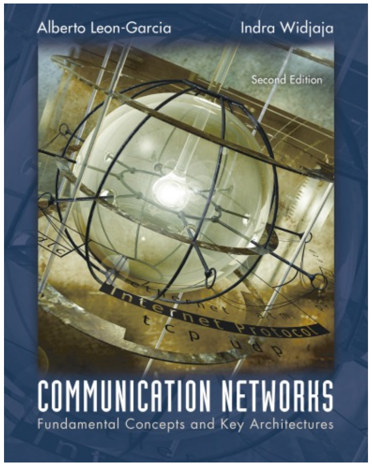
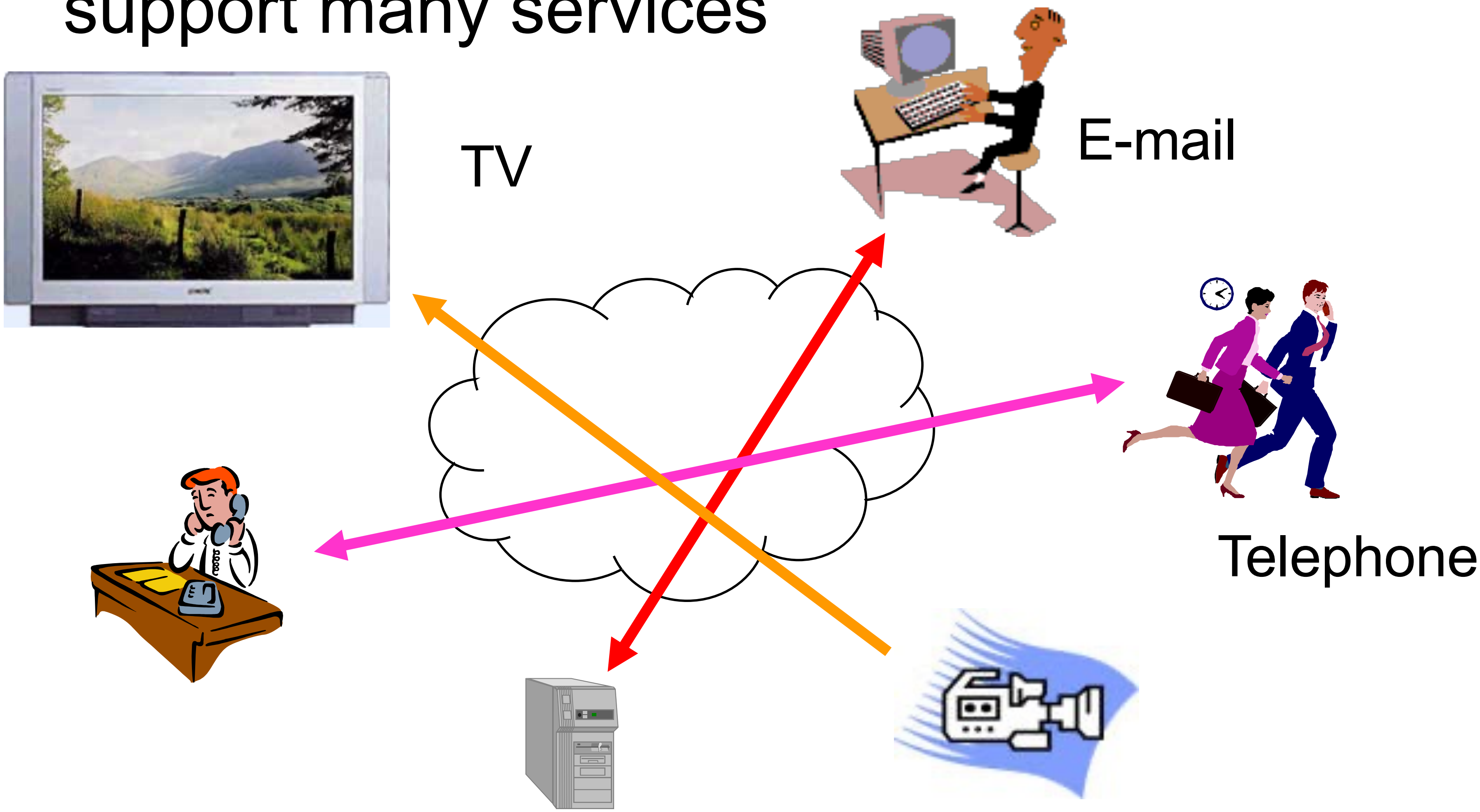
Chapter 3
Digital Transmission Fundamentals



Digital Representation of Information
Why Digital Communications?
Digital Representation of Analog Signals
Characterization of Communication Channels
Fundamental Limits in Digital Transmission
Line Coding
Modems and Digital Modulation
Properties of Media and Digital Transmission Systems
Error Detection and Correction

Digital Networks

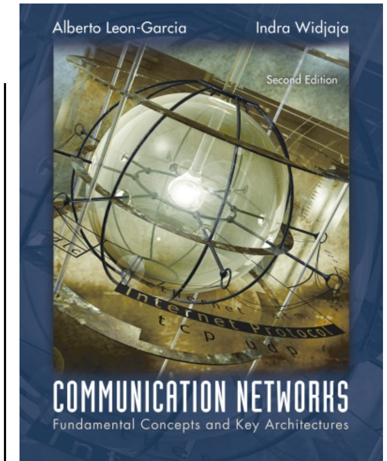
- Digital transmission enables networks to support many services



Questions of Interest

- How long will it take to transmit a message?
  - How many bits are in the message (text, image)?
  - How fast does the network/system transfer information?
- Can a network/system handle a voice (video) call?
  - How many bits/second does voice/video require? At what quality?
- How long will it take to transmit a message without errors?
  - How are errors introduced?
  - How are errors detected and corrected?
- What transmission speed is possible over radio, copper cables, fiber, infrared, …?

Chapter 3
Digital Transmission Fundamentals



Digital Representation of Information

Bits, numbers, information

- Bit: number with value 0 or 1
  - n bits: digital representation for 0, 1, … , 2^n
  - Byte or Octet, $n = 8$
  - Computer word, $n = 16$, 32, or 64
- n bits allows enumeration of 2^n possibilities
  - n-bit field in a header
  - n-bit representation of a voice sample
  - Message consisting of n bits
- *The number of bits required to represent a message is a measure of its information content*
  - More bits → More content

Block vs. Stream Information

Block

- Information that occurs in a single block
  - Text message
  - Data file
  - JPEG image
  - MPEG file
- Size = Bits / block

  or bytes/block
  - 1 kbyte = 2^{10} bytes
  - 1 Mbyte = 2^{20} bytes
  - 1 Gbyte = 2^{30} bytes

Stream

- Information that is produced & transmitted *continuously*
  - Real-time voice
  - Streaming video
- Bit rate = bits / second
  - 1 kbps = 10^3 bps
  - 1 Mbps = 10^6 bps
  - 1 Gbps = 10^9 bps

Total Transmission Time (one-way)

- L — number of bits in message
- R bps — transmission speed of digital system
- L/R — time to transmit the information
- t_{prop} = d/c — time for signal to propagate across medium
- d — distance in meters
- c — speed of signal impulse (meters per second)

$$Time = t_{prop} + L/R = d/c + L/R \text{ seconds}$$

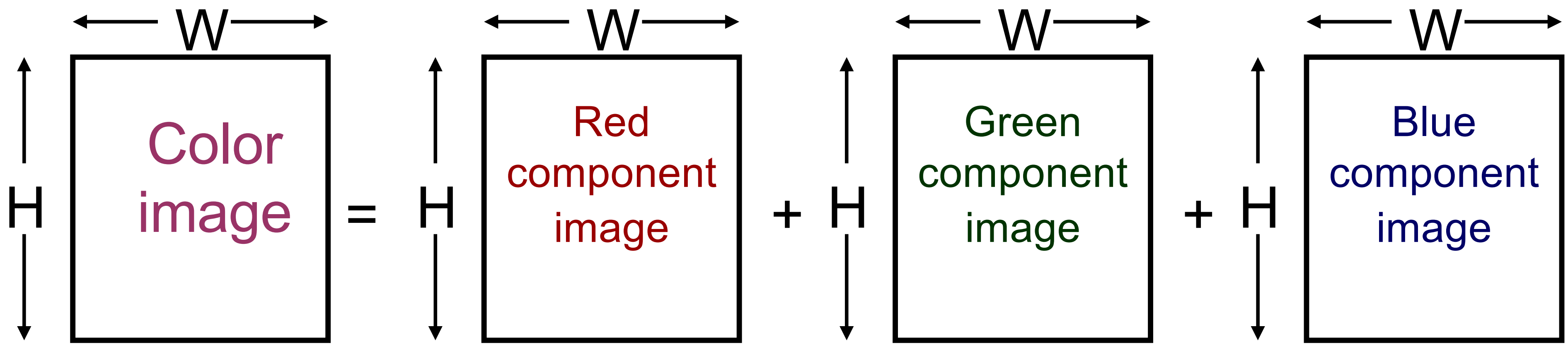
Use data compression to reduce L
Use higher speed modem to increase R
Place server closer to reduce d

Compression

- Information usually not represented efficiently
- Data compression algorithms
  - Represent the information using fewer bits
  - Lossless: original information recovered exactly
    - E.g. `zip`, `compress`, GIF, fax
  - Lossy: recover information approximately
    - JPEG
    - Tradeoff: # bits vs. quality
- Compression Ratio

  #bits (original file) / #bits (compressed file)

Color Image



Total bits = 3 × H × W pixels × B bits/pixel = 3HWB bits

Example: 8×10 inch picture at 400 × 400 pixels per inch^2

400 × 400 × 8 × 10 = 12.8 million pixels

8 bits/pixel/color

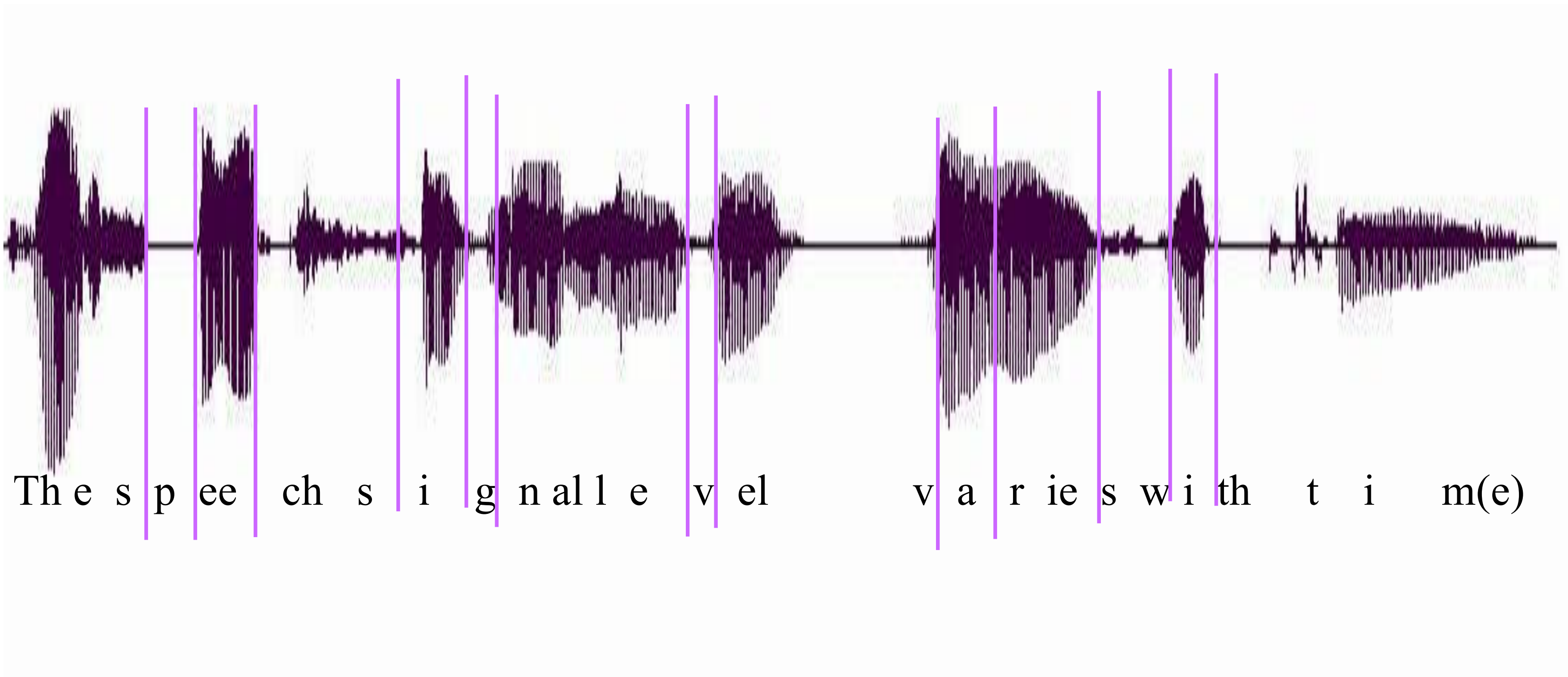
12.8 megapixels × 3 bytes/pixel = 38.4 megabytes

Examples of Block Information

Type	Method	Format	Original	Compressed (Ratio)
Text	Zip, compress	ASCII	Kbytes-Mbytes	(2-6)
Fax	CCITT Group 3	A4 page 200x100 pixels/in^2	256 kbytes	5-54 kbytes (5-50)
Color Image	JPEG	8x10 in^2 photo 400^2 pixels/in^2	38.4 Mbytes	1-8 Mbytes (5-30)

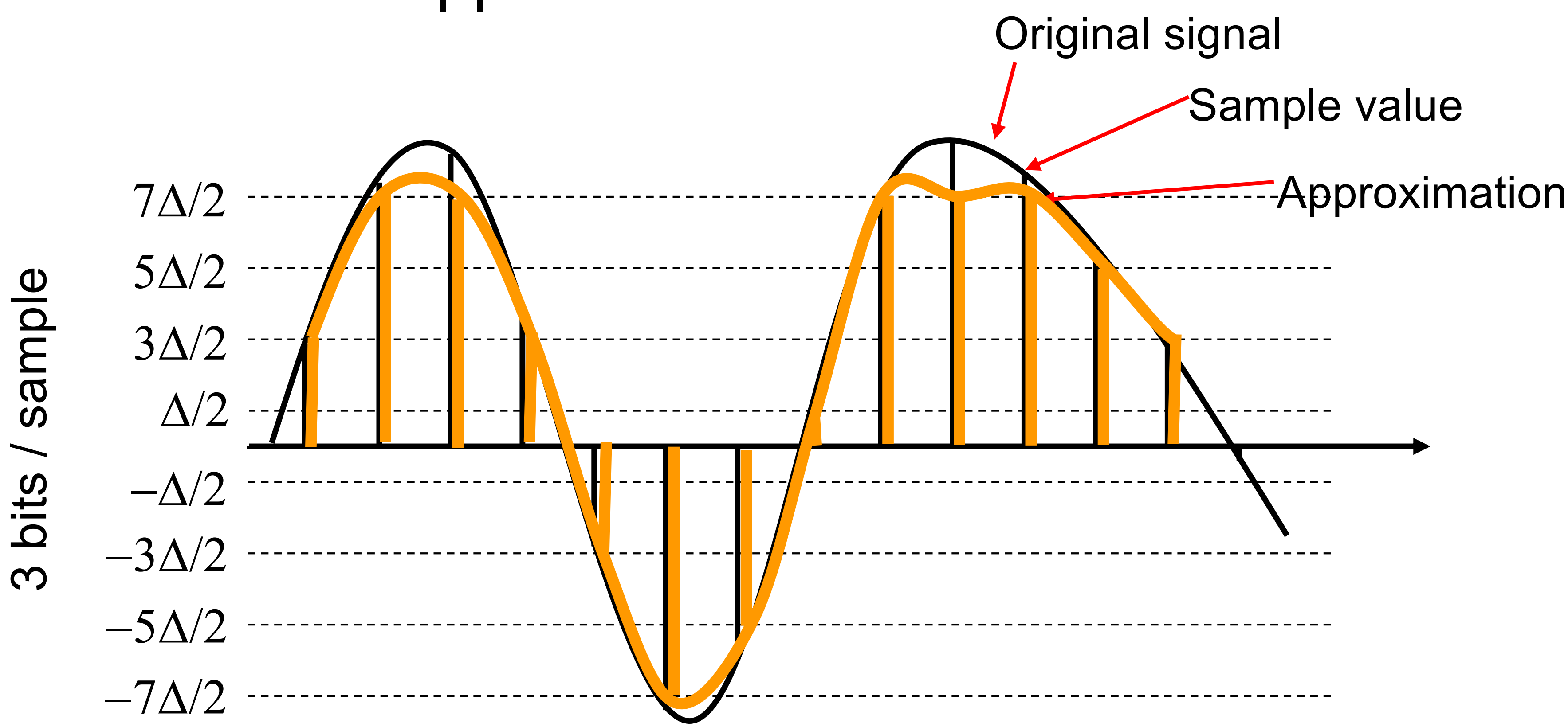
Stream Information

- A real-time voice signal must be digitized & transmitted as it is produced
- Analog signal level varies continuously in time



Digitization of Analog Signal

- Sample analog signal in time and amplitude
- Find closest approximation



R_s = Bit rate = # bits/sample x # samples/second

Bit Rate of Digitized Signal

- Bandwidth W_s Hertz: how fast the signal changes
  - Higher bandwidth → more frequent samples
  - Minimum sampling rate = 2 x W_s

- Representation accuracy: range of approximation error
  - Higher accuracy

  → smaller spacing between approximation values

  → more bits per sample

Example: Voice & Audio

Telephone voice

- W_s = 4 kHz → 8000 samples/sec
- 8 bits/sample
- R_s=8 x 8000 = 64 kbps

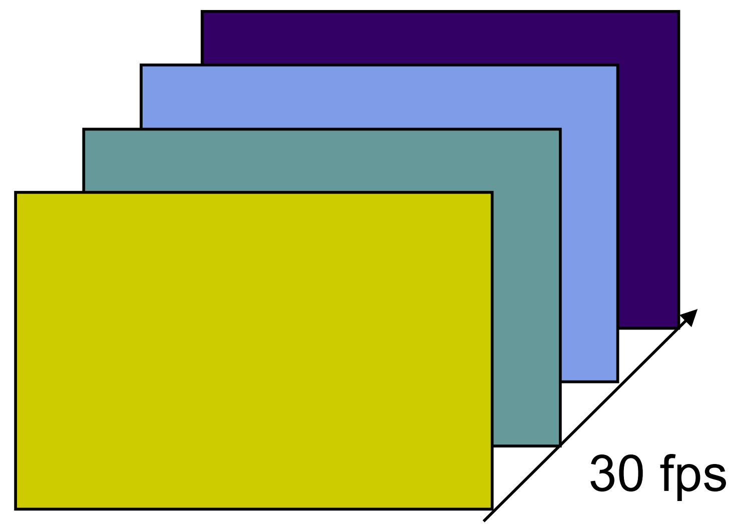
- Cellular phones use more powerful compression algorithms: 8-12 kbps

CD Audio

- W_s = 22 kHertz → 44000 samples/sec
- 16 bits/sample
- R_s=16 x 44000= 704 kbps per audio channel
- MP3 uses more powerful compression algorithms: 50 kbps per audio channel

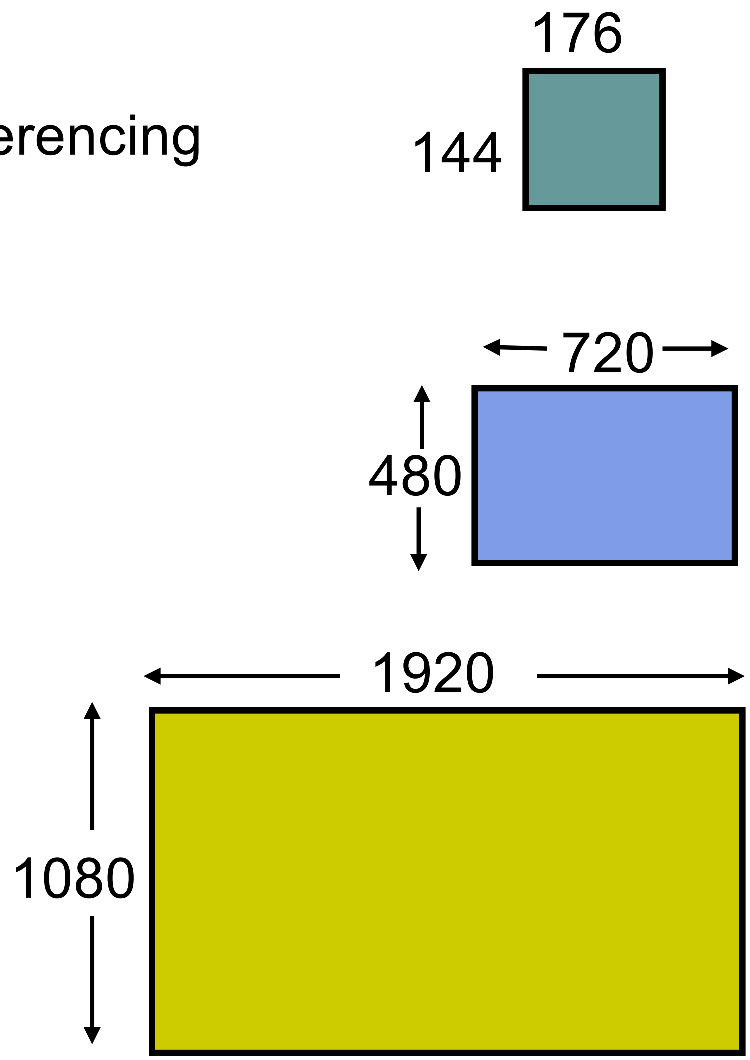
Video Signal

- Sequence of picture frames
  - Each picture digitized & compressed
- Frame repetition rate
  - 10-30-60 frames/second depending on quality
- Frame resolution
  - Small frames for videoconferencing
  - Standard frames for conventional broadcast TV
  - HDTV frames



Rate = M bits/pixel **x** (WxH) pixels/frame **x** F frames/second

Video Frames



QCIF videoconferencing

at 30 frames/sec =

760,000 pixels/sec

Broadcast TV

at 30 frames/sec =

10.4×10^6 pixels/sec

HDTV

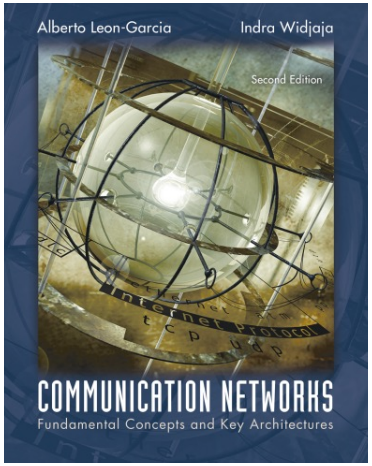
at 30 frames/sec =

67×10^6 pixels/sec

Digital Video Signals

Type	Method	Format	Original	Compressed
Video Confer-ence	H.261	176x144 or 352x288 pix @10-30 fr/sec	2-36 Mbps	64-1544 kbps
Full Motion	MPEG 2	720x480 pix @30 fr/sec	249 Mbps	2-6 Mbps
HDTV	MPEG 2	1920x1080 @30 fr/sec	1.6 Gbps	19-38 Mbps

Chapter 3
Communication Networks and Services



Why Digital Communications?

