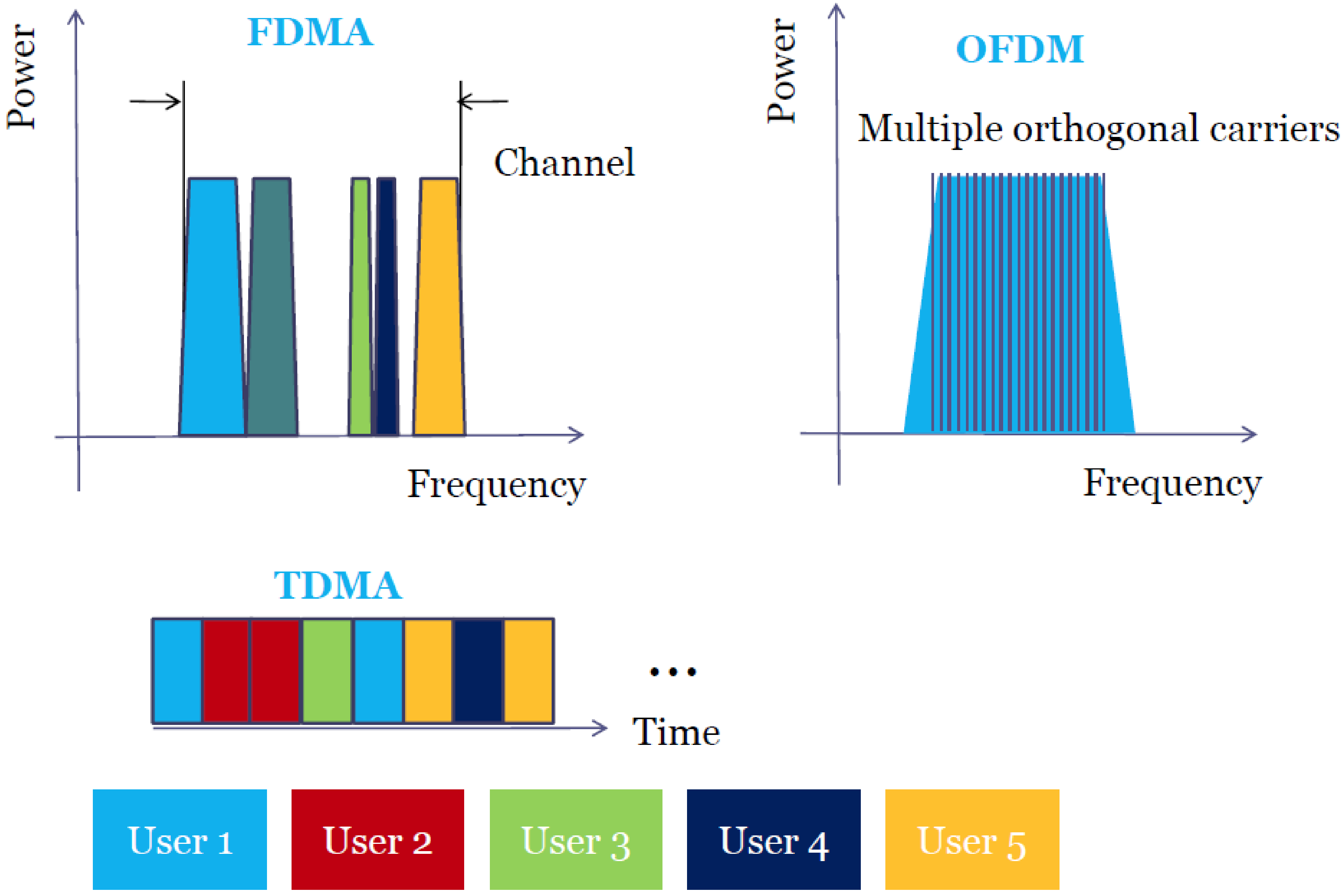
- Multi-Antenna Technology
  - UL: Multi-user collaborative MIMO
  - DL: TxAA, spatial multiplexing, CDD, max 4x4 array
- Peak data rate in LTE
  - UL: 75Mbps(20MHz bandwidth)
  - DL: 150Mbps(UE Category 4, 2x2 MIMO, 20MHz bandwidth)
  - DL: 300Mbps(UE category 5, 4x4 MIMO, 20MHz bandwidth)
- MIMO (Multiple Input Multiple Output)
  - UL: 1 x 2, 1 x 4
  - DL: 2 x 2, 4 x 2, 4 x 4
- Coverage: 5 - 100km with slight degradation after 30km
- QoS: E2E QOS allowing prioritization of different class of service
- Latency: End-user latency < 10mS

	700Mhz	900Mhz	1800Mhz	2600Mhz	CA載波聚合
中華電信	X	O	O	O	3
台灣大哥大	O	X	O	X	2
遠傳電信	O	X	O	B7(FDD) B41(TDD)	3
台灣之星	X	O	X	O	2
亞太電信	O	X	X	B41(TDD)	X
GT4G R (台哥大訊號)	O	X	O	X	2

LTE OFDM Technology

- Orthogonal Frequency Division Multiplexing (OFDM)
- To overcome the effect of multi path fading problem available in UMTS

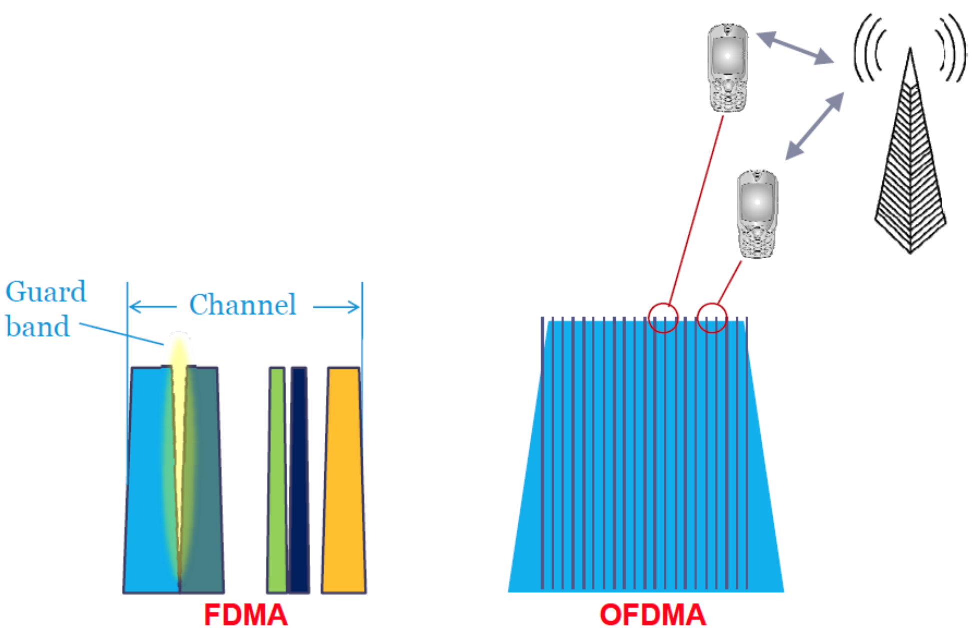


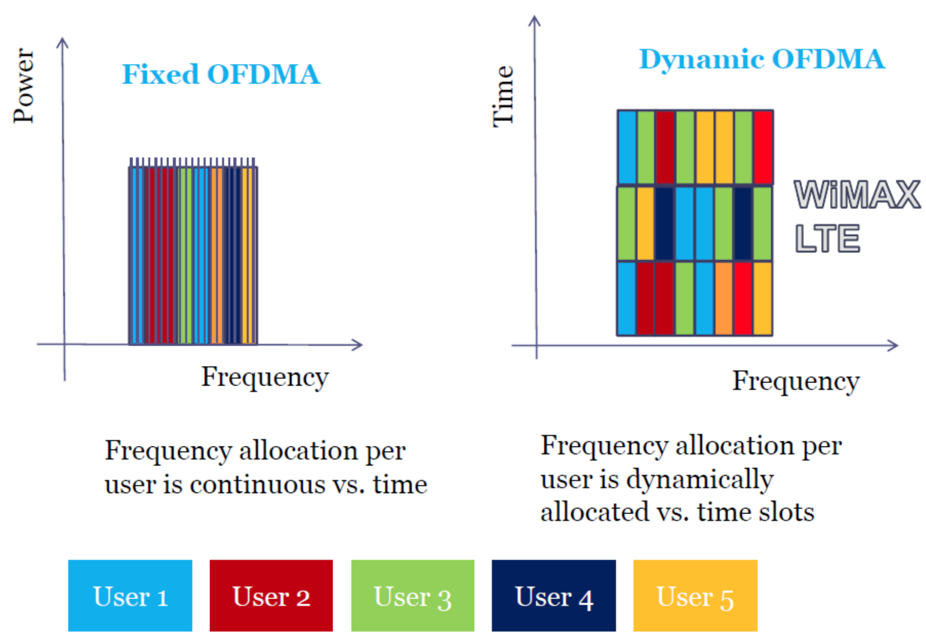

FDMA
Power
Channel
Frequency
OFDM
Power
Multiple orthogonal carriers
Frequency
TDMA
...
Time
User 1
User 2
User 3
User 4
User 5