A Transmission System

Transmitter

- Converts information into *signal* suitable for transmission
- Injects energy into communications medium or channel
  - Telephone converts voice into electric current
  - Modem converts bits into tones

Receiver

- Receives energy from medium
- Converts received signal into form suitable for delivery to user
  - Telephone converts current into voice
  - Modem converts tones into bits

Transmission Impairments

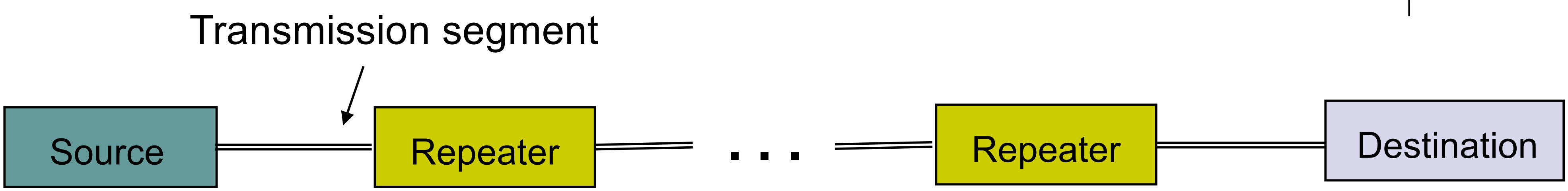
Communication Channel

- Pair of copper wires
- Coaxial cable
- Radio
- Light in optical fiber
- Light in air
- Infrared

Transmission Impairments

- Signal attenuation
- Signal distortion
- Noise
  - Thermal noise
  - Impulse noise
  - Crosstalk
  - Intermodulation noise

Analog Long-Distance Communications



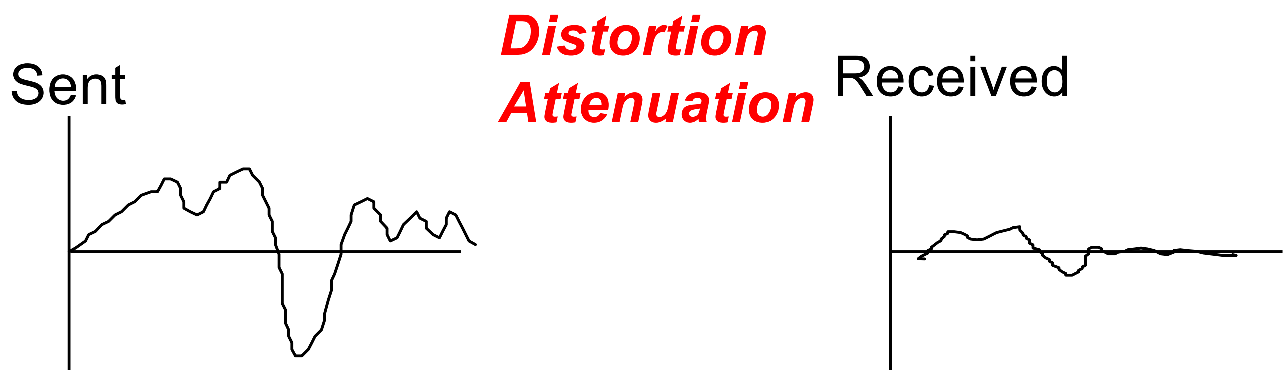
- Each repeater attempts to restore analog signal to its original form
- Restoration is imperfect
  - Distortion is not completely eliminated
  - Noise & interference is only partially removed
- Signal quality decreases with # of repeaters
- Communications is distance-limited
- Still used in analog cable TV systems
- Analogy: Copy a song using a cassette recorder

Signal Impairments

- Attenuation – the loss in signal power as a signal is transferred across a medium, usually expressed in dB.
- Distortion – the alteration of signal due to the physical attributes of the medium as well as the signal itself.
- Noise
  - Thermal – electronic noise generated by thermal agitation
  - Impulse – sudden, random, unpredictable noise bursts
  - Crosstalk – one signal creates an undesirable effect in another signal
  - Intermodulation – unwanted alteration of signals containing two or more different frequencies

Analog vs. Digital Transmission

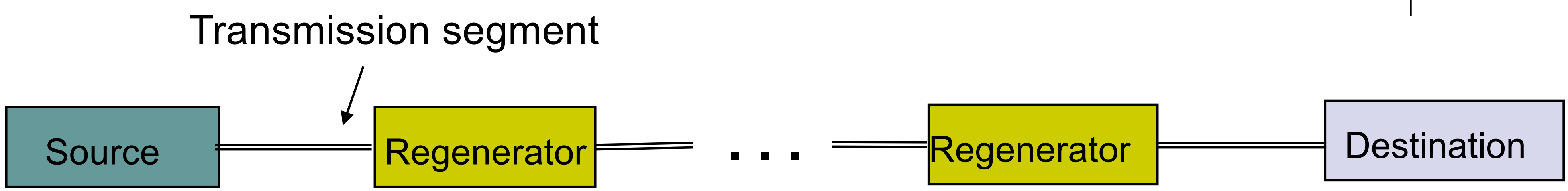
Analog transmission: all details must be reproduced accurately



Digital transmission: only discrete levels need to be reproduced

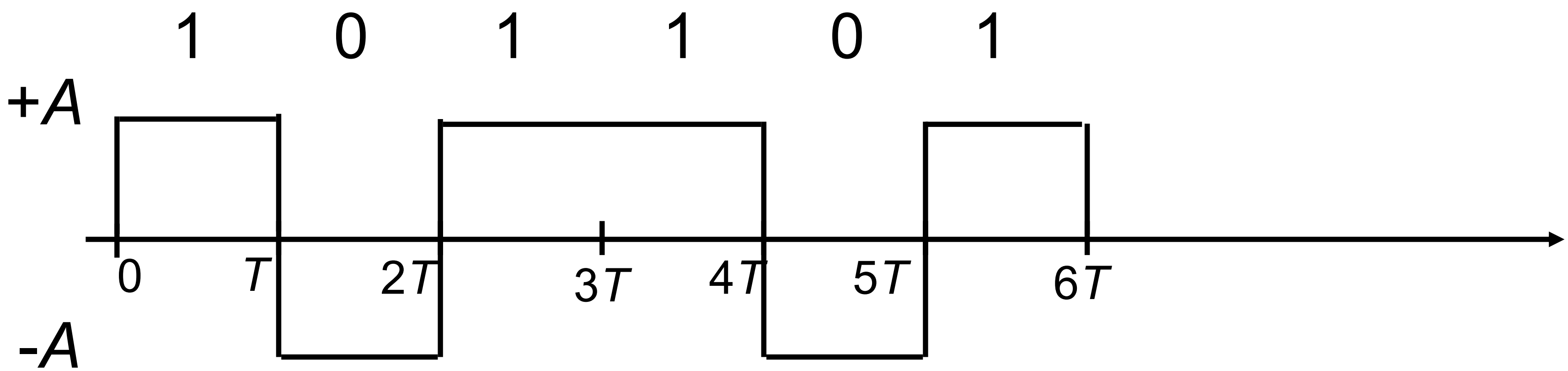
Simple Receiver: Was original pulse positive or negative?

Digital Long-Distance Communications



- Regenerator recovers original data sequence and retransmits on next segment
- Can design so error probability is very small
- Then each regeneration is like the first time!
- Analogy: copy an MP3 file
- Communications is possible over very long distances
- Digital systems vs. analog systems
  - Less power, longer distances, lower system cost
  - Monitoring, multiplexing, coding, encryption, protocols…

Digital Binary Signal



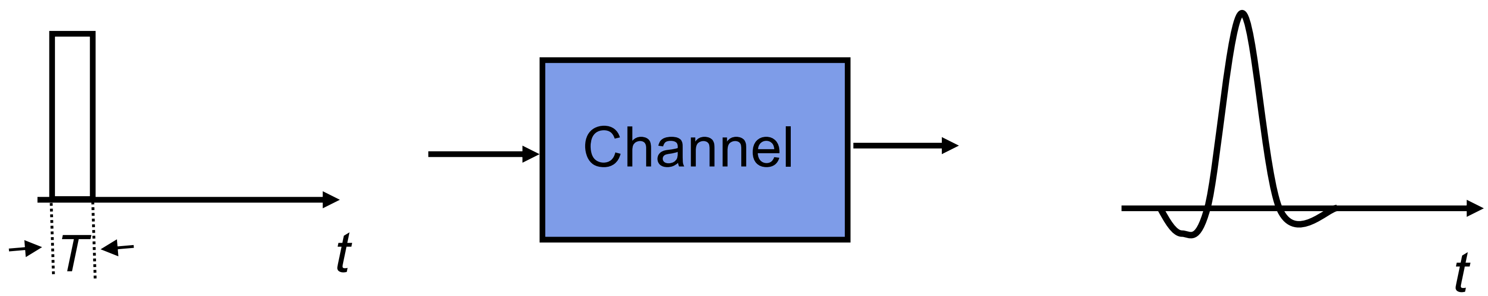
Bit rate = 1 bit / T seconds

For a given communications medium:

- How do we increase transmission speed?
- How do we achieve reliable communications?
- Are there limits to speed and reliability?

Pulse Transmission Rate

- Objective: Maximize pulse rate through a channel, that is, make T as small as possible



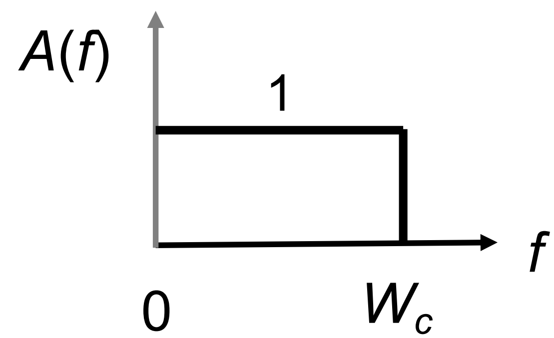
- If input is a narrow pulse, then typical output is a spread-out pulse with ringing
- Question: How frequently can these pulses be transmitted without interfering with each other?
- Answer: $2 \times W_c$ pulses/second

where W_c is the bandwidth of the channel

Bandwidth of a Channel

$X(t) = a \cos(2\pi ft) \longrightarrow$ Channel $\longrightarrow Y(t) = A(f)\, a \cos(2\pi ft)$

- If input is sinusoid of frequency f, then
  - output is a sinusoid of same frequency f
  - Output is attenuated by an amount A(f) that depends on f
  - A(f)≈1, then input signal passes readily
  - A(f)≈0, then input signal is blocked
- **Bandwidth W_c is range of frequencies passed by channel**



Ideal low-pass channel

Multilevel Pulse Transmission

- Assume channel of bandwidth W_c, and transmit 2 W_c pulses/sec (without interference)
- If pulses amplitudes are either $-A$ or $+A$, then each pulse conveys 1 bit, so

  Bit Rate = 1 bit/pulse x 2W_c pulses/sec = 2W_c bps
- If amplitudes are from $\{-A, -A/3, +A/3, +A\}$, then bit rate is 2 x 2W_c bps
- By going to $M = 2^m$ amplitude levels, we achieve

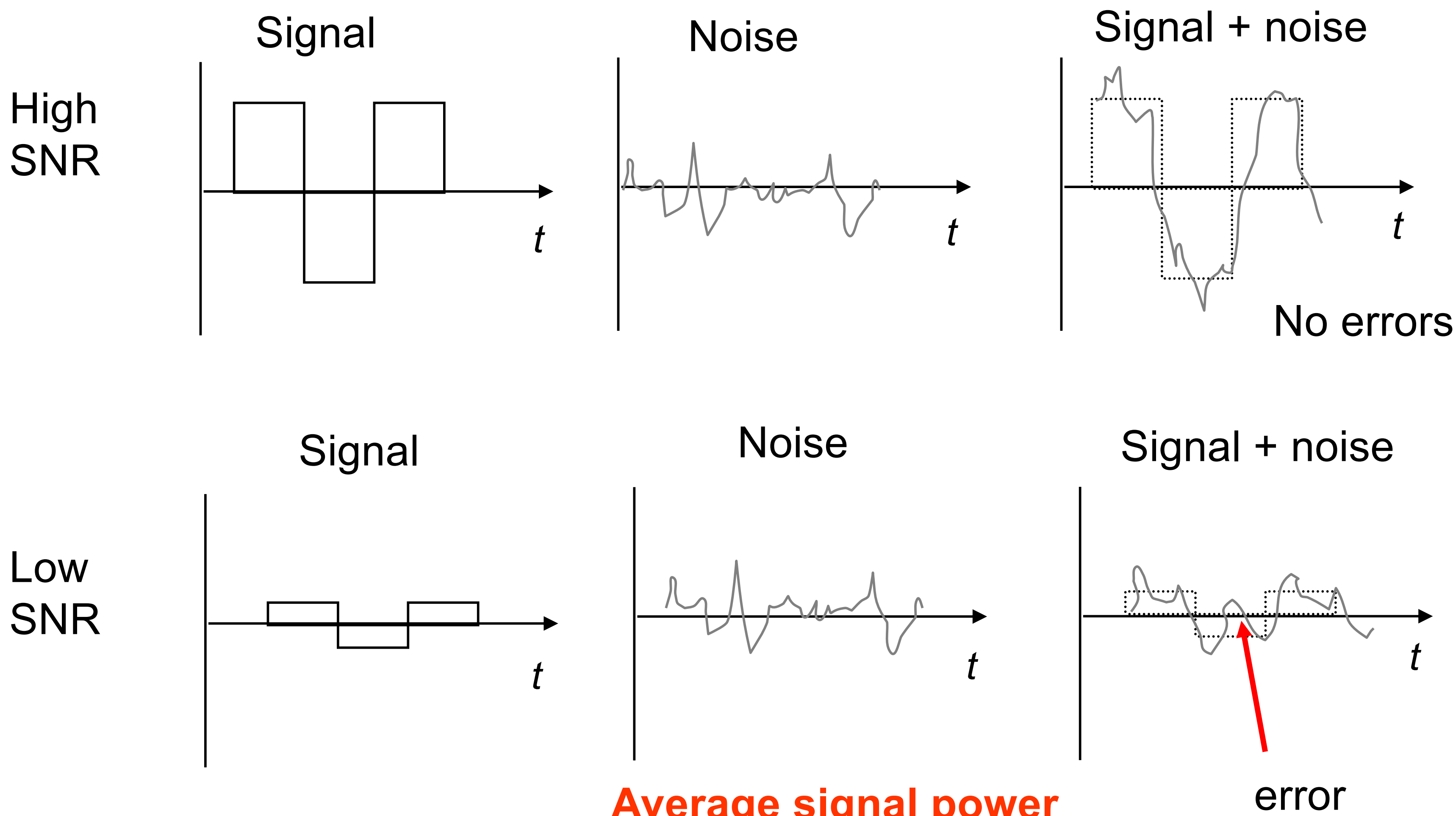
Bit Rate = m bits/pulse x 2W_c pulses/sec = 2mW_c bps

In the absence of noise, the bit rate can be increased without limit by increasing m

Noise & Reliable Communications

- All physical systems have noise
  - Electrons always vibrate at non-zero temperature
  - Motion of electrons induces noise
- Presence of noise limits accuracy of measurement of received signal amplitude
- Errors occur if signal separation is comparable to noise level
- Bit Error Rate (BER) increases with decreasing signal-to-noise ratio
- Noise places a limit on how many amplitude levels can be used in pulse transmission

Signal-to-Noise Ratio



$$\text{SNR} = \frac{\textbf{Average signal power}}{\textbf{Average noise power}}$$

$$\text{SNR (dB)} = 10 \log_{10} \text{SNR}$$

Shannon Channel Capacity

$$C = W_c \log_2 (1 + SNR) \text{ bps}$$

- Arbitrarily reliable communications is possible if the transmission rate $R < C$.
- If $R > C$, then arbitrarily reliable communications is not possible.
- "Arbitrarily reliable" means the BER can be made arbitrarily small through sufficiently complex coding.
- C can be used as a measure of how close a system design is to the best achievable performance.
- Bandwidth W_c & SNR determine C

Example

- Find the Shannon channel capacity for a telephone channel with W_c = 3400 Hz and SNR = 10000

$$C = 3400 \log_2 (1 + 10000)$$
$$= 3400 \log_{10} (10001)/\log_{10} 2 = 45200 \text{ bps}$$

Note that SNR = 10000 corresponds to

SNR (dB) = $10 \log_{10}(10001) = 40$ dB

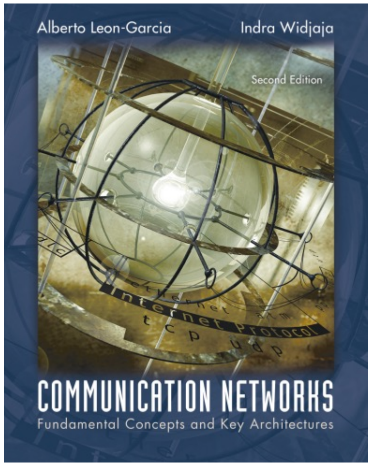
Bit Rates of Digital Transmission Systems

System	Bit Rate	Observations
Telephone twisted pair	33.6-56 kbps	4 kHz telephone channel
Ethernet twisted pair	10 Mbps, 100 Mbps	100 meters of unshielded twisted copper wire pair
Cable modem	500 kbps-4 Mbps	Shared CATV return channel
ADSL twisted pair	64-640 kbps in, 1.536-6.144 Mbps out	Coexists with analog telephone signal
2.4 GHz radio	2-11 Mbps	IEEE 802.11b wireless LAN
28 GHz radio	1.5-45 Mbps	5 km multipoint radio
Optical fiber	2.5-10 Gbps	1 wavelength
Optical fiber	>1600 Gbps	Many wavelengths

Examples of Channels

Channel	Bandwidth	Bit Rates
Telephone voice channel	3 kHz	33 kbps
Copper pair	1 MHz	1-6 Mbps
Coaxial cable	500 MHz (6 MHz channels)	30 Mbps/ channel
5 GHz radio (IEEE 802.11)	300 MHz (11 channels)	54 Mbps / channel
Optical fiber	Many TeraHertz	40 Gbps / wavelength

Chapter 3
Digital Transmission Fundamentals



Fundamental Limits in Digital Transmission

Multilevel Signaling

- Nyquist pulses achieve the maximum signalling rate with zero ISI,

  $2W_c$ pulses per second or
  $2W_c$ pulses / W_c Hz = 2 pulses / Hz

- With two signal levels, each pulse carries one bit of information

  Bit rate = $2W_c$ bits/second

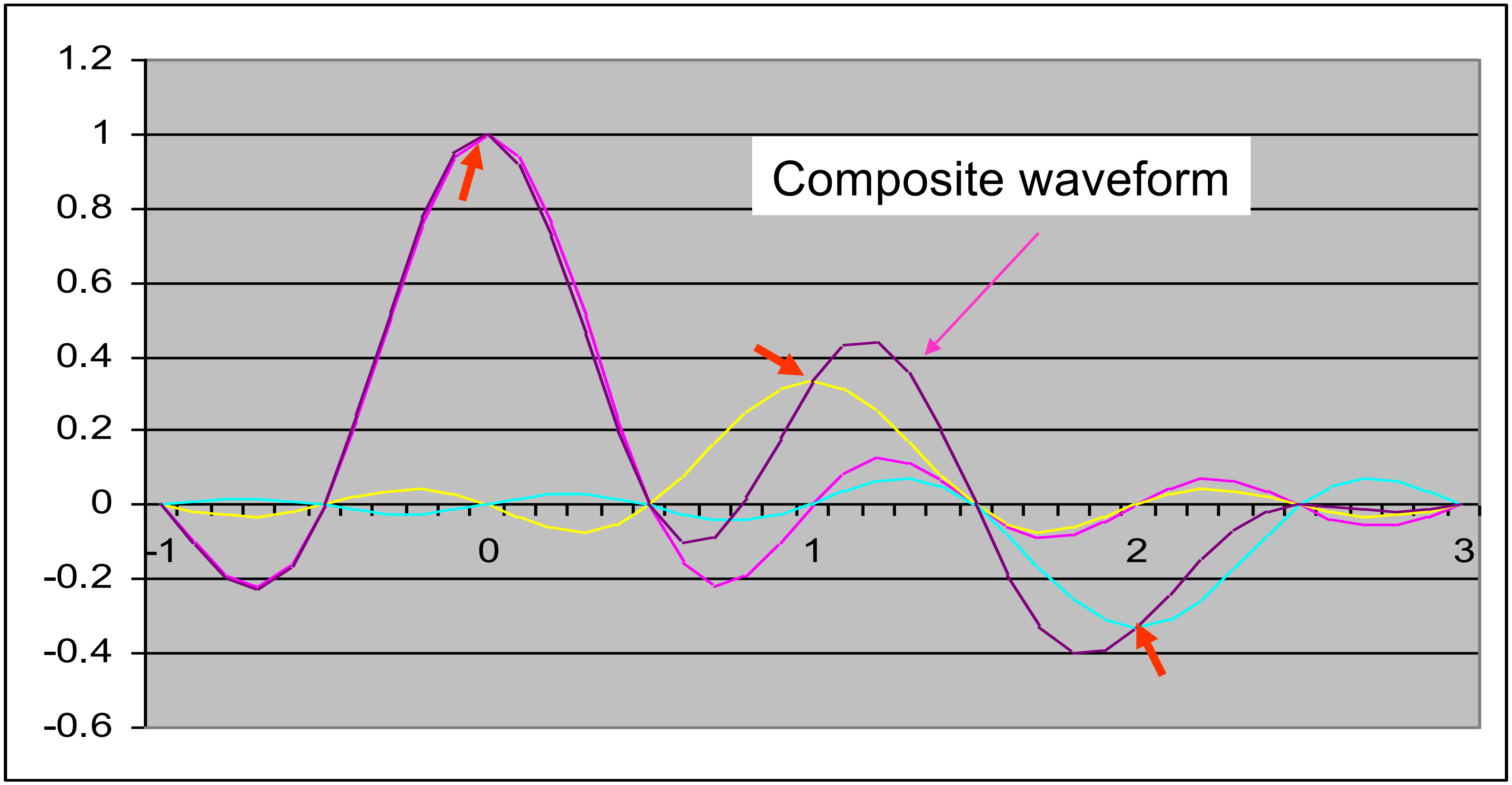
- With $M = 2^m$ signal levels, each pulse carries m bits

  Bit rate = $2W_c$ pulses/sec. * m bits/pulse = $2W_c\,m$ bps

- *Bit rate can be increased by increasing number of levels*
- *r(t) includes additive noise, that limits number of levels that can be used reliably.*