FDMA v.s. OFDMA

OFDMA is more frequency efficient than FDMA

- Each station is assigned a set of subcarriers, eliminating frequency guard bands between users

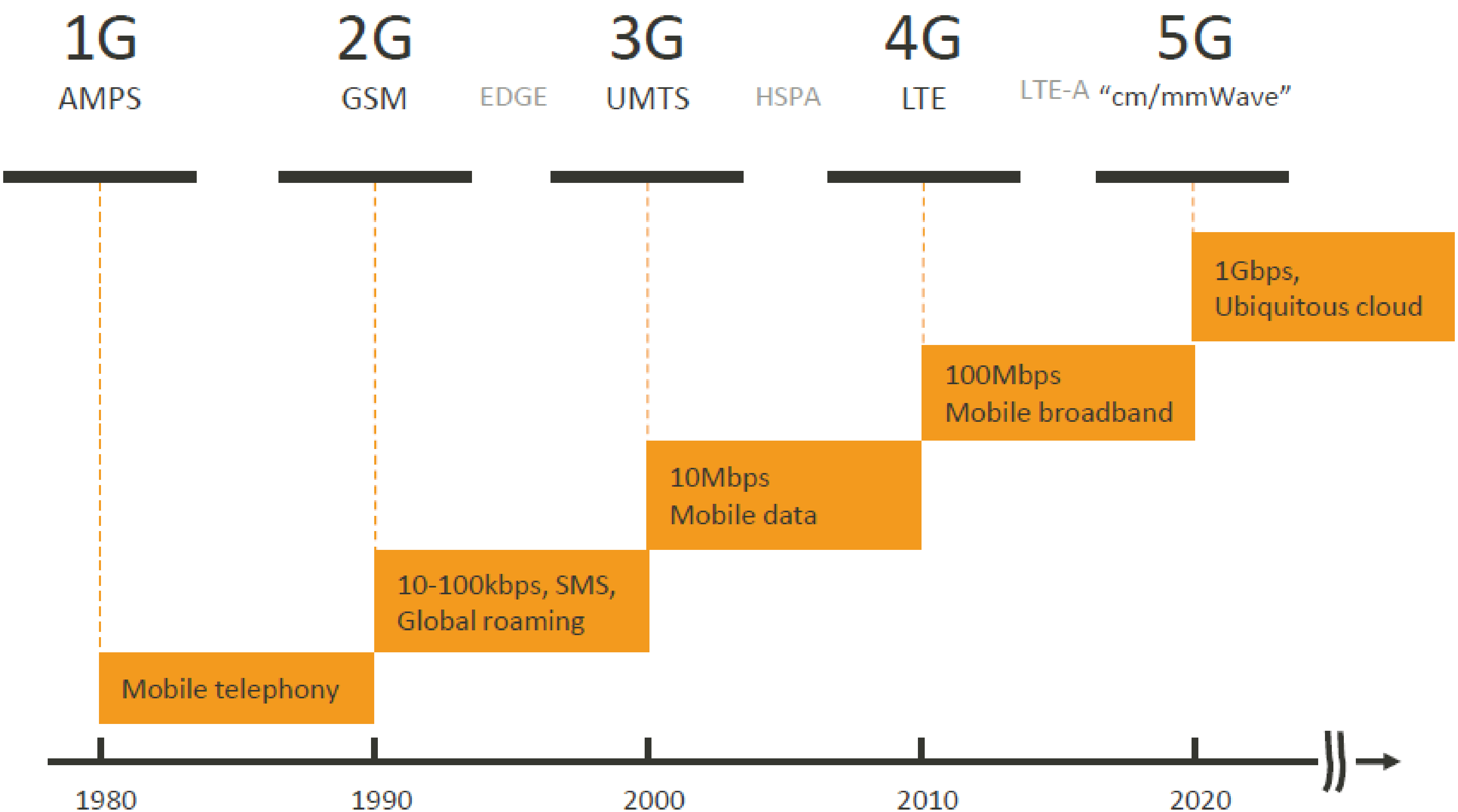




Advantages of OFDM

- Ability to cope with severe channel conditions without complex equalization filters
- Channel equalization is simplified
- Possible to eliminate inter symbol interference (ISI)