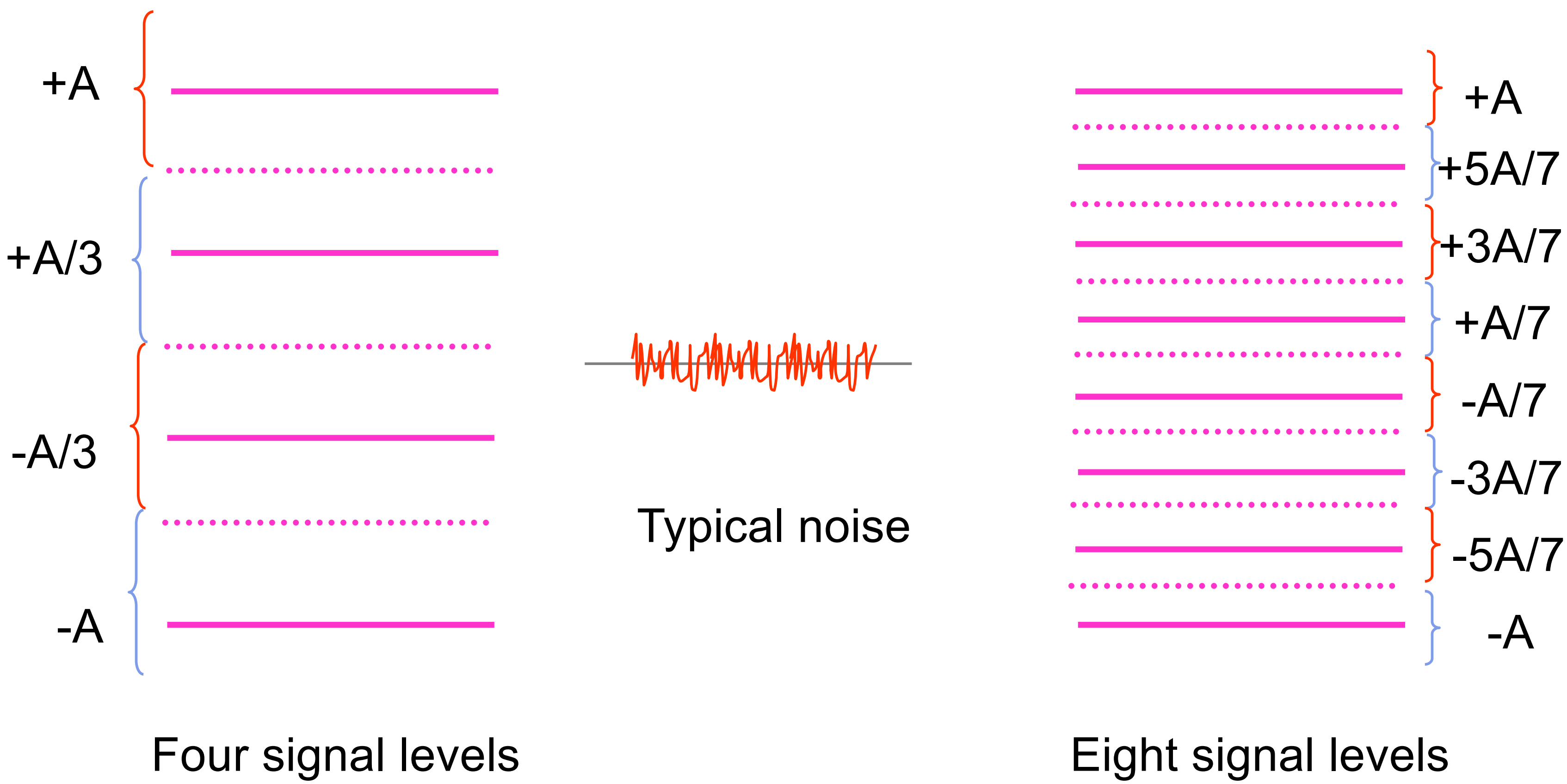
Example of Multilevel Signaling

- Four levels {-1, -1/3, 1/3, +1} for {00,01,10,11}
- Waveform for 11,10,01 sends +1, +1/3, -1/3
- Zero ISI at sampling instants



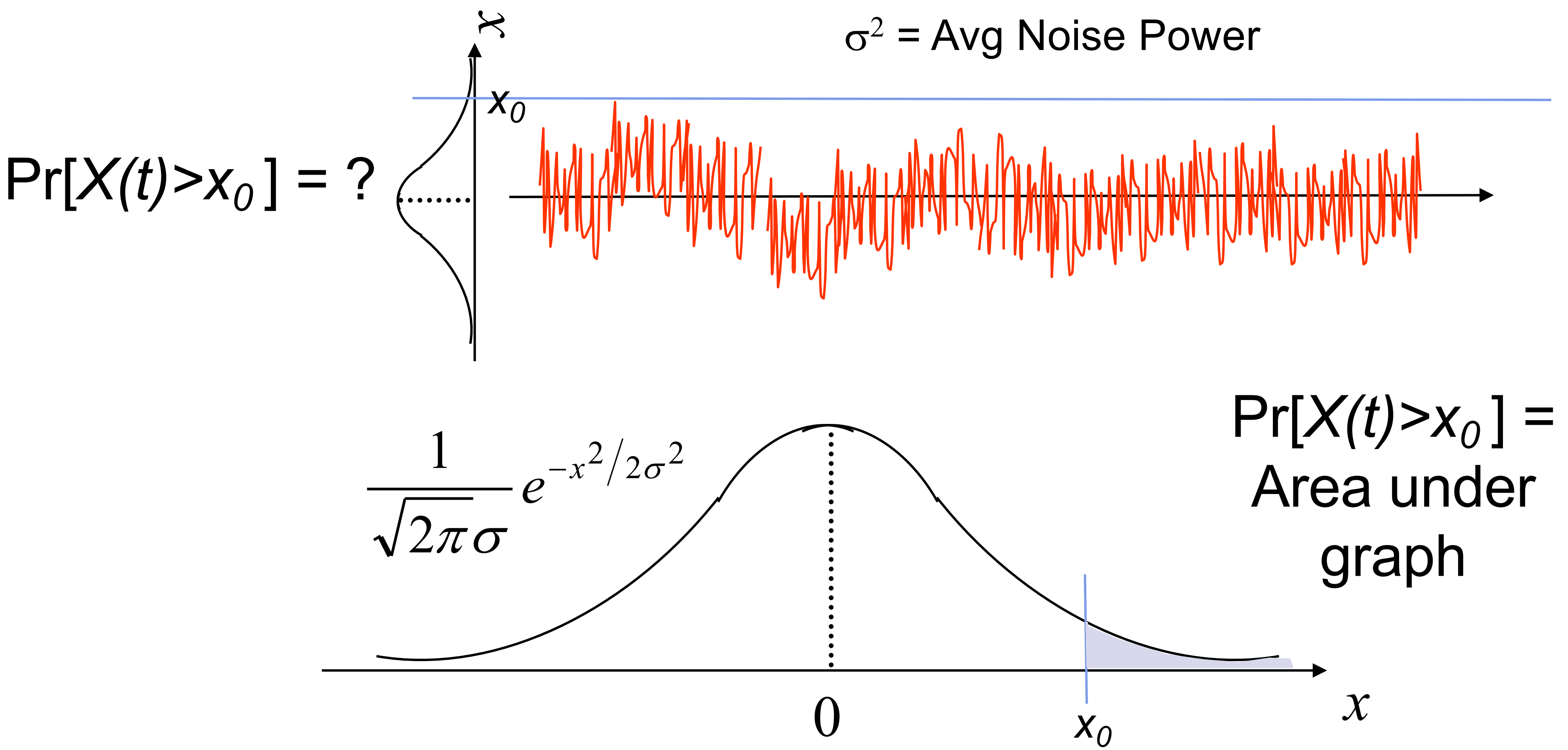
Noise Limits Accuracy

- Receiver makes decision based on transmitted pulse level + noise
- Error rate depends on relative value of noise amplitude and spacing between signal levels
- Large (positive or negative) noise values can cause wrong decision
- Noise level below impacts 8-level signaling more than 4-level signaling

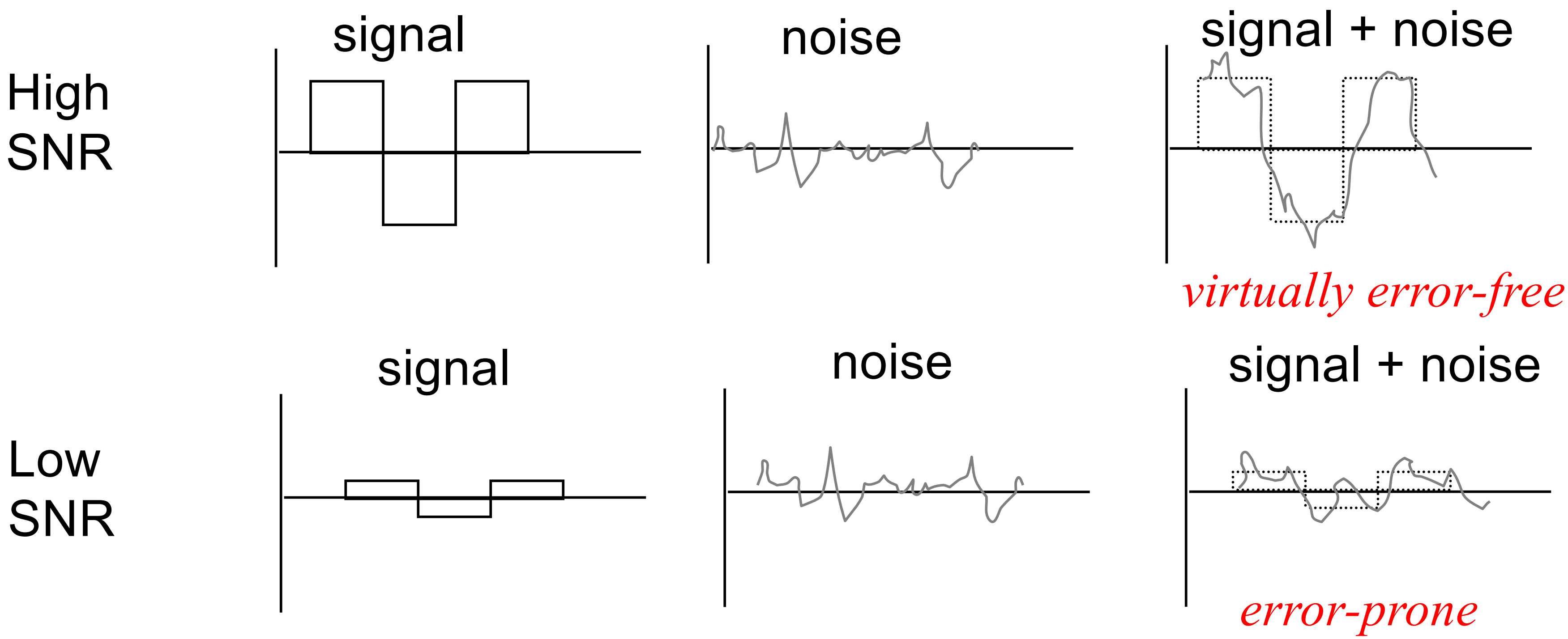


Noise distribution

- Noise is characterized by probability density of amplitude samples
- Likelihood that certain amplitude occurs
- Thermal electronic noise is inevitable (due to vibrations of electrons)
- Noise distribution is Gaussian (bell-shaped) as below



Channel Noise affects Reliability



$$\text{SNR} = \frac{\text{Average Signal Power}}{\text{Average Noise Power}}$$

$$\text{SNR (dB)} = 10 \log_{10} \text{SNR}$$

Shannon Channel Capacity

- If transmitted power is limited, then as *M* increases spacing between levels decreases
- Presence of noise at receiver causes more frequent errors to occur as *M* is increased

Shannon Channel Capacity:

The maximum reliable transmission rate over an ideal channel with bandwidth *W* Hz, with Gaussian distributed noise, and with SNR *S/N* is

$$C = W \log_2 (1 + S/N) \text{ bits per second}$$

- Reliable means error rate can be made arbitrarily small by proper coding

Example

- Consider a 3 kHz channel with 8-level signaling. Compare bit rate to channel capacity at 20 dB SNR

- 3KHz telephone channel with 8 level signaling

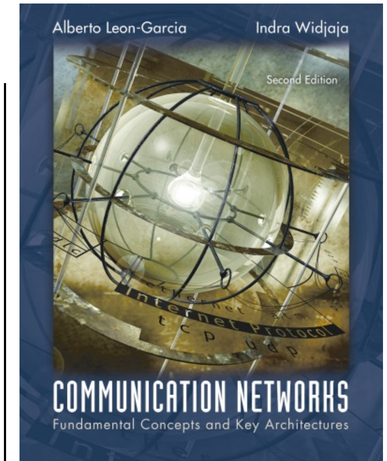
  Bit rate = 2*3000 pulses/sec * 3 bits/pulse = 18 kbps

- 20 dB SNR means $10 \log_{10} S/N = 20$

  Implies $S/N = 100$
- Shannon Channel Capacity is then

  $C = 3000 \log (1 + 100) = 19,963$ bits/second

Chapter 3
Digital Transmission Fundamentals

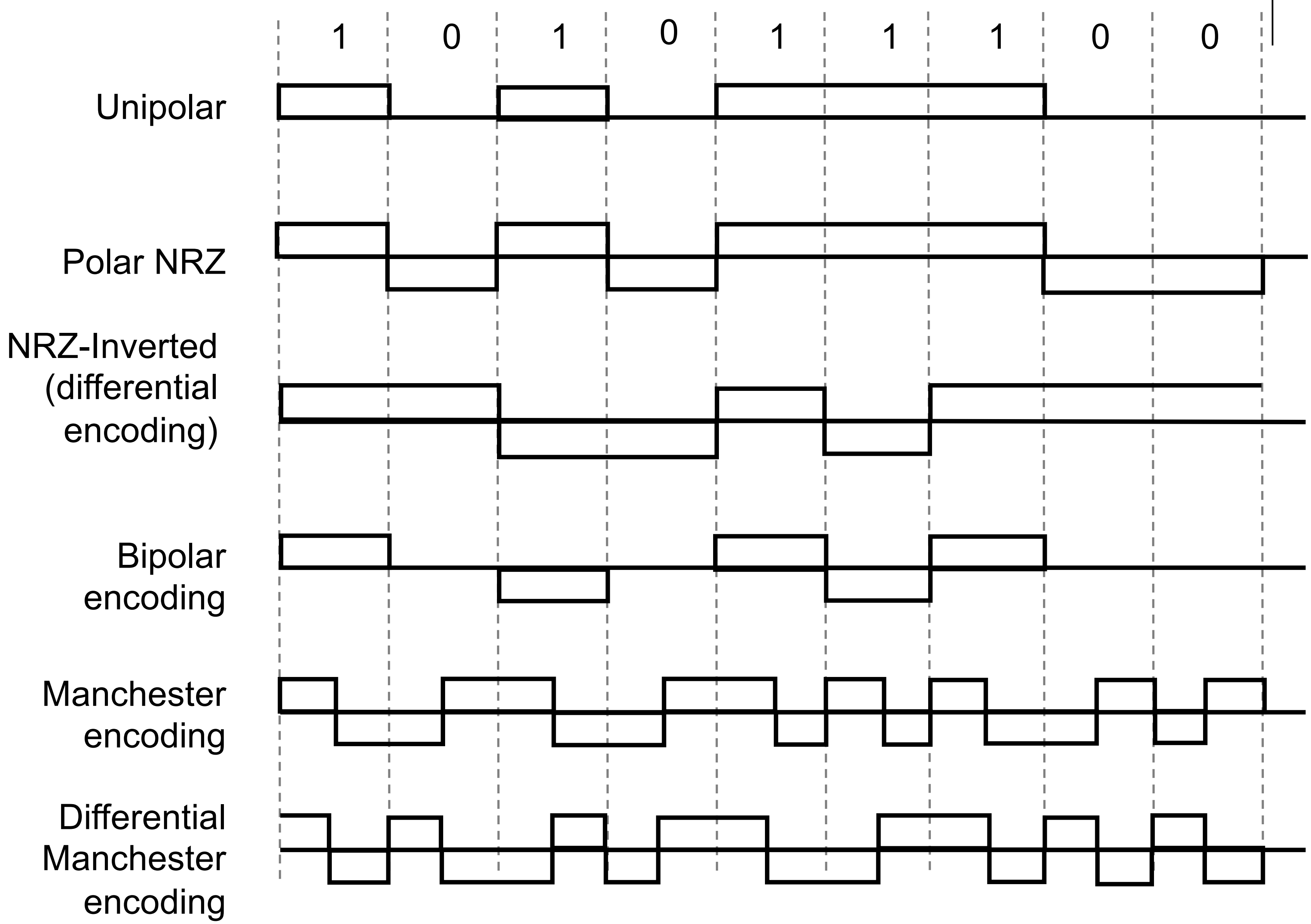


Line Coding

What is Line Coding?

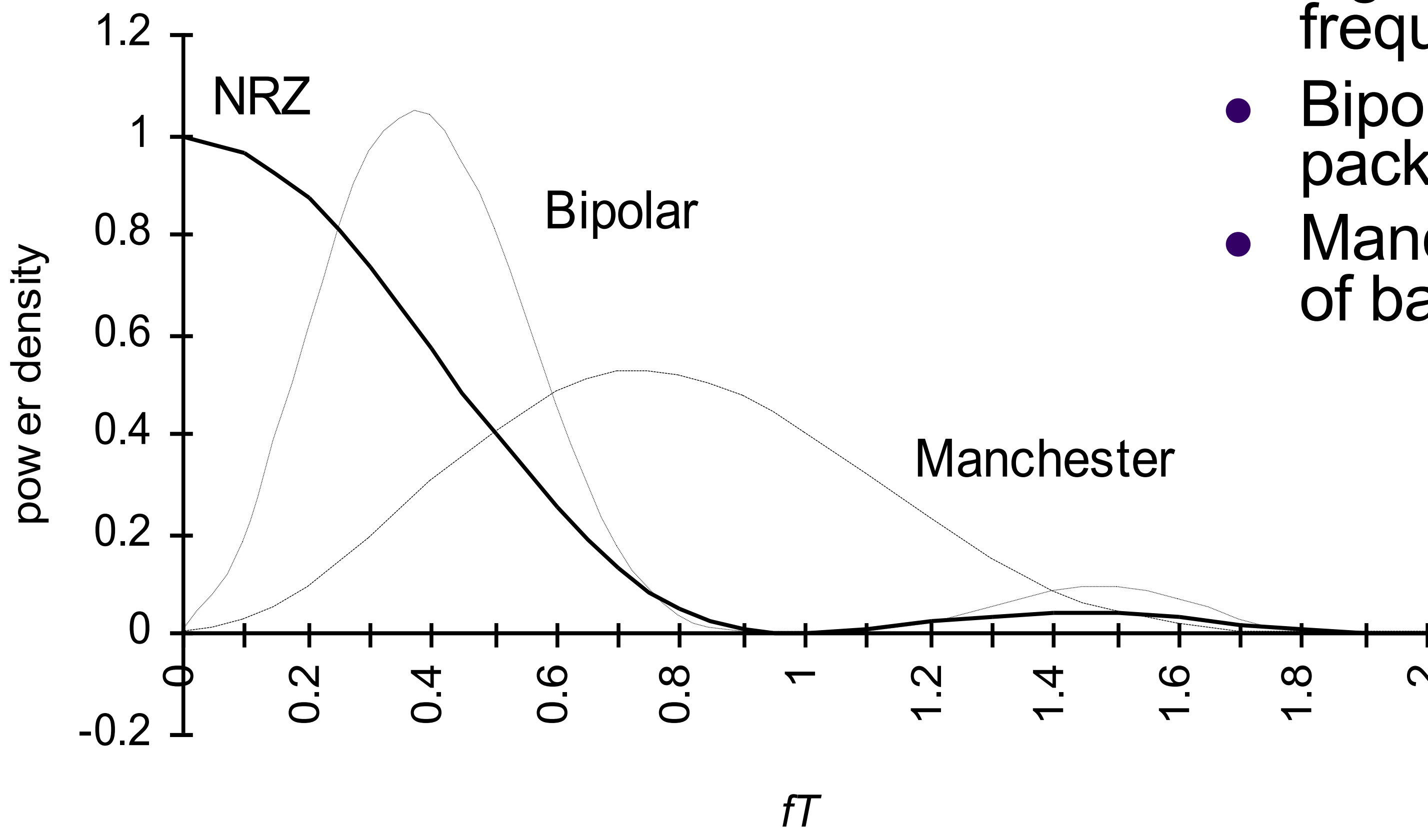
- Mapping of binary information sequence into the digital signal that enters the channel
  - Ex. “1” maps to +A square pulse; “0” to –A pulse
- Line code selected to meet desired system requirements:
  - *Transmitted power*: Power consumption = $ (SNR concerns)
  - *Bit timing*: Transitions in signal help timing recovery
  - *Bandwidth efficiency* (*bit rate*) : Excessive transitions wastes bw
  - *Low frequency content*: Some channels block low frequencies
    - long periods of +A or of –A causes signal to “droop”
    - Waveform should not have low-frequency content (DC component)
  - *Error detection*: Ability to detect errors helps
  - *Polarity Concerns*: Easier / cheaper installation?
  - *Complexity/cost*: Is code implementable in chip at high speed?

Line coding examples



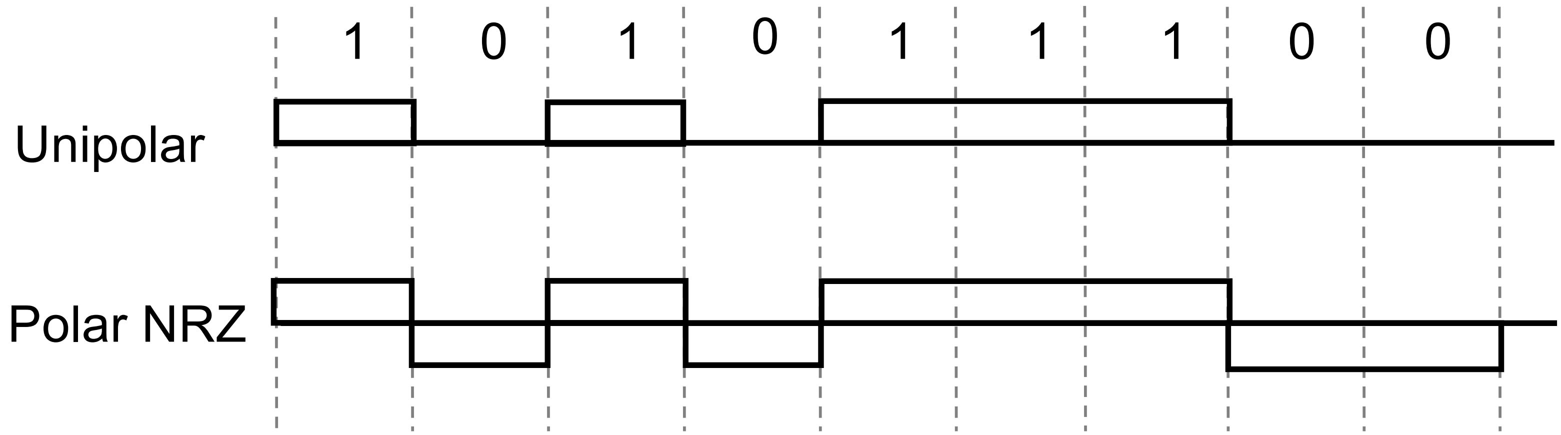
Spectrum of Line codes

- Assume 1s & 0s independent & equiprobable



- NRZ-Inverted has high content at low frequencies
- Bipolar tightly packed around *T*/2
- Manchester wasteful of bandwidth

Unipolar & Polar Non-Return-to-Zero (NRZ)



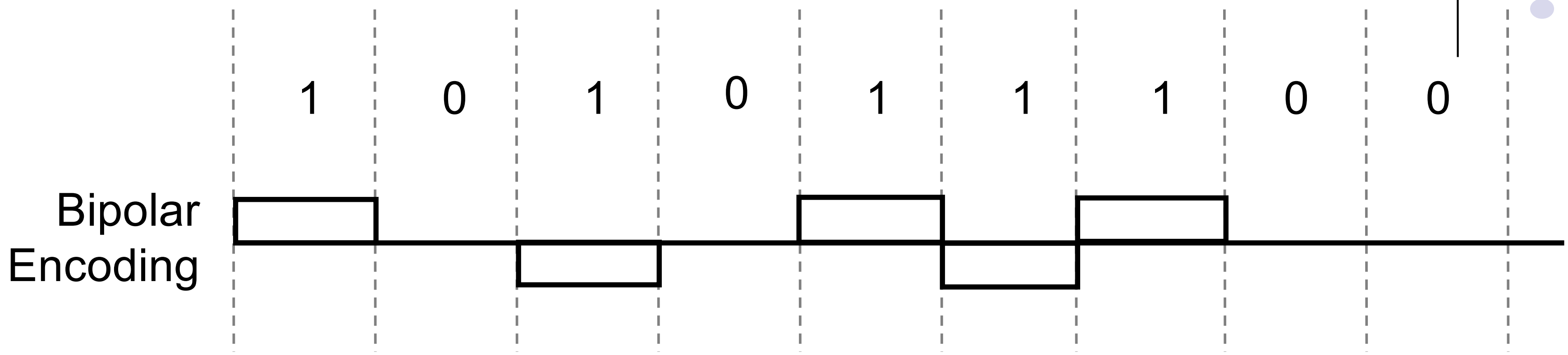
Unipolar

- "1" maps to +A pulse
- "0" maps to no pulse
- High Average Power
  $0.5*A^2 +0.5*0^2=A^2/2$
- Long strings of A or 0
  - Poor timing
  - Low-frequency content
- Simple