Cellular Evolution toward 5G



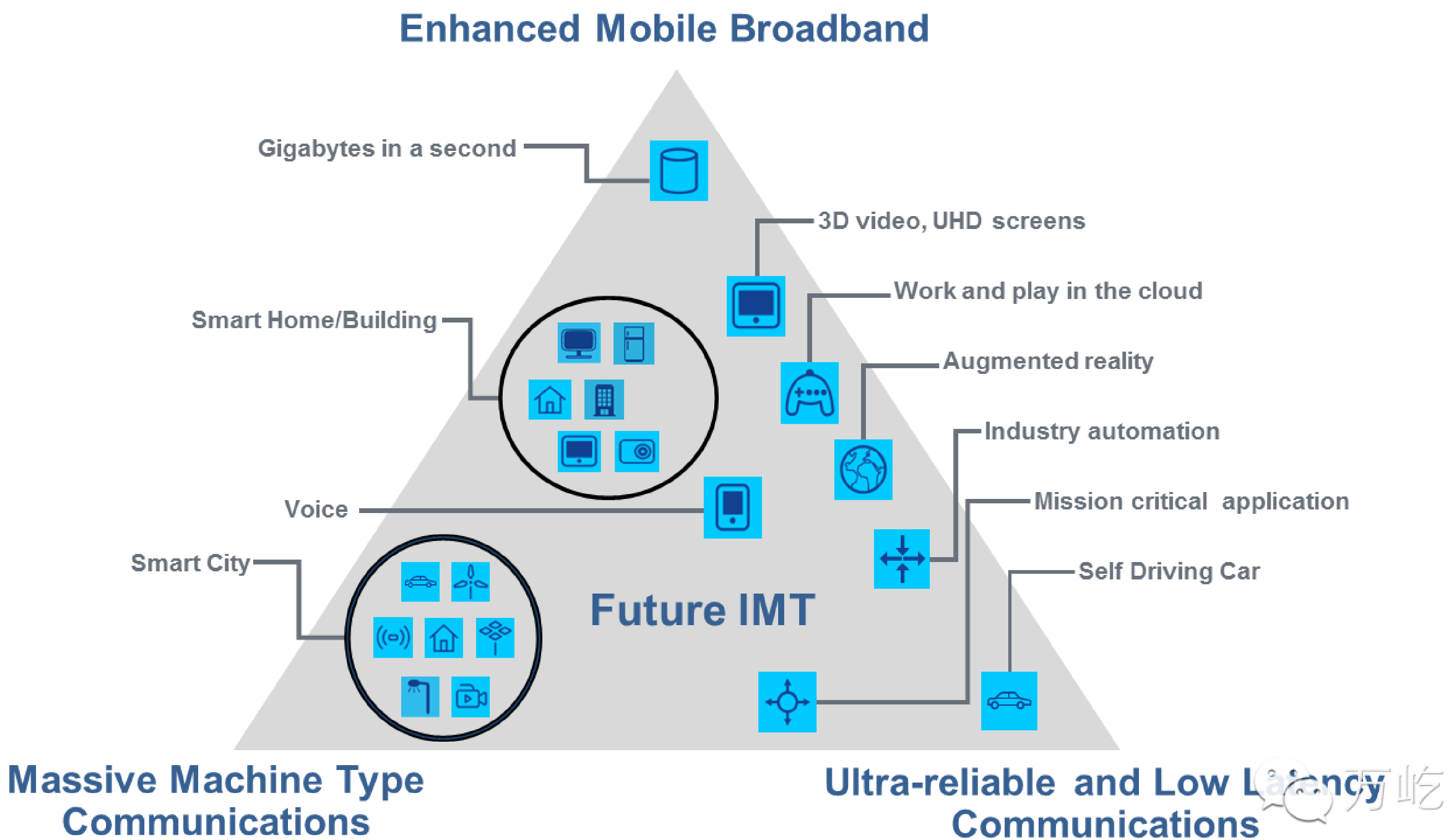
IMT-2020 in ITU and 5G in 3GPP

- The International Telecommunications Union (ITU)
    - ITU-Radiocommunication (ITU-R)
- 3GPP
- IMT-2020 as known as 5G