Polar NRZ

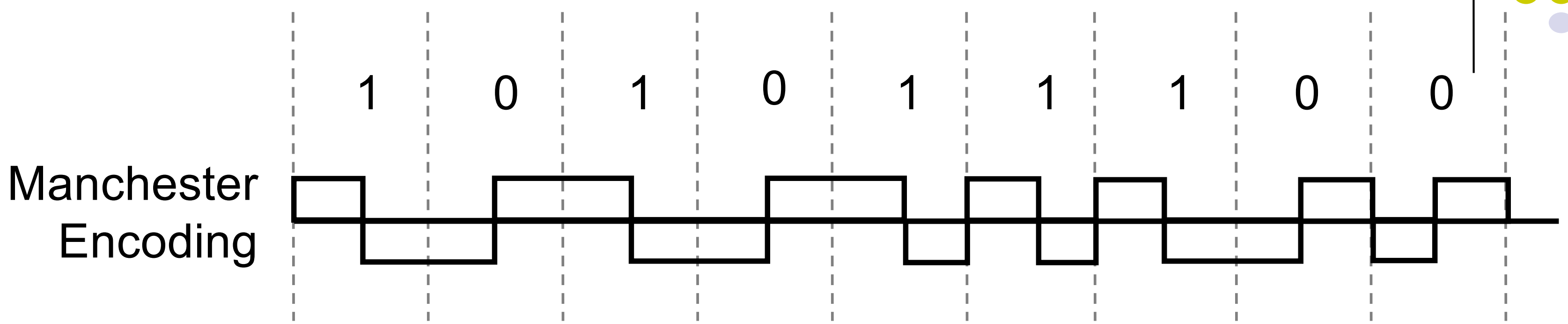
- "1" maps to +A/2 pulse
- "0" maps to –A/2 pulse
- Better Average Power
  $0.5*(A/2)^2 +0.5*(-A/2)^2=A^2/4$
- Long strings of +A/2 or –A/2
  - Poor timing
  - Low-frequency content
- Simple

Bipolar Code



- Three signal levels: {-A, 0, +A}
- "1" maps to +A or –A in alternation
- "0" maps to no pulse
  - Every +pulse matched by –pulse so little content at low frequencies
- String of 1s produces a square wave
  - Spectrum centered at *T*/2
- Long string of 0s causes receiver to lose synch
- Zero Substitution (B6ZS)

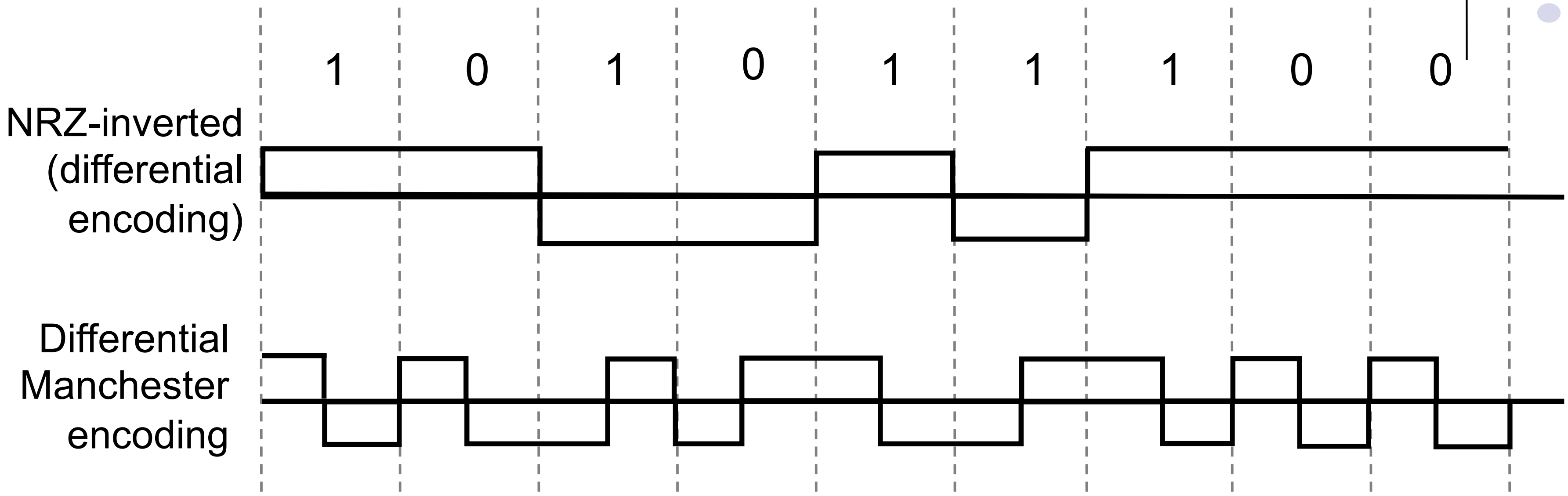
Manchester code & *m*B*n*B codes



- "1" maps into A/2 first *T*/2, -A/2 last T/2
- "0" maps into -A/2 first *T*/2, A/2 last *T*/2
- Every interval has transition in middle
  - Timing recovery easy
  - Uses double the minimum bandwidth
- Simple to implement
- Used in 10-Mbps Ethernet & other LAN standards

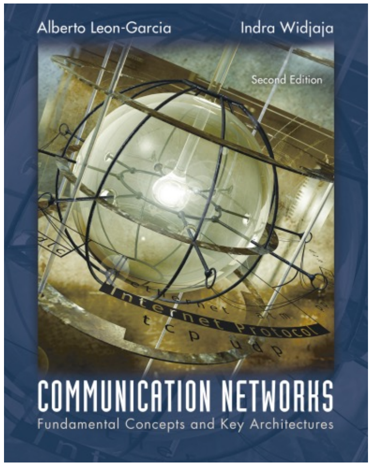
- *m*B*n*B line code
- Maps block of *m* bits into *n* bits
- Manchester code is 1B2B code
- 4B5B code used in 100 Mbps Ethernet & FDDI
- 8B10b code used in Gigabit Ethernet
- 64B66B code used in 10G Ethernet

Differential Coding



- Errors in some systems cause transposition in polarity, +A become –A and vice versa
  - All subsequent bits in Polar coding would be in error
- Differential line coding provides robustness to this type of error
- “1” mapped into transition in signal level
- “0” mapped into no transition in signal level
- Same spectrum as Polar-NRZ
- Errors occur in pairs
- Also used with Manchester coding

Chapter 3
Digital Transmission Fundamentals

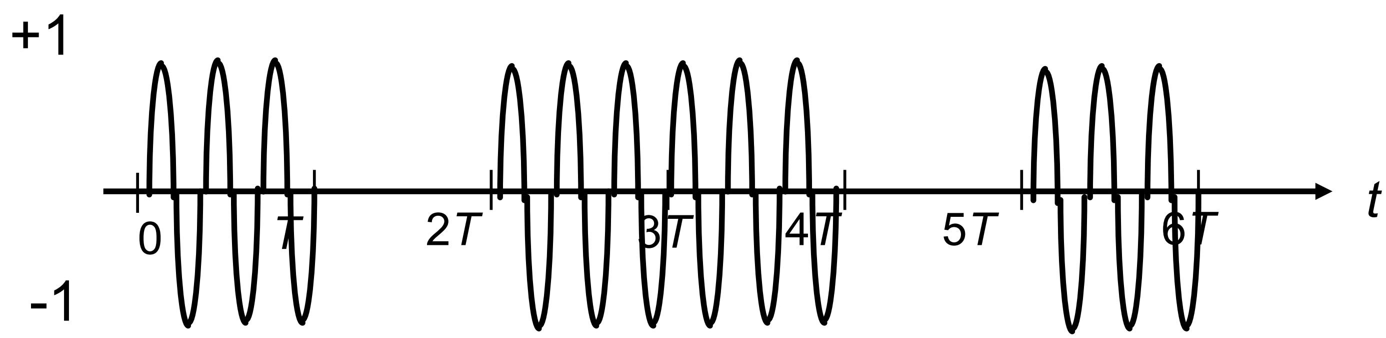


Modems and Digital Modulation

Amplitude Modulation and Frequency Modulation

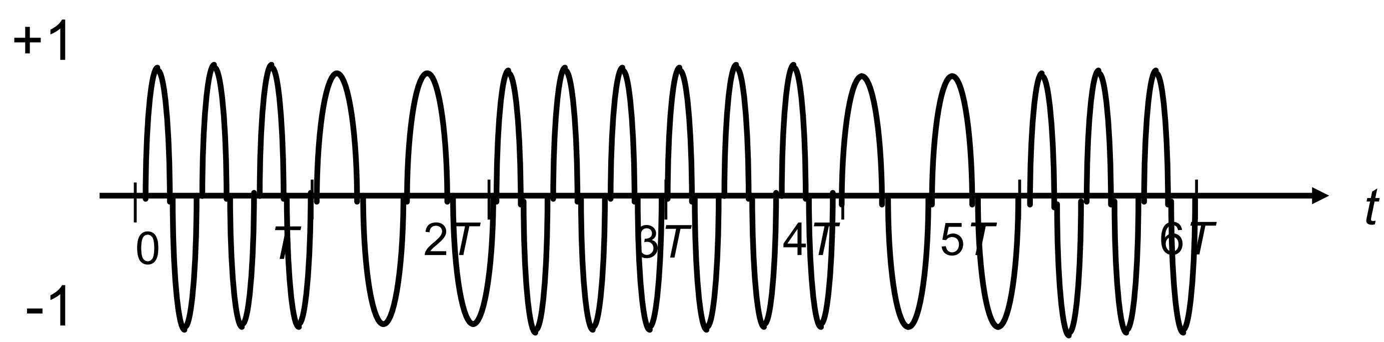
Information 1 0 1 1 0 1

Amplitude Shift Keying



Map bits into amplitude of sinusoid: “1” send sinusoid; “0” no sinusoid
Demodulator looks for signal vs. no signal

Frequency Shift Keying

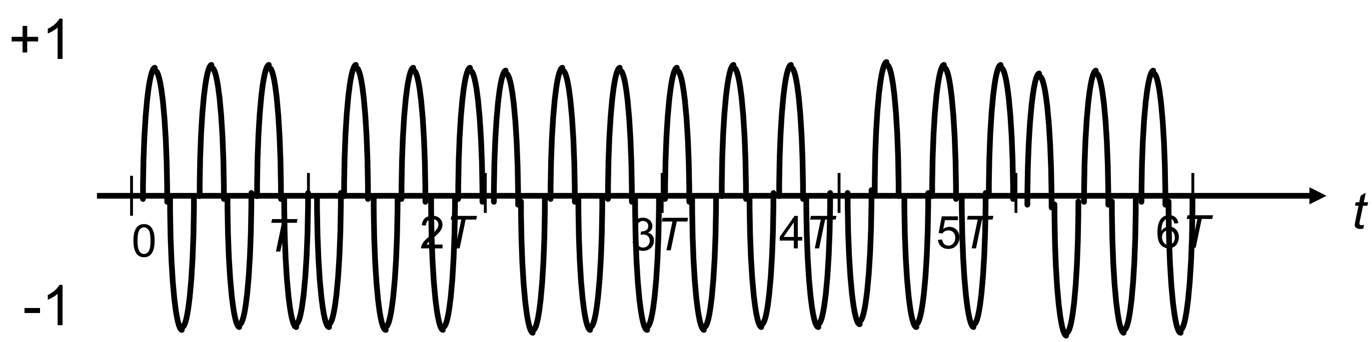


Map bits into frequency: “1” send frequency $f_c + \delta$; “0” send frequency $f_c - \delta$
Demodulator looks for power around $f_c + \delta$ or $f_c - \delta$

Phase Modulation

Information 1 0 1 1 0 1

Phase Shift Keying

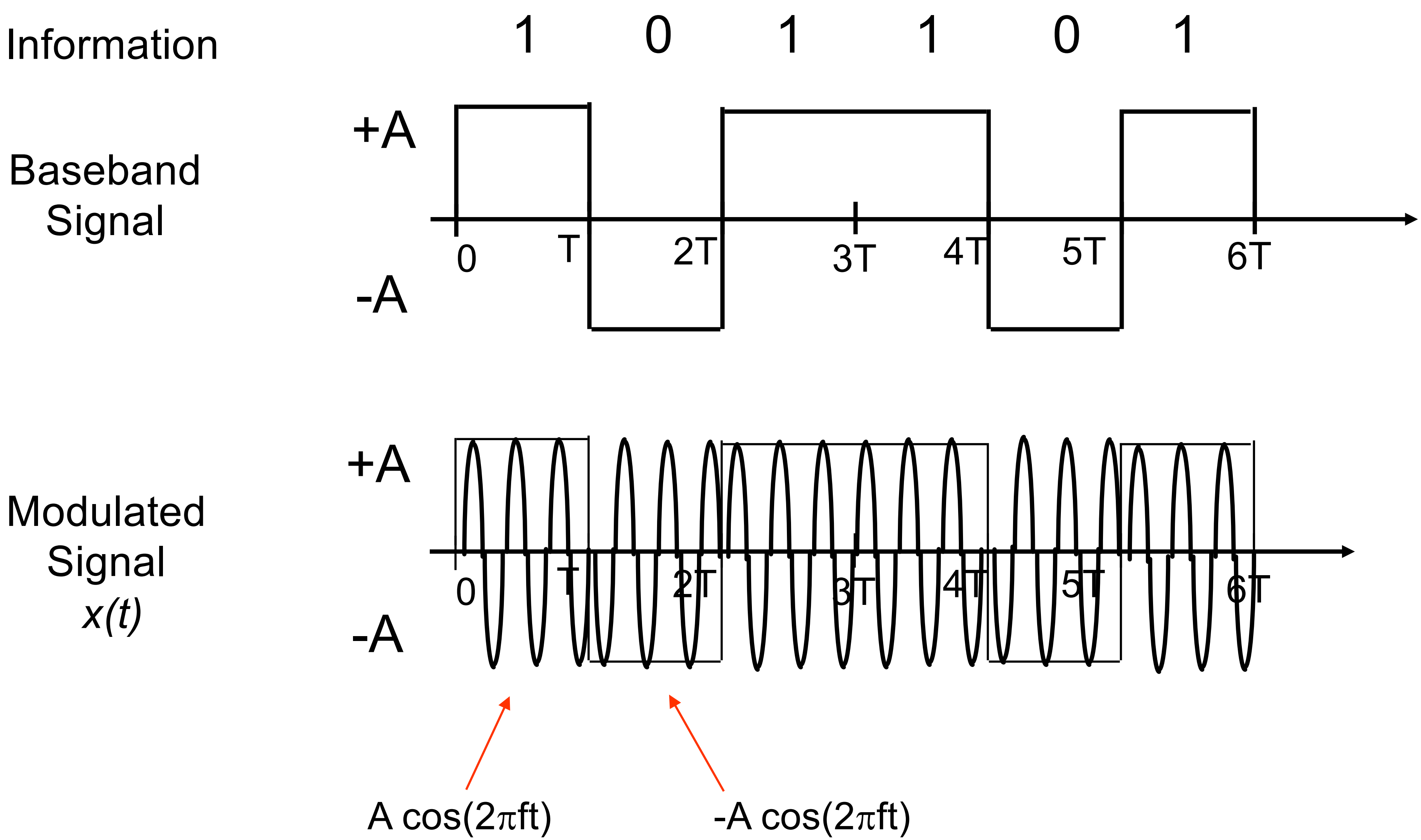


- Map bits into phase of sinusoid:
  - "1" send A $\cos(2\pi ft)$, i.e. phase is 0
  - "0" send A $\cos(2\pi ft+\pi)$, i.e. phase is π

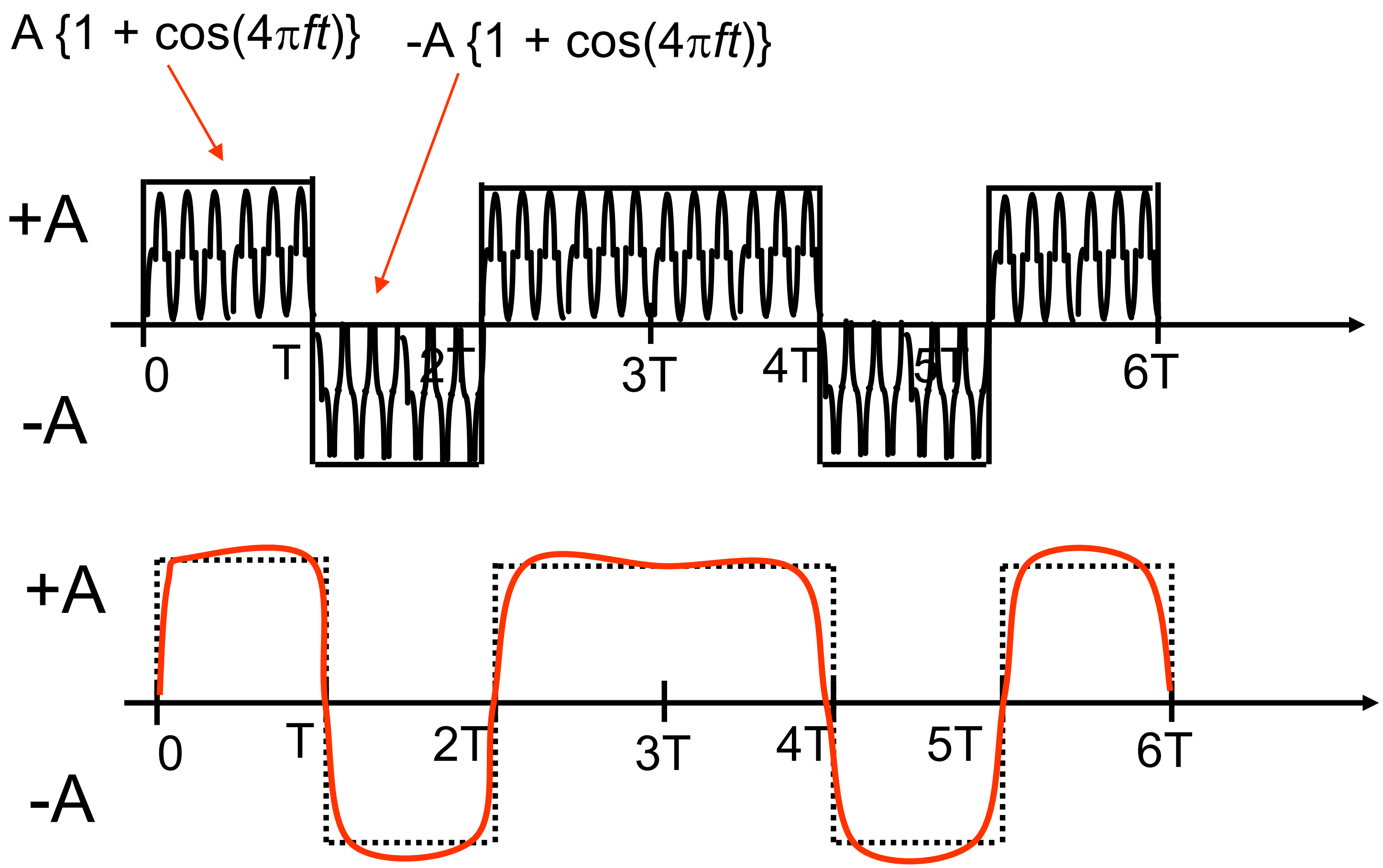
- Equivalent to multiplying $\cos(2\pi ft)$ by +A or -A
  - "1" send A $\cos(2\pi ft)$, i.e. multiply by 1
  - "0" send A $\cos(2\pi ft+\pi) = -A\cos(2\pi ft)$, i.e. multiply by -1

- We will focus on phase modulation

Example of Modulation



Example of Demodulation



After multiplication at receiver x(t) cos($2\pi f_c t$)

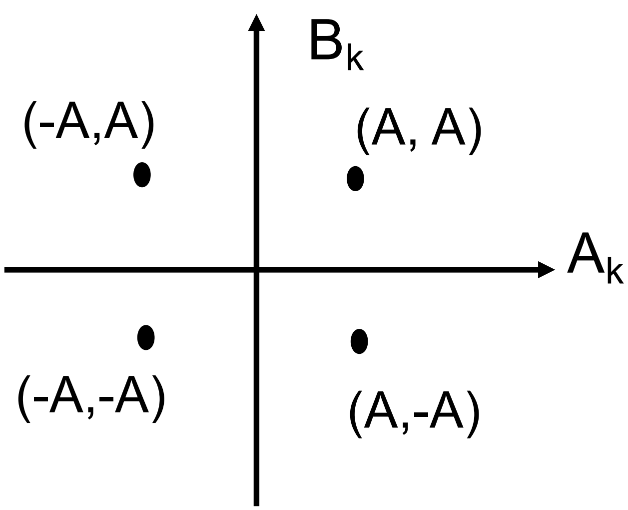
Baseband signal discernable after smoothing

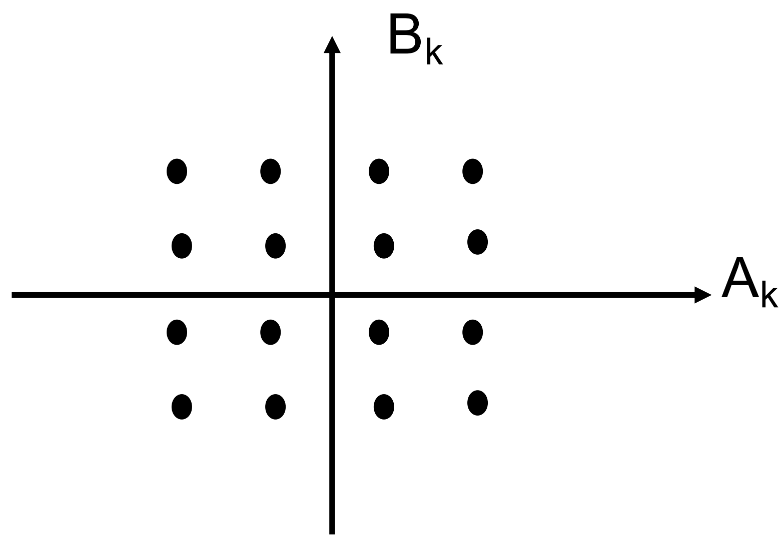
Recovered Information

1 0 1 1 0 1

Signal Constellations

- Each pair (A_k, B_k) defines a point in the plane
- *Signal constellation* set of signaling points





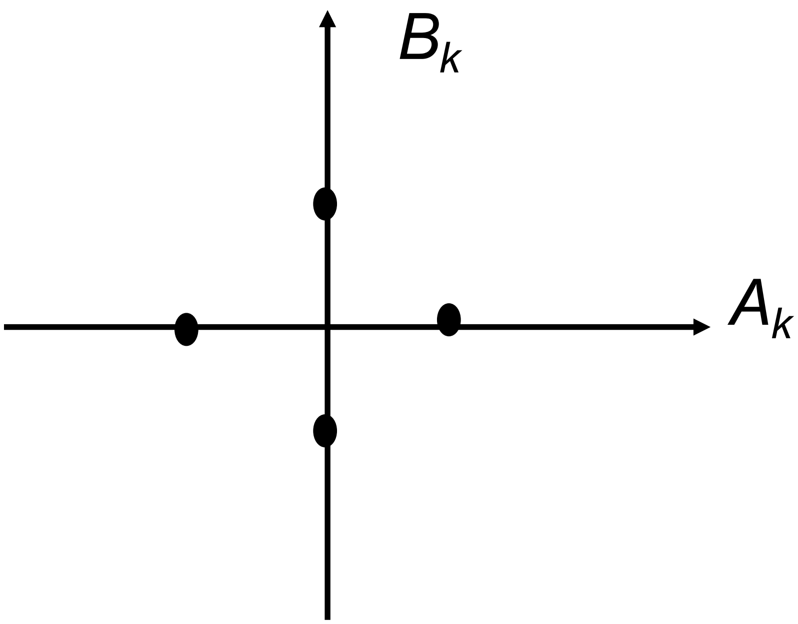
4 possible points per *T* sec.
2 bits / pulse

16 possible points per *T* sec.
4 bits / pulse

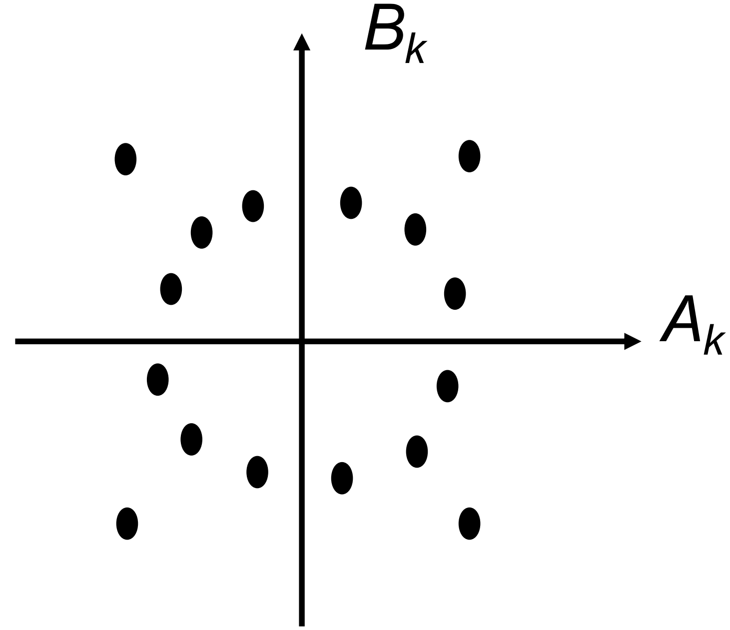
Other Signal Constellations

- Point selected by amplitude & phase

$$A_k \cos(2\pi f_c t) + B_k \sin(2\pi f_c t) = \sqrt{A_k^2 + B_k^2} \cos(2\pi f_c t + \tan^{-1}(B_k/A_k))$$

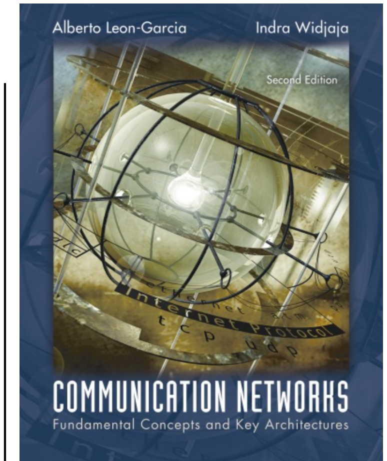


4 possible points per T sec.



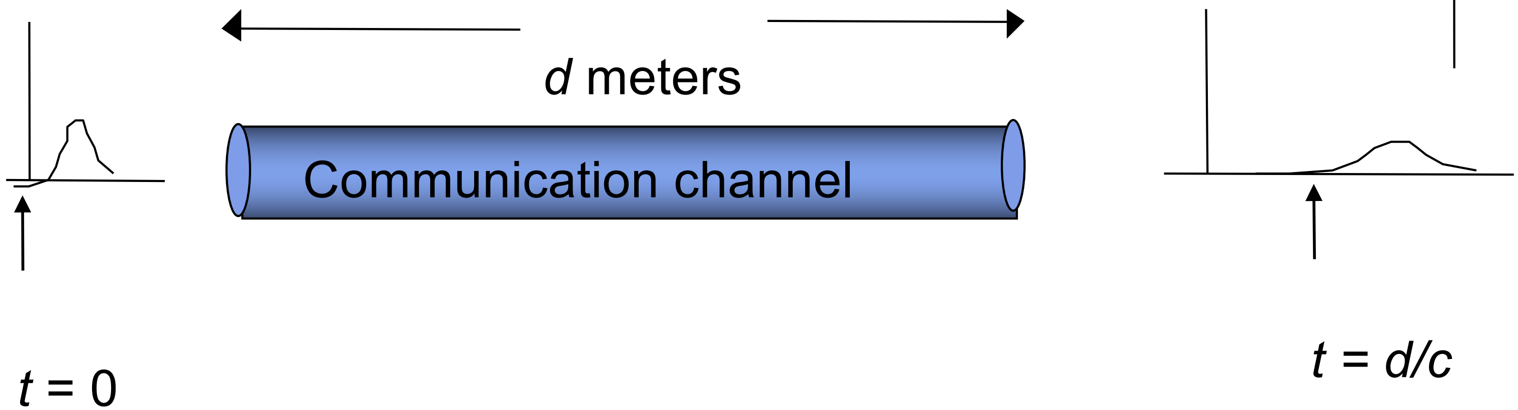
16 possible points per T sec.

Chapter 3
Digital Transmission Fundamentals



Properties of Media and Digital Transmission Systems

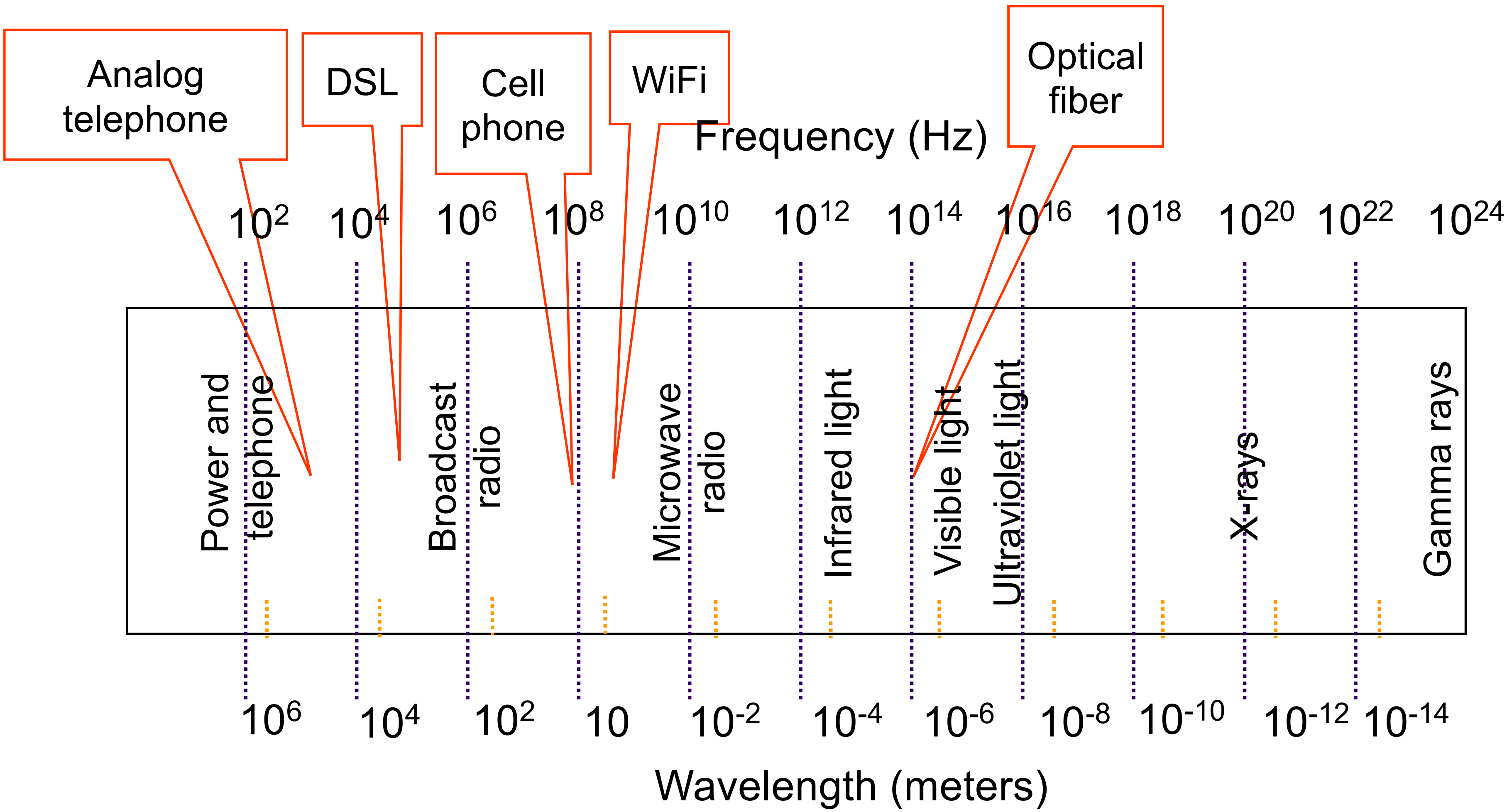
Fundamental Issues in Transmission Media



- Information bearing capacity
  - Amplitude response & bandwidth
    - dependence on distance
  - Susceptibility to noise & interference
    - Error rates & SNRs
- Propagation speed of signal
  - $c = 3 \times 10^8$ meters/second in vacuum
  - $\nu = c/\sqrt{\varepsilon}$ speed of light in medium where $\varepsilon > 1$ is the dielectric constant of the medium
  - $\nu = 2.3 \times 10^8$ m/sec in copper wire; $\nu = 2.0 \times 10^8$ m/sec in optical fiber

Communications systems & Electromagnetic Spectrum

- Frequency of communications signals



Wireless & Wired Media

Wireless Media

- Signal energy propagates in space, limited directionality
- Interference possible, so spectrum regulated
- Limited bandwidth
- Simple infrastructure: antennas & transmitters
- No physical connection between network & user
- Users can move

Wired Media

- Signal energy contained & guided within medium
- Spectrum can be re-used in separate media (wires or cables), more scalable
- Extremely high bandwidth
- Complex infrastructure: ducts, conduits, poles, right-of-way

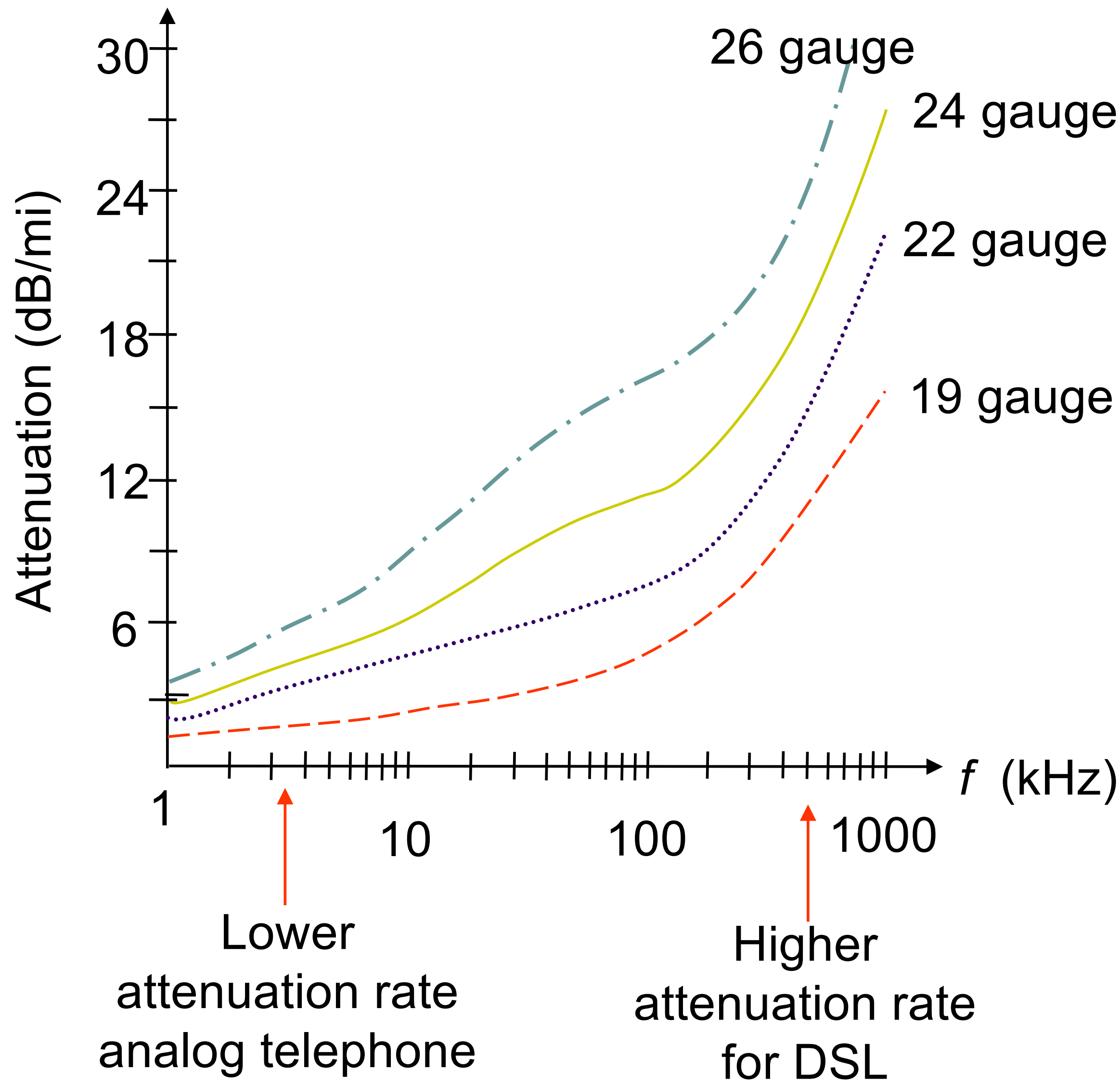
Attenuation

- Attenuation varies with media
  - Dependence on distance of central importance
- Wired media has exponential dependence
  - Received power at d meters proportional to 10^{-kd}
  - Attenuation in dB = $k\,d$, where k is dB/meter
- Wireless media has logarithmic dependence
  - Received power at d meters proportional to d^{-n}
  - Attenuation in dB = $n \log d$, where n is path loss exponent; n=2 in free space
  - Signal level maintained for much longer distances
  - Space communications possible

Twisted Pair

Twisted pair

- Two insulated copper wires arranged in a regular spiral pattern to minimize interference
- Various thicknesses, e.g. 0.016 inch (24 gauge)
- Low cost
- Telephone subscriber loop from customer to CO
- Old trunk plant connecting telephone COs
- Intra-building telephone from wiring closet to desktop
- In old installations, loading coils added to improve quality in 3 kHz band, but more attenuation at higher frequencies



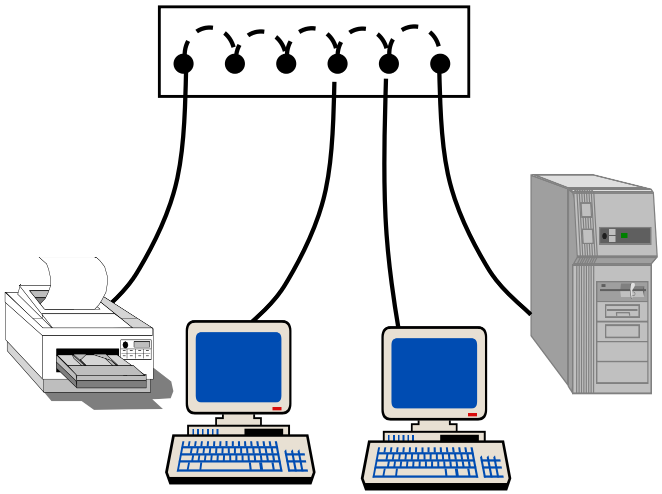
Twisted Pair Bit Rates

Table 3.5 Data rates of 24-gauge twisted pair

Standard	Data Rate	Distance
T-1	1.544 Mbps	18,000 feet, 5.5 km
DS2	6.312 Mbps	12,000 feet, 3.7 km
1/4 STS-1	12.960 Mbps	4500 feet, 1.4 km
1/2 STS-1	25.920 Mbps	3000 feet, 0.9 km
STS-1	51.840 Mbps	1000 feet, 300 m

- Twisted pairs can provide high bit rates at short distances
- Asymmetric Digital Subscriber Loop (ADSL)
  - High-speed Internet Access
  - Lower 3 kHz for voice
  - Upper band for data
  - 64 kbps inbound
  - 640 kbps outbound
- Much higher rates possible at shorter distances
  - Strategy for telephone companies is to bring fiber close to home & then twisted pair
  - Higher-speed access + video

Ethernet LANs

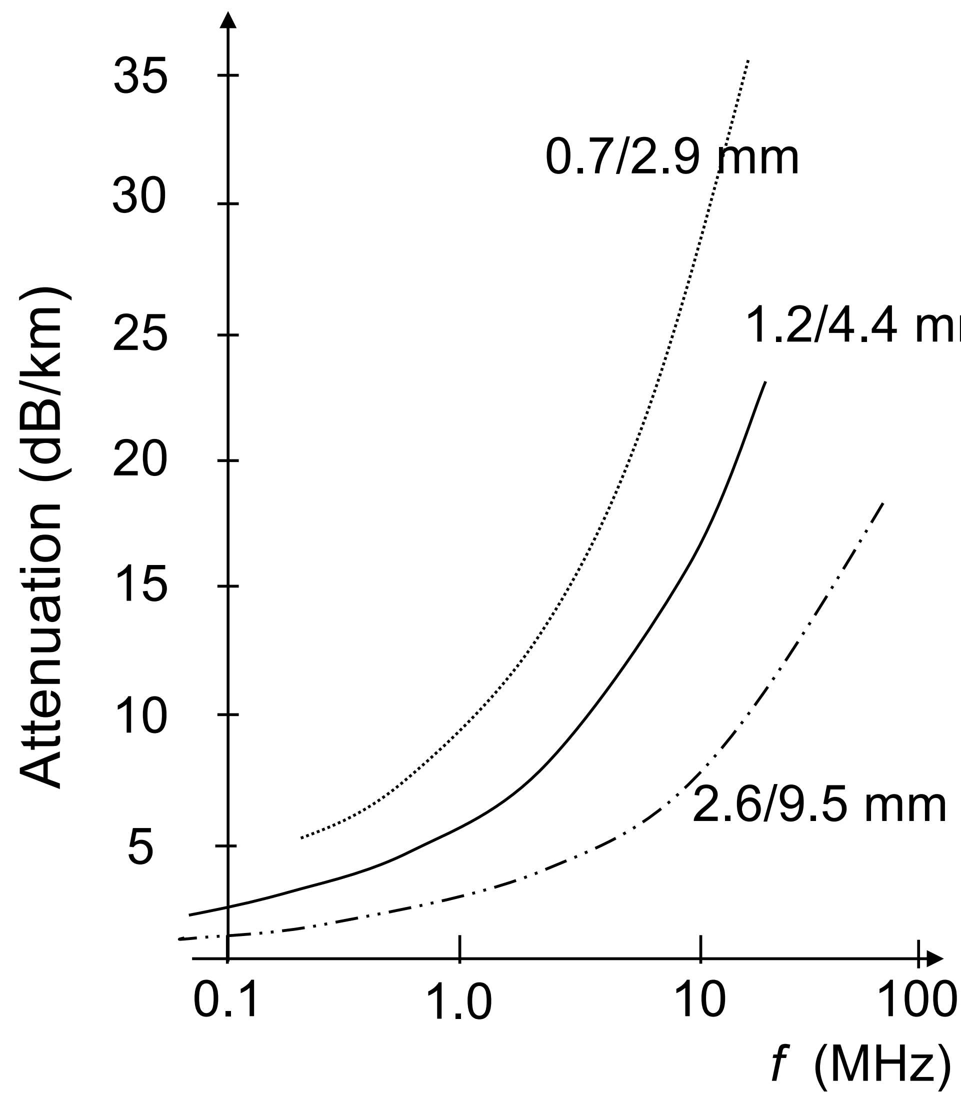


- Category 3 unshielded twisted pair (UTP): ordinary telephone wires
- Category 5 UTP: tighter twisting to improve signal quality
- Shielded twisted pair (STP): to minimize interference; costly
- 10BASE-T Ethernet
  - 10 Mbps, Baseband, Twisted pair
  - Two Cat3 pairs
  - Manchester coding, 100 meters
- 100BASE-T4 *Fast* Ethernet
  - 100 Mbps, Baseband, Twisted pair
  - Four Cat3 pairs
  - Three pairs for one direction at-a-time
  - 100/3 Mbps per pair;
  - 3B6T line code, 100 meters
- Cat5 & STP provide other options

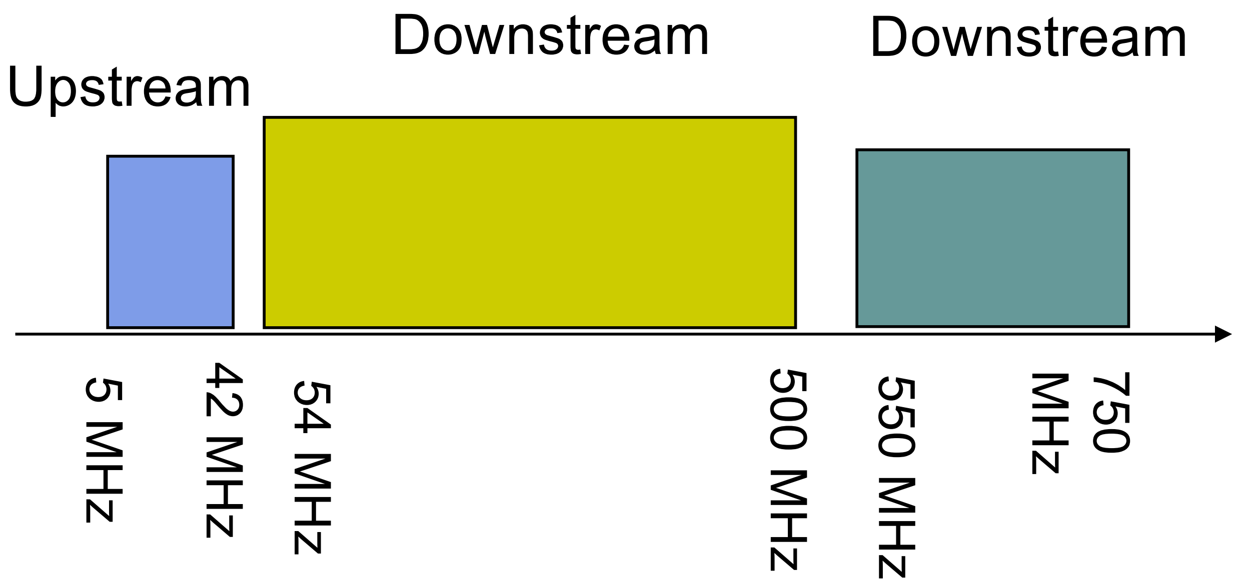
Coaxial Cable

Twisted pair

- Cylindrical braided outer conductor surrounds insulated inner wire conductor
- High interference immunity
- Higher bandwidth than twisted pair
- Hundreds of MHz
- Cable TV distribution
- Long distance telephone transmission
- Original Ethernet LAN medium