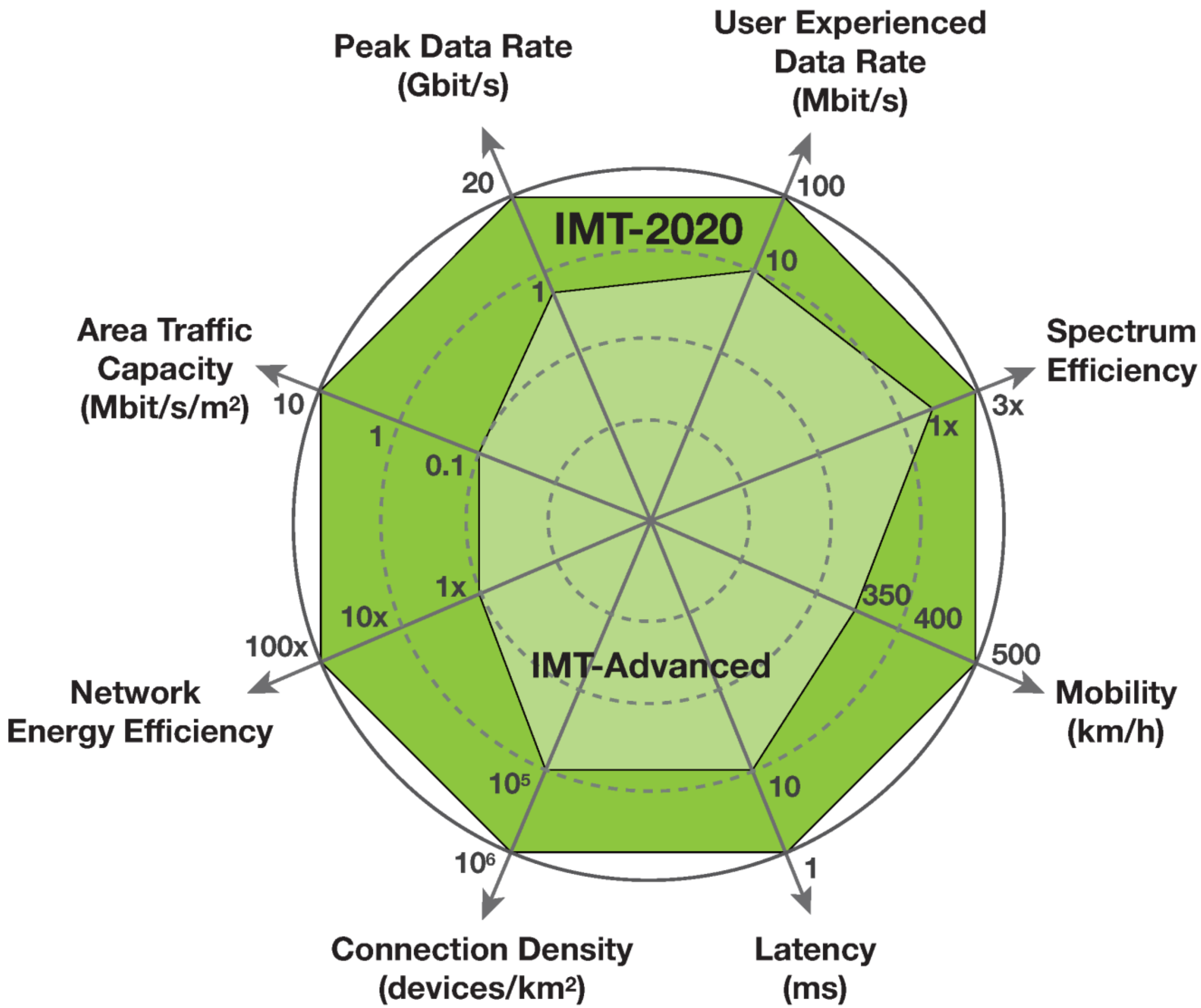
THE ITU-R “VISION” TOWARD 5G

- Enhanced Mobile Broadband
- Ultra-reliable and low latency communications
- Massive machine-type communications