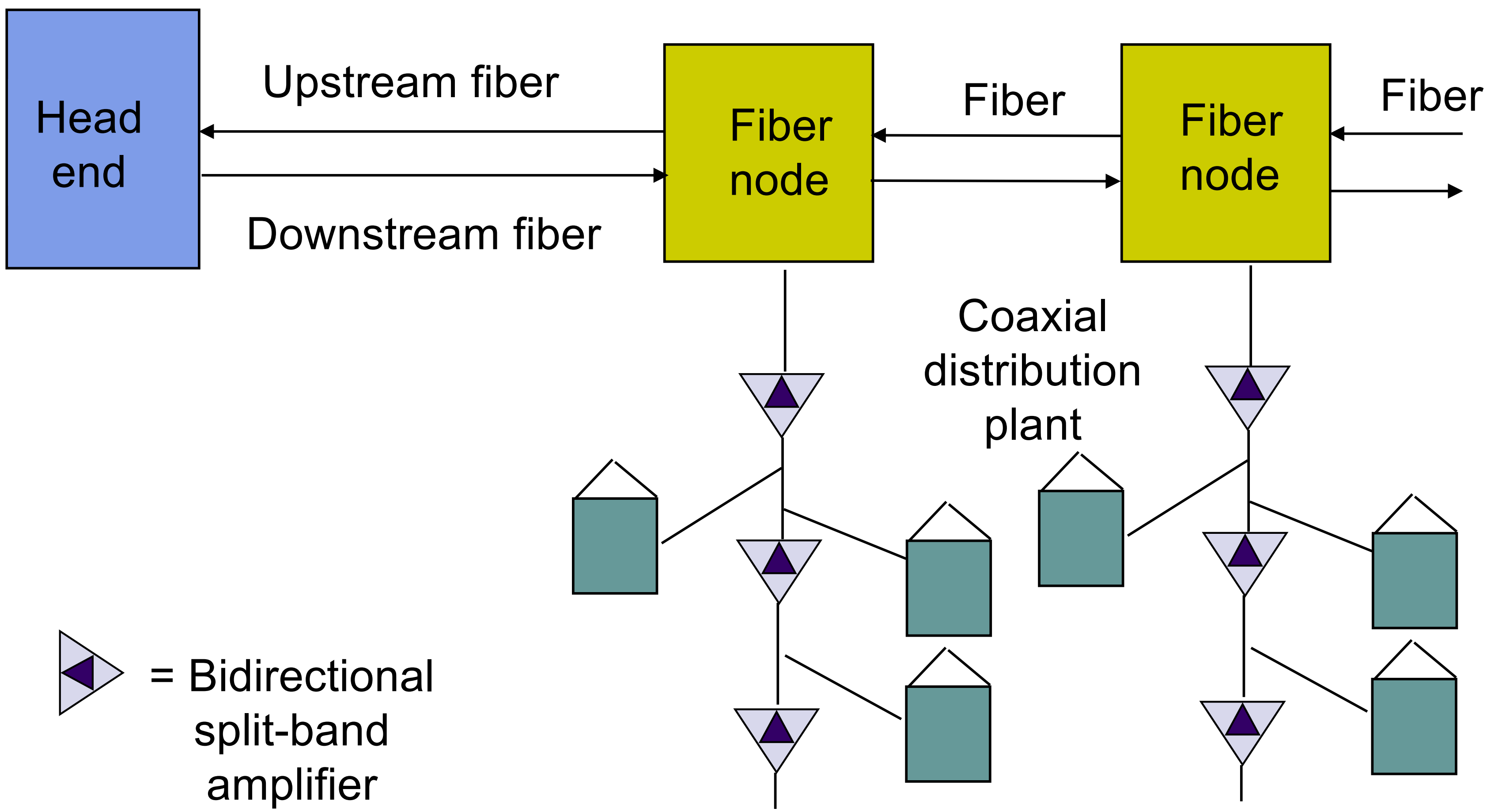


Cable Modem & TV Spectrum

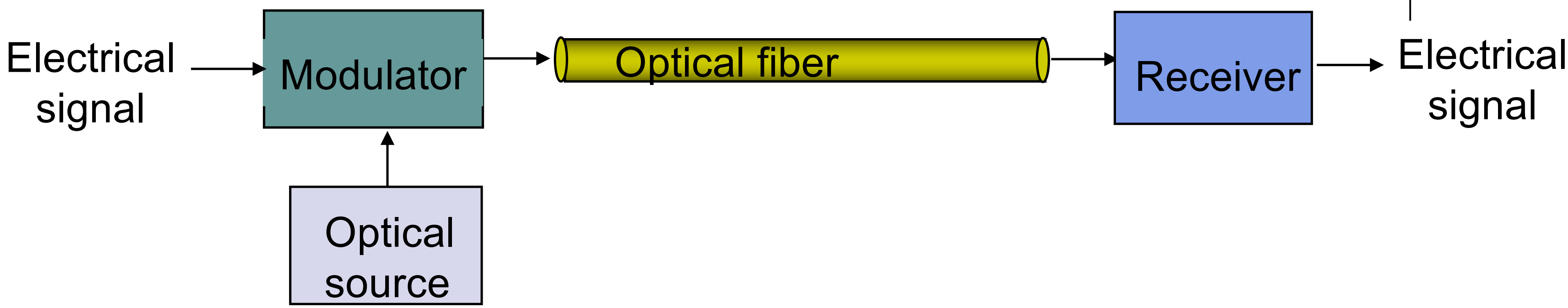


- Cable TV network originally unidirectional
- Cable plant needs upgrade to bidirectional
- 1 analog TV channel is 6 MHz, can support very high data rates
- Cable Modem: *shared* upstream & downstream
  - 5-42 MHz upstream into network; 2 MHz channels; 500 kbps to 4 Mbps
  - >550 MHz downstream from network; 6 MHz channels; 36 Mbps

Cable Network Topology



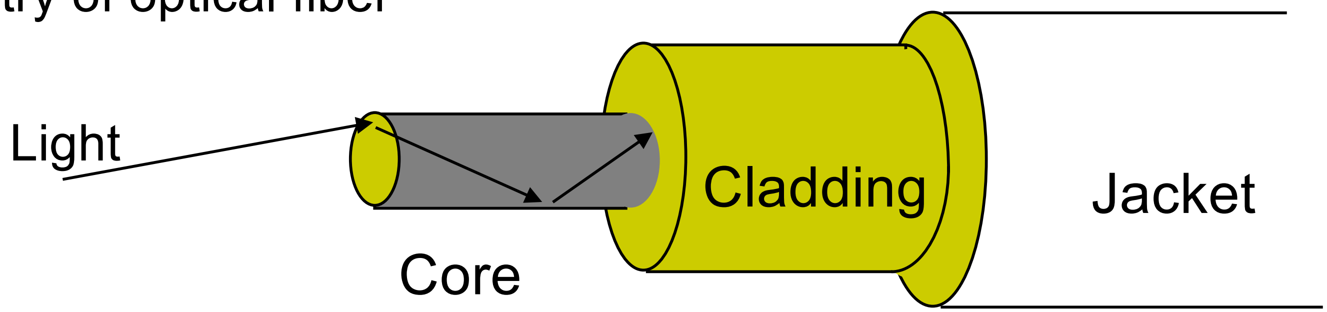
Optical Fiber



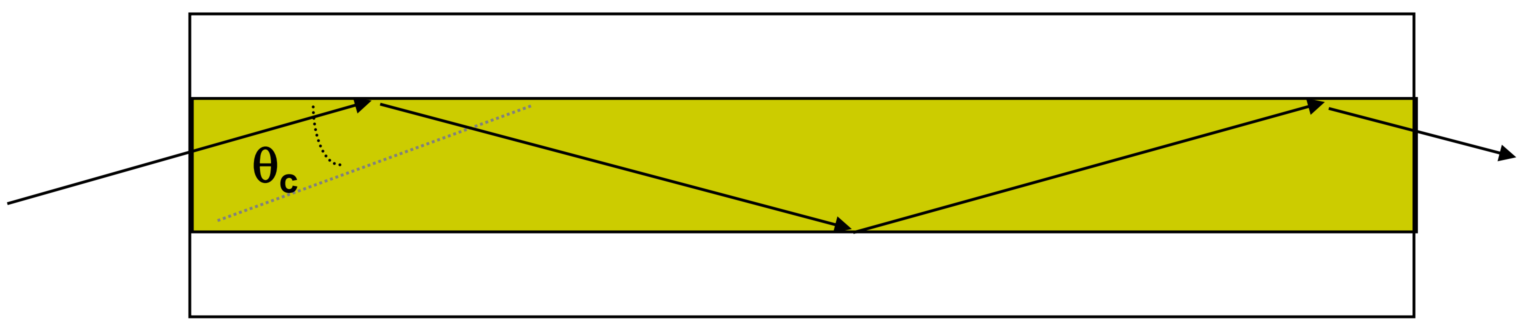
- Light sources (lasers, LEDs) generate pulses of light that are transmitted on optical fiber
  - Very long distances (>1000 km)
  - Very high speeds (>40 Gbps/wavelength)
  - Nearly error-free (BER of 10^{-15})
- Profound influence on network architecture
  - Dominates long distance transmission
  - Distance less of a cost factor in communications
  - Plentiful bandwidth for new services

Transmission in Optical Fiber

Geometry of optical fiber



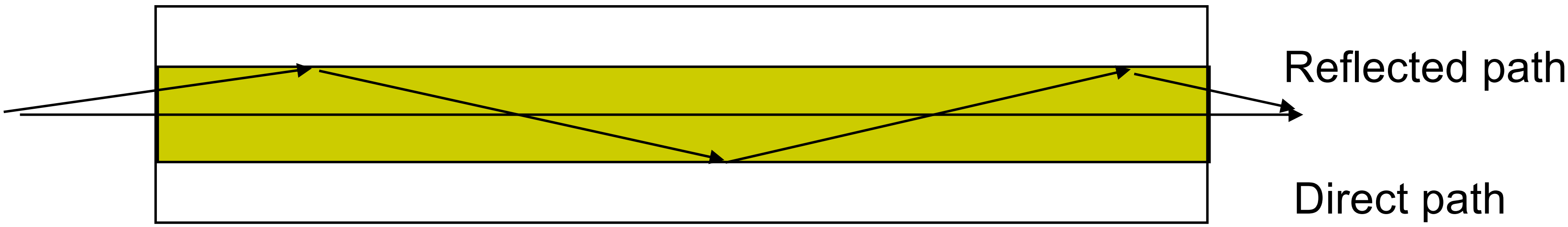
Total Internal Reflection in optical fiber



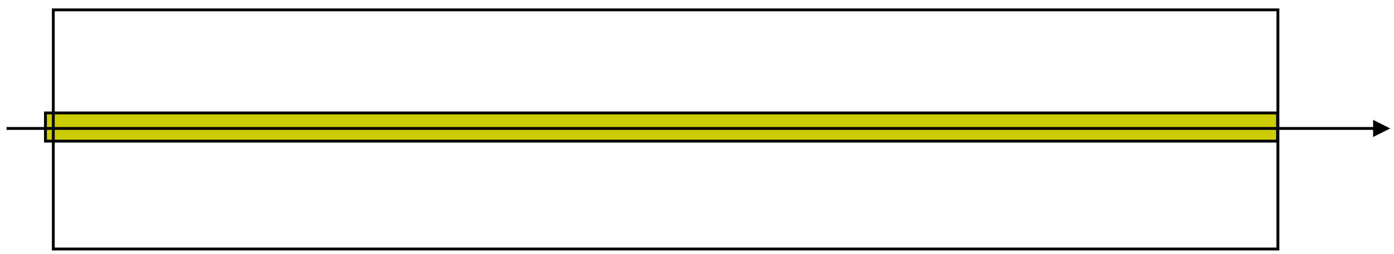
- Very fine glass cylindrical core surrounded by concentric layer of glass (cladding)
- Core has higher index of refraction than cladding
- Light rays incident at less than critical angle θ_c is completely reflected back into the core

Multimode & Single-mode Fiber

Multimode fiber: multiple rays follow different paths



Single-mode fiber: only direct path propagates in fiber



- Multimode: Thicker core, shorter reach
  - Rays on different paths interfere causing dispersion & limiting bit rate
- Single mode: Very thin core supports only one mode (path)
  - More expensive lasers, but achieves very high speeds

Optical Fiber Properties

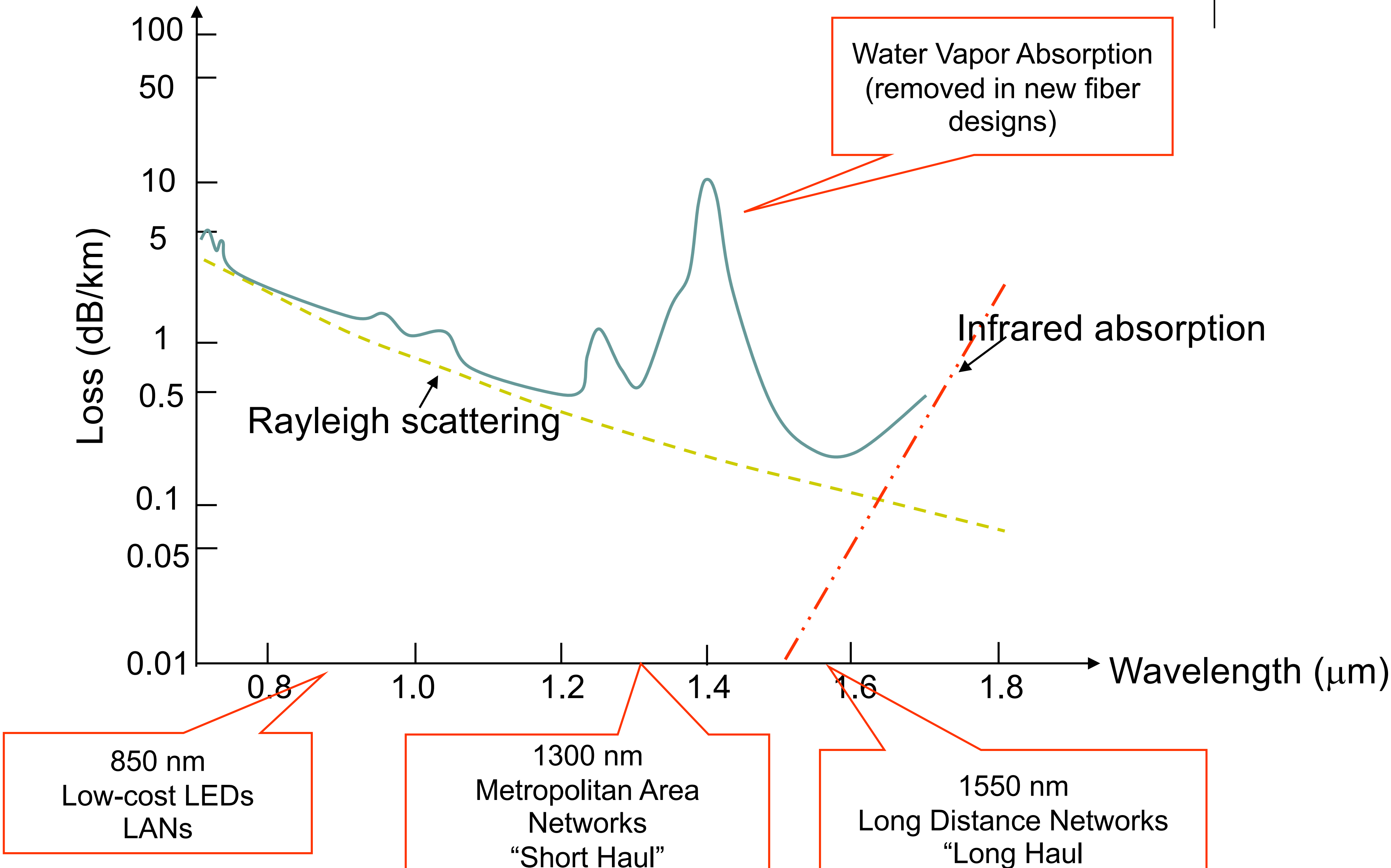
Advantages

- ***Very low attenuation***
- ***Noise immunity***
- ***Extremely high bandwidth***
- Security: Very difficult to tap without breaking
- No corrosion
- More compact & lighter than copper wire

Disadvantages

- New types of optical signal impairments & dispersion
  - Polarization dependence
  - Wavelength dependence
- Limited bend radius
  - If physical arc of cable too high, light lost or won't reflect
  - Will break
- Difficult to splice
- Mechanical vibration becomes signal noise

Very Low Attenuation



Huge Available Bandwidth

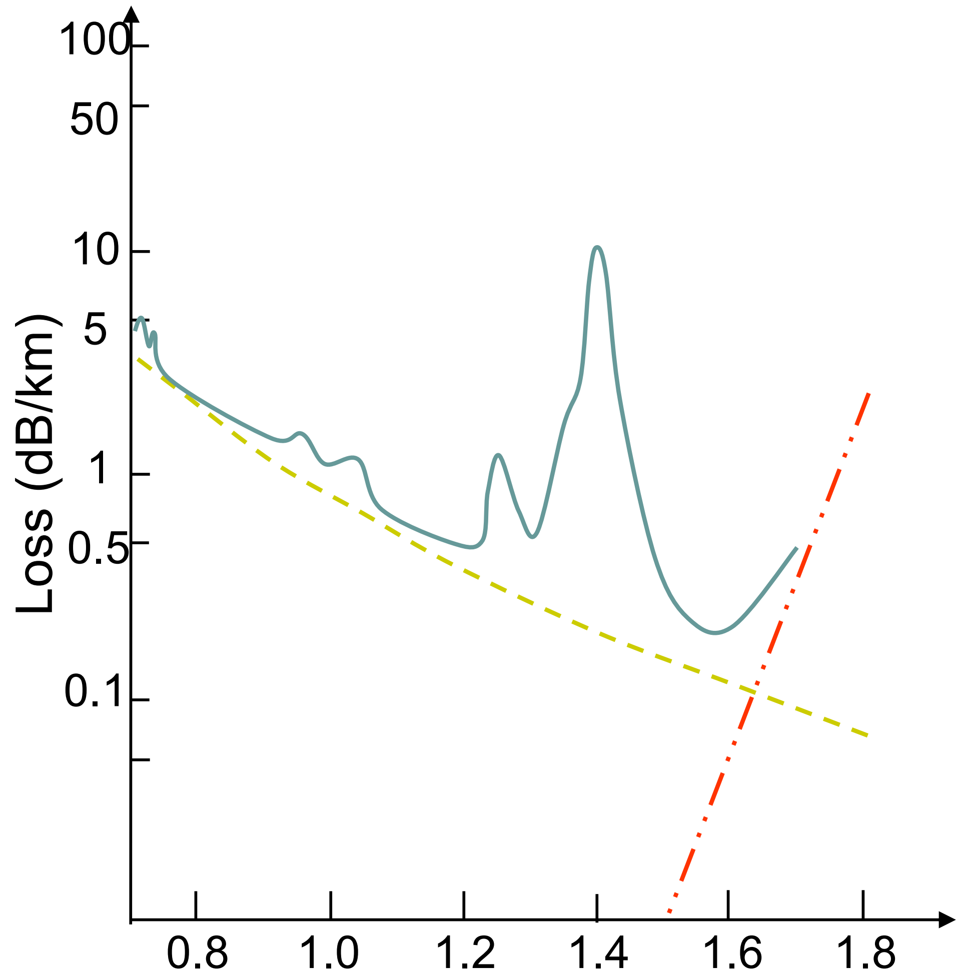
- Optical range from λ_1 to $\lambda_1 + \Delta\lambda$ contains bandwidth

$$B = f_1 - f_2 = \frac{v}{\lambda_1} - \frac{v}{\lambda_1 + \Delta\lambda}$$

$$= \frac{v}{\lambda_1}\left\{\frac{\Delta\lambda / \lambda_1}{1 + \Delta\lambda / \lambda 1}\right\} \approx \frac{v\,\Delta\lambda}{\lambda_1^2}$$

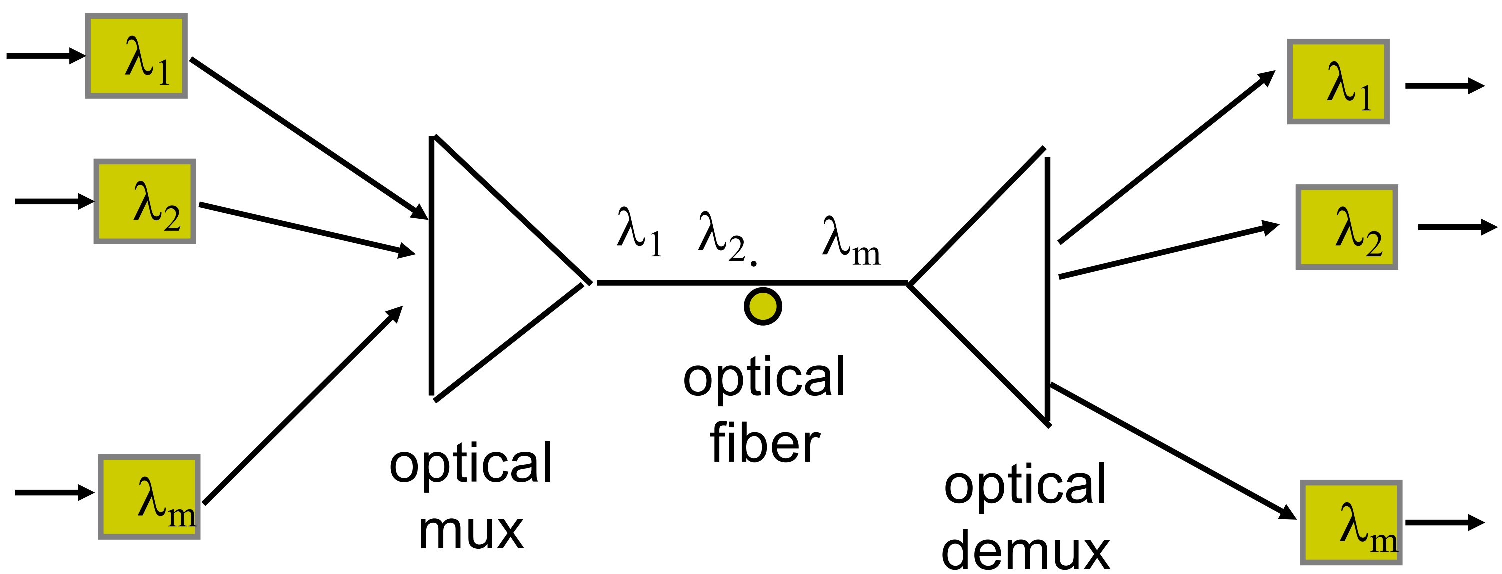
- Example: $\lambda_1 = 1450$ nm
  $\lambda_1 + \Delta\lambda = 1650$ nm:

$$B = \frac{2(10^8)\text{m/s } 200\text{nm}}{(1450\text{ nm})^2} \approx 19\text{ THz}$$

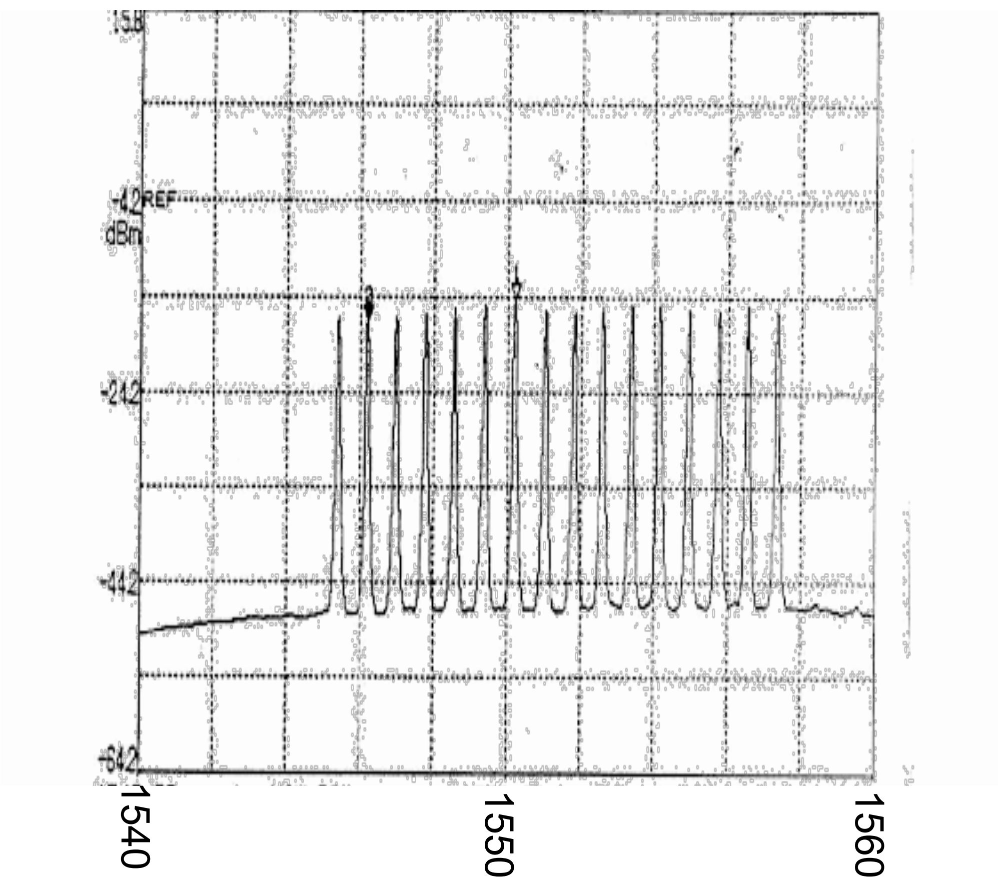


Wavelength-Division Multiplexing

- Different wavelengths carry separate signals
- Multiplex into shared optical fiber
- Each wavelength like a separate circuit
- A single fiber can carry 160 wavelengths, 10 Gbps per wavelength: 1.6 Tbps!