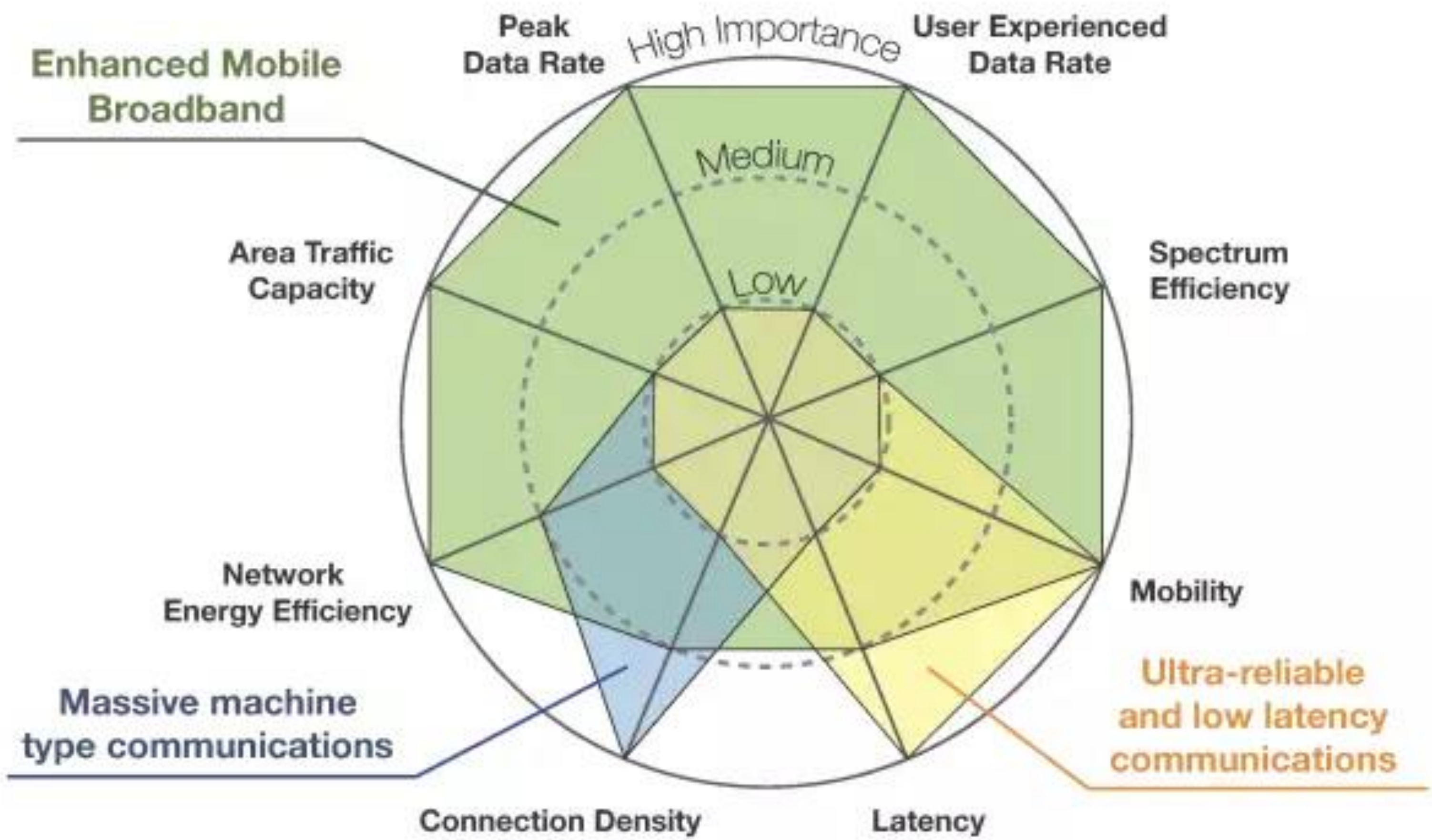
Enhanced Mobile Broadband Usage Scenarios



Enhancement of Key Capabilities from IMT-Advanced to IMT-2020



The Importance of Key Capabilities in Different Usage Scenarios



3GPP

- 3GPP submission to IMT-2020 (aka 5G) will include
  - “New Radio of 5G”, aka NR
  - LTE
- Some of the key outcomes of the 5G RAN workshop were:
  - The identification of three high level use cases for 5G
  - A consensus that there will be a new, non-backward compatible, radio access technology as part of 5G
  - Spectrum wise, both sub-6GHz and above 6 GHz (i.e., including mmWave) should be supported