Coarse & Dense WDM



Coarse WDM

- Few wavelengths 4-8 with very wide spacing
- Low-cost, simple

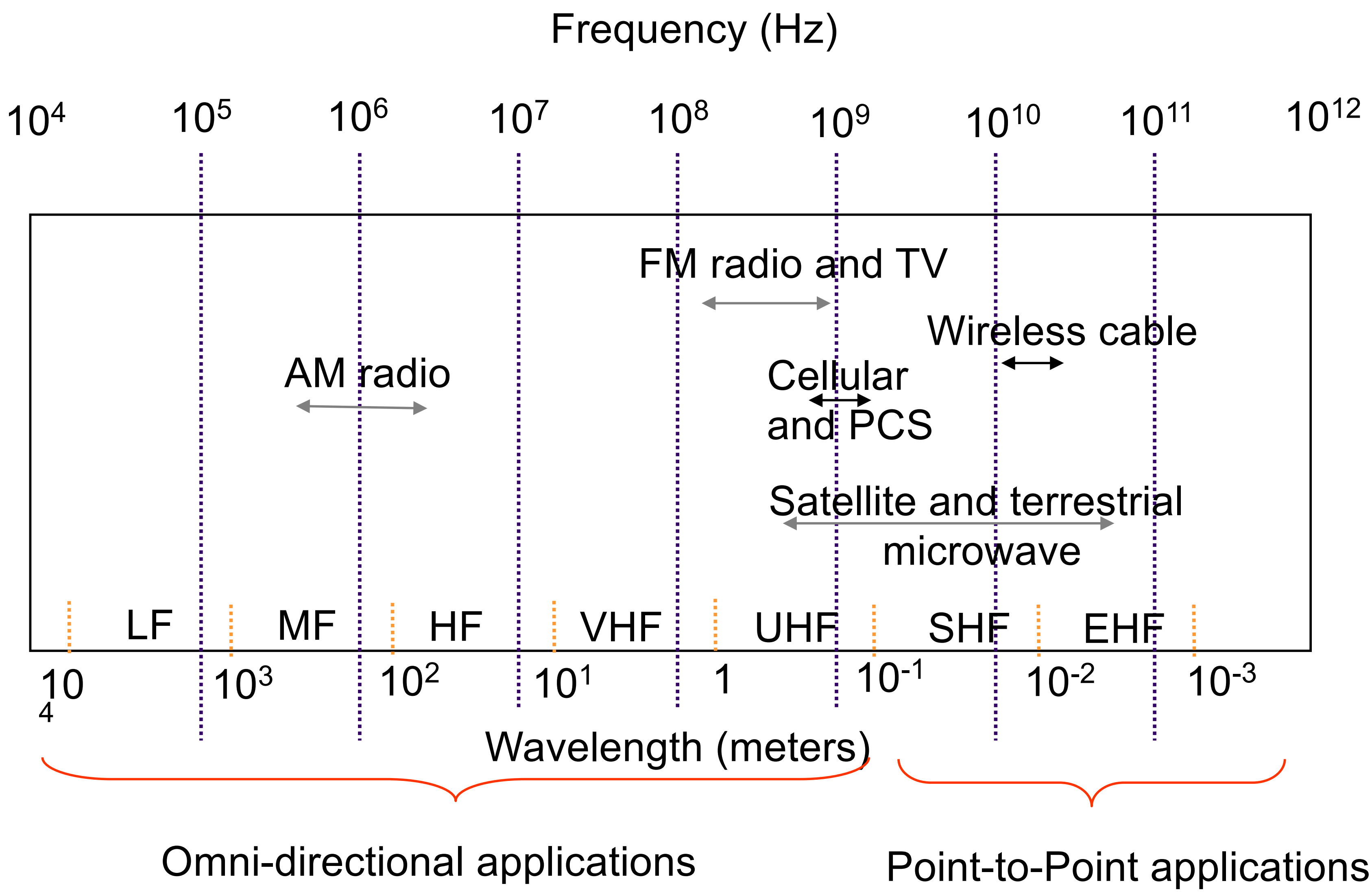
Dense WDM

- Many tightly-packed wavelengths
- ITU Grid: 0.8 nm separation for 10Gbps signals
- 0.4 nm for 2.5 Gbps

Radio Transmission

- Radio signals: antenna transmits sinusoidal signal ("carrier") that radiates in air/space
- Information embedded in carrier signal using modulation, e.g. QAM
- Communications without tethering
  - Cellular phones, satellite transmissions, Wireless LANs
- Multipath propagation causes fading
- Interference from other users
- Spectrum regulated by national & international regulatory organizations

Radio Spectrum



Examples

Cellular Phone

- Allocated spectrum
- First generation:
  - 800, 900 MHz
  - Initially analog voice
- Second generation:
  - 1800-1900 MHz
  - Digital voice, messaging

Wireless LAN

- Unlicenced ISM spectrum
  - Industrial, Scientific, Medical
  - 902-928 MHz, 2.400-2.4835 GHz, 5.725-5.850 GHz
- IEEE 802.11 LAN standard
  - 11-54 Mbps

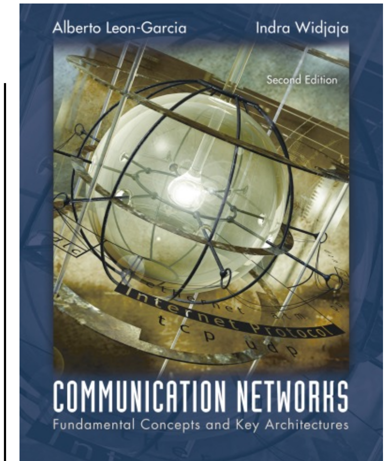
Point-to-Multipoint Systems

- Directional antennas at microwave frequencies
- High-speed digital communications between sites
- High-speed Internet Access Radio backbone links for rural areas

Satellite Communications

- Geostationary satellite @ 36000 km above equator
- Relays microwave signals from uplink frequency to downlink frequency
- Long distance telephone
- Satellite TV broadcast

Chapter 3
Digital Transmission Fundamentals



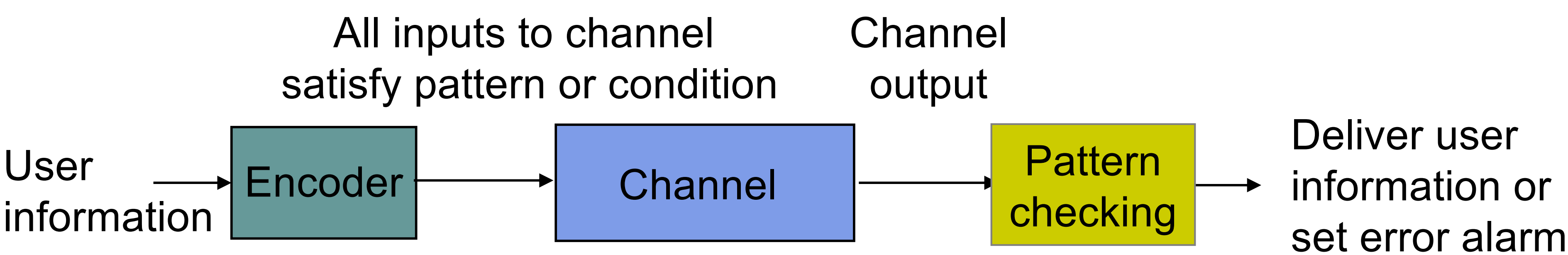
Error Detection and Correction

Error Control

- Digital transmission systems introduce errors
- Applications require certain reliability level
  - Data applications require error-free transfer
  - Voice & video applications tolerate some errors
- Error control used when transmission system does *not* meet application requirement
- Error control ensures a data stream is transmitted to a certain level of accuracy despite errors
- Two basic approaches:
  - Error ***detection*** & retransmission (ARQ)
  - Forward error ***correction*** (FEC)

Key Idea

- All transmitted data blocks ("codewords") satisfy a pattern
- If received block doesn't satisfy pattern, it is in error
- Redundancy: Only a subset of all possible blocks can be codewords
- Blindspot: when channel transforms a codeword into another codeword



Single Parity Check

- Append an overall parity check to *k* information bits

Info Bits: $b_1, b_2, b_3, \ldots, b_k$

Check Bit: $b_{k+1} = b_1 + b_2 + b_3 + \ldots + b_k$ modulo 2

Codeword: $(b_1, b_2, b_3, \ldots, b_{k,}, b_{k+!})$

- All codewords have even # of 1s
- Receiver checks to see if # of 1s is even
  - All error patterns that change an odd # of bits are detectable
  - All even-numbered patterns are undetectable
- Parity bit used in ASCII code

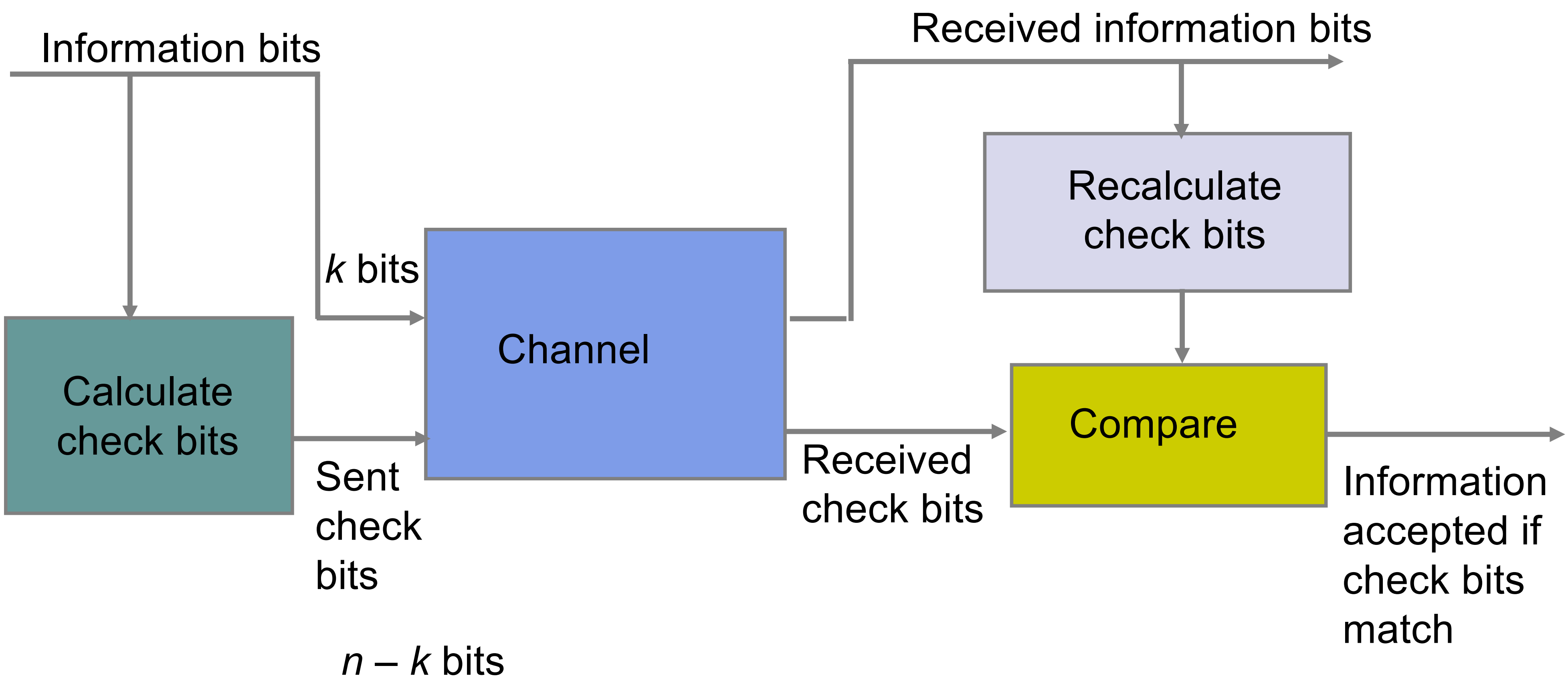
Example of Single Parity Code

- Information (7 bits): (0, 1, 0, 1, 1, 0, 0)
- Parity Bit: $b_8 = 0 + 1 + 0 + 1 + 1 + 0 = 1$
- Codeword (8 bits): (0, 1, 0, 1, 1, 0, 0, 1)

- If single error in bit 3 : (0, 1, 1, 1, 1, 0, 0, 1)
  - # of 1's =5, odd
  - Error detected

- If errors in bits 3 and 5: (0, 1, 1, 1, 0, 0, 0, 1)
  - # of 1's =4, even
  - Error not detected

Checkbits & Error Detection



How good is the single parity check code?

- *Redundancy*: Single parity check code adds 1 redundant bit per k information bits: overhead = $1/(k + 1)$
- *Coverage*: all error patterns with odd # of errors can be detected
  - An error patten is a binary $(k + 1)$-tuple with 1s where errors occur and 0's elsewhere
  - Of 2^{k+1} binary $(k + 1)$-tuples, ½ are odd, so 50% of error patterns can be detected
- Is it possible to detect more errors if we add more check bits?
- Yes, with the right codes

What if bit errors are random?

- Many transmission channels introduce bit errors at random, independently of each other, and with probability p
- Some error patterns are more probable than others:

$$P[10000000] = p(1-p)^7 = (1-p)^8 \left(\frac{p}{1-p}\right) \text{ and}$$

$$P[11000000] = p^2(1-p)^6 = (1-p)^8 \left(\frac{p}{1-p}\right)^2$$

- In any worthwhile channel $p < 0.5$, and so $(p/(1-p)) < 1$
- It follows that patterns with 1 error are more likely than patterns with 2 errors and so forth
- What is the probability that an undetectable error pattern occurs?

Single parity check code with random bit errors

- Undetectable error pattern if even # of bit errors:

$$P[\text{error detection failure}] = P[\text{undetectable error pattern}]$$
$$= P[\text{error patterns with even number of 1s}]$$

$$= \binom{n}{2} p^2(1-p)^{n-2} + \binom{n}{4} p^4(1-p)^{n-4} + \ldots$$

- Example: Evaluate above for $n = 32$, $p = 10^{-3}$

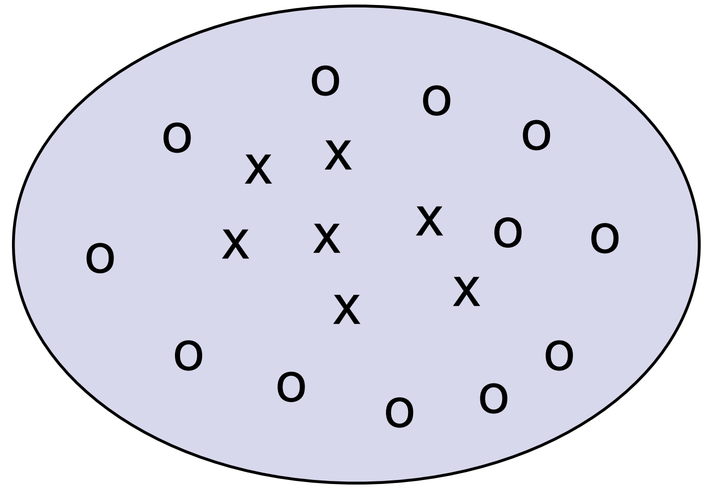
$$P[\text{undetectable error}] = \binom{32}{2} (10^{-3})^2 (1-10^{-3})^{30} + \binom{32}{4} (10^{-3})^4 (1-10^{-3})^{28}$$

$$\approx 496\,(10^{-6}) + 35960\,(10^{-12}) \approx 4.96\,(10^{-4})$$

- For this example, roughly 1 in 2000 error patterns is undetectable

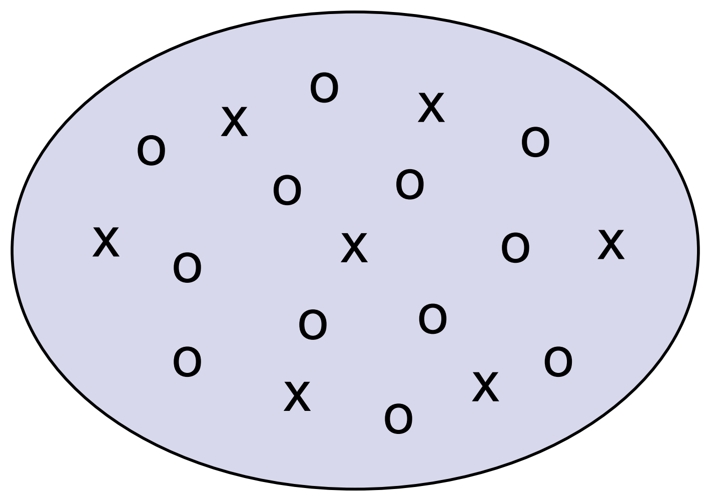
What is a good code?

- Many channels have preference for error patterns that have fewer # of errors
- These error patterns map transmitted codeword to nearby *n*-tuple
- If codewords close to each other then detection failures will occur
- Good codes should maximize separation between codewords



Poor distance properties

x = codewords
o = **noncodewords**



Good distance properties

Two-Dimensional Parity Check

- More parity bits to improve coverage
- Arrange information as columns
- Add single parity bit to each column
- Add a final “parity” column
- Used in early error control systems

$$
\begin{array}{ccccc|c}
1 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 1 & 0 & 0 \\
1 & 1 & 0 & 1 & 1 & 0 \\
\hline
1 & 0 & 0 & 1 & 1 & 1
\end{array}
$$

Last column consists of check bits for each row

Bottom row consists of check bit for each column

Error-detecting capability

1	0	0	1	0	0
0	0	0	0	0	1
1	0	0	1	0	0
1	1	0	1	1	0
1	0	0	1	1	1

One error

1	0	0	1	0	0
0	0	0	0	0	1
1	0	0	1	0	0
1	0	0	1	1	0
1	0	0	1	1	1

Two errors

1, 2, or 3 errors can always be detected; Not all patterns >4 errors can be detected

1	0	0	1	0	0
0	0	0	1	0	1
1	0	0	1	0	0
1	0	0	1	1	0
1	0	0	1	1	1

Three errors

1	0	0	1	0	0
0	0	0	1	0	1
1	0	0	1	0	0
1	0	0	0	1	0
1	0	0	1	1	1

Four errors (undetectable)

Arrows indicate failed check bits

Other Error Detection Codes

- Many applications require very low error rate
- Need codes that detect the vast majority of errors
- Single parity check codes do not detect enough errors
- Two-dimensional codes require too many check bits
- The following error detecting codes used in practice:
  - Internet Check Sums
  - CRC Polynomial Codes

Internet Checksum

- Several Internet protocols (e.g. IP, TCP, UDP) use check bits to detect errors in the *IP header* (or in the header and data for TCP/UDP)
- A checksum is calculated for header contents and included in a special field.
- Checksum recalculated at every router, so algorithm selected for ease of implementation in software
- Let header consist of L, 16-bit words,

  $\mathbf{b}_0, \mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_{L-1}$
- The algorithm appends a 16-bit checksum $\mathbf{b}_L$

Checksum Calculation

The checksum $\mathbf{b}_L$ is calculated as follows:

- Treating each 16-bit word as an integer, find

$$\mathbf{x} = \mathbf{b}_0 + \mathbf{b}_1 + \mathbf{b}_2 + \ldots + \mathbf{b}_{L-1} \text{ modulo } 2^{16}-1$$

- The checksum is then given by:

$$\mathbf{b}_L = -\,\mathbf{x} \quad \text{modulo } 2^{16}-1$$

Thus, the headers must satisfy the following ***pattern***:

$$\mathbf{0} = \mathbf{b}_0 + \mathbf{b}_1 + \mathbf{b}_2 + \ldots + \mathbf{b}_{L-1} + \mathbf{b}_L \text{ modulo } 2^{16}-1$$

- The checksum calculation is carried out in software using one’s complement arithmetic

Internet Checksum Example

Use Modulo Arithmetic

- Assume 4-bit words
- Use mod 2^4-1 arithmetic
- $\underline{b}_0$=1100 = 12
- $\underline{b}_1$=1010 = 10
- $\underline{b}_0$+$\underline{b}_1$=12+10=7 mod15
- $\underline{b}_2$ = -7 = 8 mod15
- Therefore
- $\underline{b}_2$=1000

Use Binary Arithmetic

- Note 16 =1 mod15
- So: 10000 = 0001 mod15
- leading bit wraps around

$b_0 + b_1$ = 1100+1010
=10110
=10000+0110
=0001+0110
=0111
=7

Take 1s complement

b_2 = -0111 =1000

Polynomial Codes

- Polynomials instead of vectors for codewords
- Polynomial arithmetic instead of check sums
- Implemented using shift-register circuits
- Also called *cyclic redundancy check (CRC)* codes
- Most data communications standards use polynomial codes for error detection
- Polynomial codes also basis for powerful error-correction methods

Binary Polynomial Arithmetic

- Binary vectors map to polynomials

$$(i_{k-1}, i_{k-2}, \ldots, i_2, i_1, i_0) \rightarrow i_{k-1}x^{k-1} + i_{k-2}x^{k-2} + \ldots + i_2x^2 + i_1x + i_0$$

Addition:

$$(x^7 + x^6 + 1) + (x^6 + x^5) = x^7 + x^6 + x^6 + x^5 + 1$$

$$= x^7 + (1+1)x^6 + x^5 + 1$$

$$= x^7 + x^5 + 1 \quad \text{since } 1+1=0 \text{ mod2}$$

Multiplication:

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

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

$$= x^3 + 1$$

Binary Polynomial Division

- Division with Decimal Numbers

$$
\begin{array}{rl}
 & 34 \quad \leftarrow \text{quotient} \\ \hline
35\,) & 1222 \quad \leftarrow \text{dividend} \\
 & 105 \\ \hline
 & 172 \\
 & 140 \\ \hline
 & 32 \quad \leftarrow \text{remainder}
\end{array}
$$

divisor

dividend = quotient x divisor +remainder

$$1222 = 34 \times 35 + 32$$

- Polynomial Division

$$
\begin{array}{rl}
 & x^3 + x^2 + x \qquad = q(x) \ \text{quotient} \\ \hline
x^3 + x + 1\,) & x^6 + x^5 \\
 & x^6 + \phantom{x^5 +{}} x^4 + x^3 \\ \hline
 & \phantom{x^6 +{}} x^5 + x^4 + x^3 \\
 & \phantom{x^6 +{}} x^5 + \phantom{x^4 +{}} x^3 + x^2 \\ \hline
 & \phantom{x^6 + x^5 +{}} x^4 + \phantom{x^3 +{}} x^2 \\
 & \phantom{x^6 + x^5 +{}} x^4 + \phantom{x^3 +{}} x^2 + x \\ \hline
 & \phantom{x^6 + x^5 + x^4 + x^3 + x^2 +{}} x \qquad = r(x) \ \text{remainder}
\end{array}
$$

divisor

dividend

Note: Degree of r(x) is less than degree of divisor

Polynomial Coding

- Code has binary *generating polynomial* of degree $n\text{–}k$

$$g(x) = x^{n-k} + g_{n-k-1}x^{n-k-1} + \ldots + g_2x^2 + g_1x + 1$$

- *k information bits* define polynomial of degree $k – 1$

$$i(x) = i_{k-1}x^{k-1} + i_{k-2}x^{k-2} + \ldots + i_2x^2 + i_1x + i_0$$

- Find *remainder polynomial* of at most degree $n – k – 1$

$$g(x)\ \overset{\displaystyle q(x)}{\overline{\big)\ x^{n-k}\, i(x)}} \qquad \qquad x^{n-k}i(x) = q(x)g(x) + r(x)$$
$$r(x)$$

- Define the *codeword polynomial* of degree $n – 1$

$$\boxed{\underbrace{b(x)}_{n \text{ bits}} = \underbrace{x^{n-k}i(x)}_{k \text{ bits}} + \underbrace{r(x)}_{n-k \text{ bits}}}$$

Polynomial example: k = 4, $n–k$ = 3

Generator polynomial: $g(x) = x^3 + x + 1$

Information: (1,1,0,0) $\quad i(x) = x^3 + x^2$

Encoding: $x^3 i(x) = x^6 + x^5$

$$
\begin{array}{r l}
 & x^3 + x^2 + x \\
\hline
x^3 + x + 1 \;) & x^6 + x^5 \\
 & x^6 + \; x^4 + x^3 \\
\hline
 & \; x^5 + x^4 + x^3 \\
 & \; x^5 + \; x^3 + x^2 \\
\hline
 & \; x^4 + \; x^2 \\
 & \; x^4 + \; x^2 + x \\
\hline
 & \; x
\end{array}
$$

$$
\begin{array}{r l}
 & 1110 \\
\hline
1011 \;) & 1100000 \\
 & 1011 \\
\hline
 & 1110 \\
 & 1011 \\
\hline
 & 1010 \\
 & 1011 \\
\hline
 & 010
\end{array}
$$

Transmitted codeword:

$b(x) = x^6 + x^5 + x$

$\Longrightarrow \quad \underline{b} = (1,1,0,0,0,1,0)$

The *Pattern* in Polynomial Coding

- All codewords satisfy the following **pattern**:

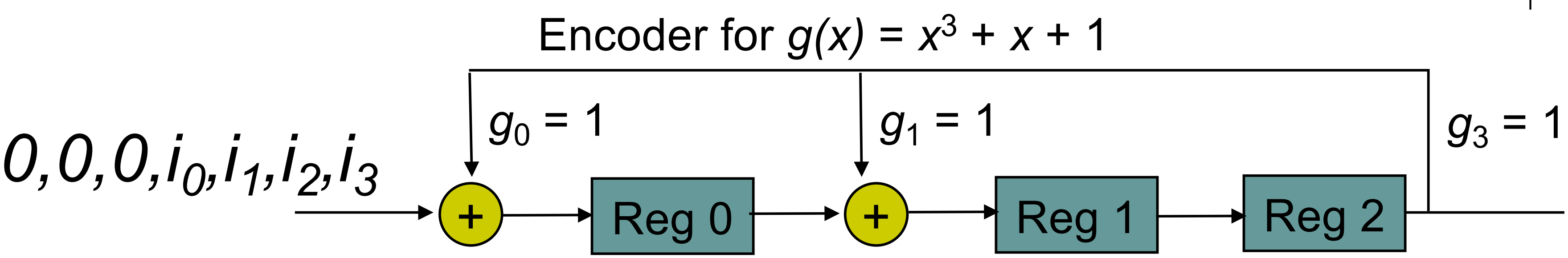
$$b(x) = x^{n-k}i(x) + r(x) = q(x)g(x) + r(x) + r(x) = q(x)g(x)$$

- All codewords are a multiple of *g(x)!*
- Receiver should divide received n-tuple by g(x) and check if remainder is zero
- If remainder is nonzero, then received n-tuple is not a codeword

Shift-Register Implementation

1. Accept information bits $i_{k-1}, i_{k-2}, \ldots, i_2, i_1, i_0$
2. Append $n - k$ zeros to information bits
3. Feed sequence to shift-register circuit that performs polynomial division
4. After n shifts, the shift register contains the remainder

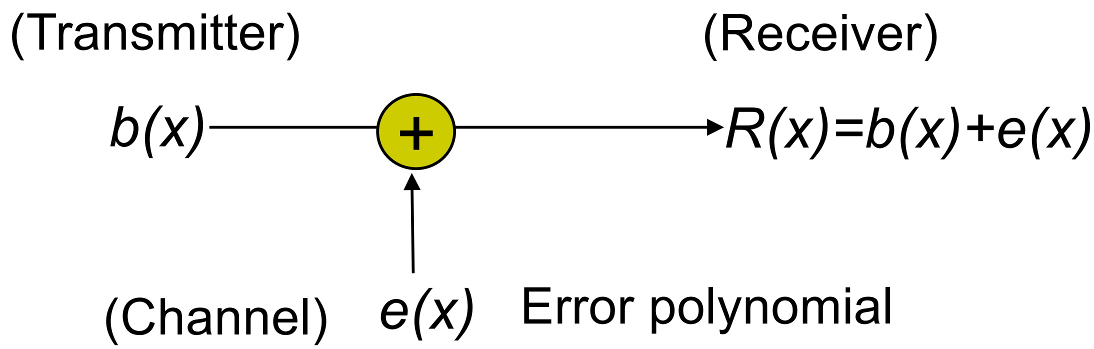
Division Circuit



Clock	Input	Reg 0	Reg 1	Reg 2
0	-	0	0	0
1	$1 = i_3$	1	0	0
2	$1 = i_2$	1	1	0
3	$0 = i_1$	0	1	1
4	$0 = i_0$	1	1	1
5	0	1	0	1
6	0	1	0	0
7	0	**0**	**1**	**0**
	Check bits:	$\boldsymbol{r_0 = 0}$	$\boldsymbol{r_1 = 1}$	$\boldsymbol{r_2 = 0}$

$\Longrightarrow$ $\boldsymbol{r(x) = x}$

Undetectable error patterns



- $e(x)$ has 1s in error locations & 0s elsewhere
- Receiver divides the received polynomial $R(x)$ by $g(x)$
- Blindspot: If $e(x)$ is a multiple of $g(x)$, that is, $e(x)$ is a nonzero codeword, then

$$R(x) = b(x) + e(x) = q(x)g(x) + q'(x)g(x)$$

- *The set of undetectable error polynomials is the set of nonzero code polynomials*
- *Choose the generator polynomial so that selected error patterns can be detected.*

Designing good polynomial codes

- Select generator polynomial so that likely error patterns are not multiples of $g(x)$
- *Detecting Single Errors*
  - $e(x) = x^i$ for error in location $i + 1$
  - If $g(x)$ has more than 1 term, it cannot divide x^i
- *Detecting Double Errors*
  - $e(x) = x^i + x^j = x^i(x^{j-i}+1)$ *where* $j>i$
  - If $g(x)$ has more than 1 term, it cannot divide x^i
  - If g(x) is a *primitive* polynomial, it cannot divide x^m+1 for all $m<2^{n-k}-1$ (Need to keep codeword length less than $2^{n-k}-1$)
  - Primitive polynomials can be found by consulting coding theory books

Designing good polynomial codes

- *Detecting Odd Numbers of Errors*
  - Suppose all codeword polynomials have an even # of 1s, then all odd numbers of errors can be detected
  - As well, $b(x)$ evaluated at $x = 1$ is zero because $b(x)$ has an even number of 1s
  - This implies $x + 1$ must be a factor of all $b(x)$
  - Pick $g(x) = (x + 1)\, p(x)$ where $p(x)$ is primitive

Standard Generator Polynomials

CRC = cyclic redundancy check

- CRC-8:

  $= x^8 + x^2 + x + 1$ ATM

- CRC-16:

  $= x^{16} + x^{15} + x^2 + 1$ Bisync

  $= (x + 1)(x^{15} + x + 1)$

- CCITT-16:

  $= x^{16} + x^{12} + x^5 + 1$ HDLC, XMODEM, V.41

- CCITT-32: IEEE 802, DoD, V.42

  $= x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$

Hamming Codes

- Class of *error-correcting* codes
- Capable of correcting all *single-error* patterns
- For each $m \geq 2$, there is a Hamming code of length $n = 2^m - 1$ with $n - k = m$ parity check bits

Redundancy

m	$n = 2^m - 1$	$k = n - m$	m/n
3	7	4	3/7
4	15	11	4/15
5	31	26	5/31
6	63	57	6/63

m = 3 Hamming Code

- Information bits are b_1, b_2, b_3, b_4
- Equations for parity checks b_5, b_6, b_7

$$
\begin{aligned}
b_5 &= b_1 \phantom{{}+b_2} + b_3 + b_4 \\
b_6 &= b_1 + b_2 \phantom{{}+b_3} + b_4 \\
b_7 &= + b_2 + b_3 + b_4
\end{aligned}
$$

- There are $2^4 = 16$ codewords
- (0,0,0,0,0,0,0) is a codeword

Hamming (7,4) code

Information				Codeword							Weight
b_1	b_2	b_3	b_4	b_1	b_2	b_3	b_4	b_5	b_6	b_7	$w(\underline{b})$
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1	1	1	1	4
0	0	1	0	0	0	1	0	1	0	1	3
0	0	1	1	0	0	1	1	0	1	0	3
0	1	0	0	0	1	0	0	0	1	1	3
0	1	0	1	0	1	0	1	1	0	0	3
0	1	1	0	0	1	1	0	1	1	0	4
0	1	1	1	0	1	1	1	0	0	1	4
1	0	0	0	1	0	0	0	1	1	0	3
1	0	0	1	1	0	0	1	0	0	1	3
1	0	1	0	1	0	1	0	0	1	1	4
1	0	1	1	1	0	1	1	1	0	0	4
1	1	0	0	1	1	0	0	1	0	1	4
1	1	0	1	1	1	0	1	0	1	0	4
1	1	1	0	1	1	1	0	0	0	0	3
1	1	1	1	1	1	1	1	1	1	1	7

Parity Check Equations

- Rearrange parity check equations:

$$0 = b_5 + b_5 = b_1 + b_3 + b_4 + b_5$$

$$0 = b_6 + b_6 = b_1 + b_2 + b_4 + b_6$$

$$0 = b_7 + b_7 = + b_2 + b_3 + b_4 + b_7$$

- In matrix form:

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \begin{matrix} = \\ = \\ = \end{matrix} \begin{bmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \\ b_7 \end{bmatrix} = \mathbf{H}\, \underline{b}^t = \underline{0}$$

- All codewords must satisfy these equations
- Note: each nonzero 3-tuple appears once as a column in **check matrix H**

Error Detection with Hamming Code

$$\underline{s} = \mathbf{H}\,\underline{e} = \begin{bmatrix} 1&0&1&1&1&0&0 \\ 1&1&0&1&0&1&0 \\ 0&1&1&1&0&0&1 \end{bmatrix} \begin{bmatrix} 0\\0\\1\\0\\0\\0\\0 \end{bmatrix} = \begin{bmatrix} 1\\0\\1 \end{bmatrix}$$

Single error detected

$$\underline{s} = \mathbf{H}\,\underline{e} = \begin{bmatrix} 1&0&1&1&1&0&0 \\ 1&1&0&1&0&1&0 \\ 0&1&1&1&0&0&1 \end{bmatrix} \begin{bmatrix} 0\\1\\0\\0\\1\\0\\0 \end{bmatrix} = \begin{bmatrix} 0\\1\\1 \end{bmatrix} + \begin{bmatrix} 1\\0\\0 \end{bmatrix} = \begin{bmatrix} 1\\1\\1 \end{bmatrix}$$

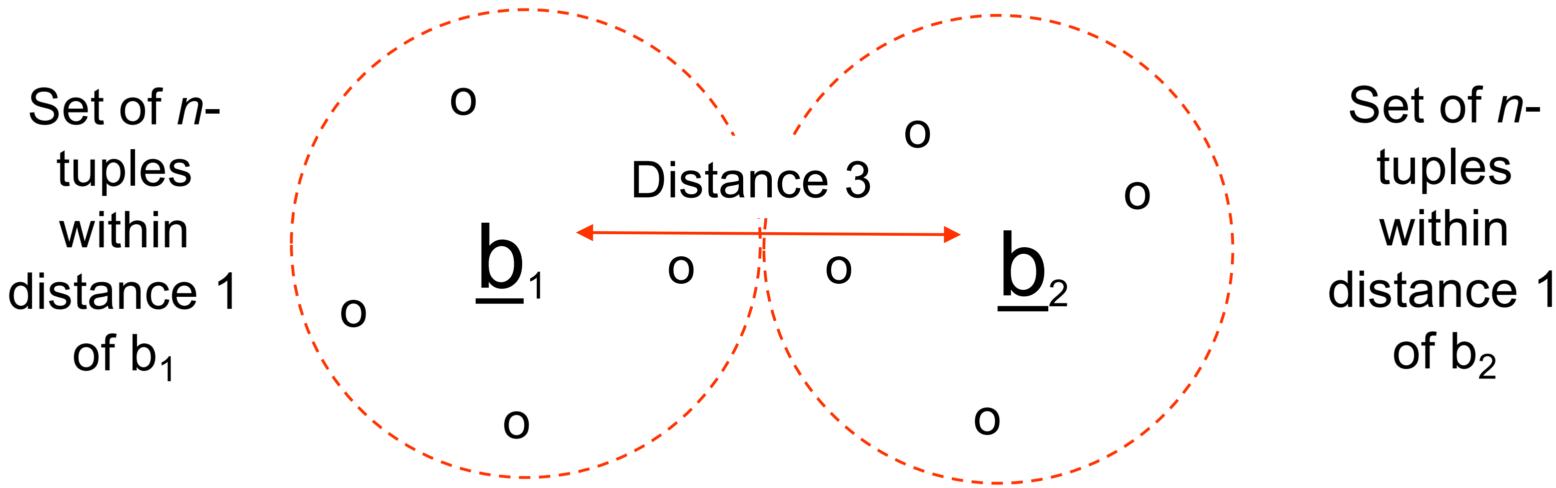
Double error detected

$$\underline{s} = \mathbf{H}\,\underline{e} = \begin{bmatrix} 1&0&1&1&1&0&0 \\ 1&1&0&1&0&1&0 \\ 0&1&1&1&0&0&1 \end{bmatrix} \begin{bmatrix} 1\\1\\1\\0\\0\\0\\0 \end{bmatrix} = \begin{bmatrix} 1\\1\\0 \end{bmatrix} + \begin{bmatrix} 0\\1\\1 \end{bmatrix} + \begin{bmatrix} 1\\0\\1 \end{bmatrix} = \underline{0}$$

Triple error not detected

Minimum distance of Hamming Code

- Previous slide shows that undetectable error pattern must have 3 or more bits
- At least 3 bits must be changed to convert one codeword into another codeword

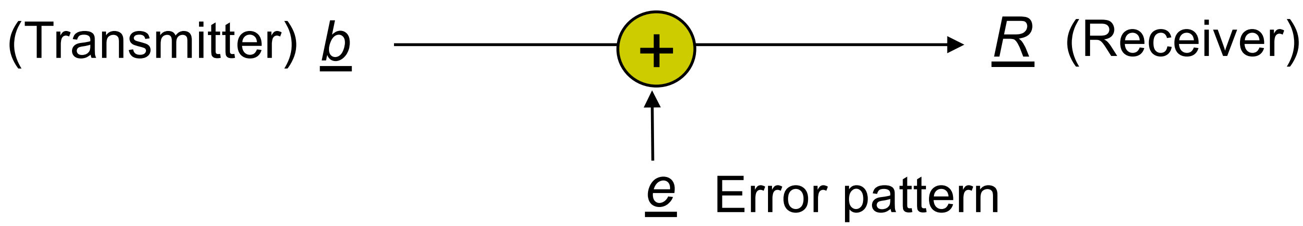


- Spheres of distance 1 around each codeword do not overlap
- If a single error occurs, the resulting *n*-tuple will be in a unique sphere around the original codeword

General Hamming Codes

- For $m \geq 2$, the Hamming code is obtained through the check matrix H:
  - Each nonzero m-tuple appears once as a column of H
  - The resulting code corrects all single errors
- For each value of m, there is a polynomial code with $g(x)$ of degree m that is equivalent to a Hamming code and corrects all single errors
  - For $m = 3$, $g(x) = x^3+x+1$

Error-correction using Hamming Codes



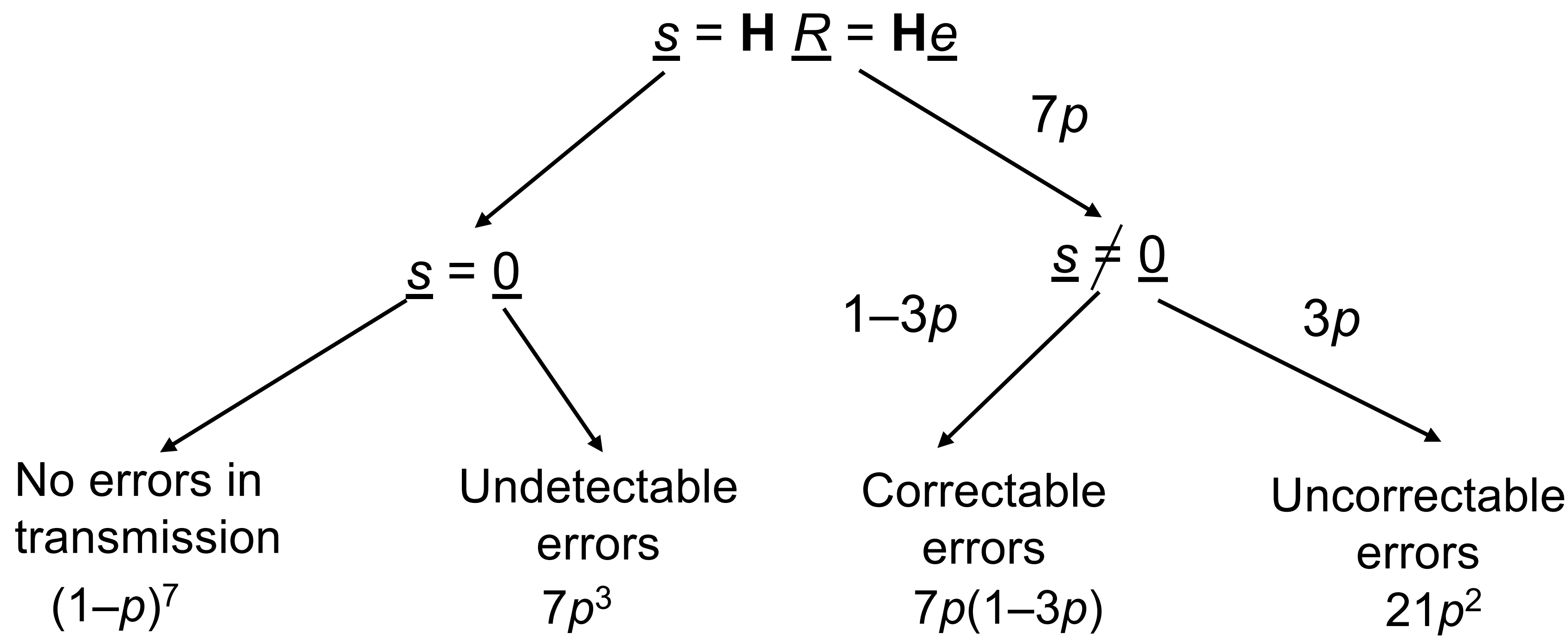
- The receiver first calculates the syndrome:

  $$\underline{s} = H\underline{R} = H(\underline{b} + \underline{e}) = H\underline{b} + H\underline{e} = H\underline{e}$$

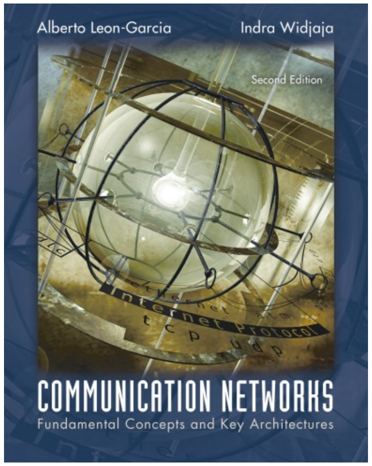
- If $\underline{s} = \underline{0}$, then the receiver accepts $\underline{R}$ as the transmitted codeword
- If $\underline{s}$ is nonzero, then an error is detected
  - Hamming decoder *assumes* a single error has occurred
  - Each single-bit error pattern has a unique syndrome
  - The receiver matches the syndrome to a single-bit error pattern and corrects the appropriate bit

Performance of Hamming Error-Correcting Code

- Assume bit errors occur independent of each other and with probability p



Chapter 3
Digital Transmission Fundamentals



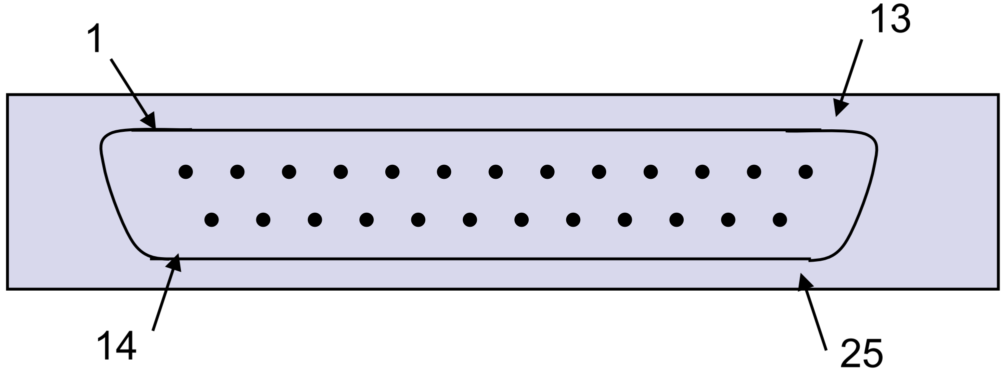
RS-232 Asynchronous Data Transmission

Recommended Standard (RS) 232

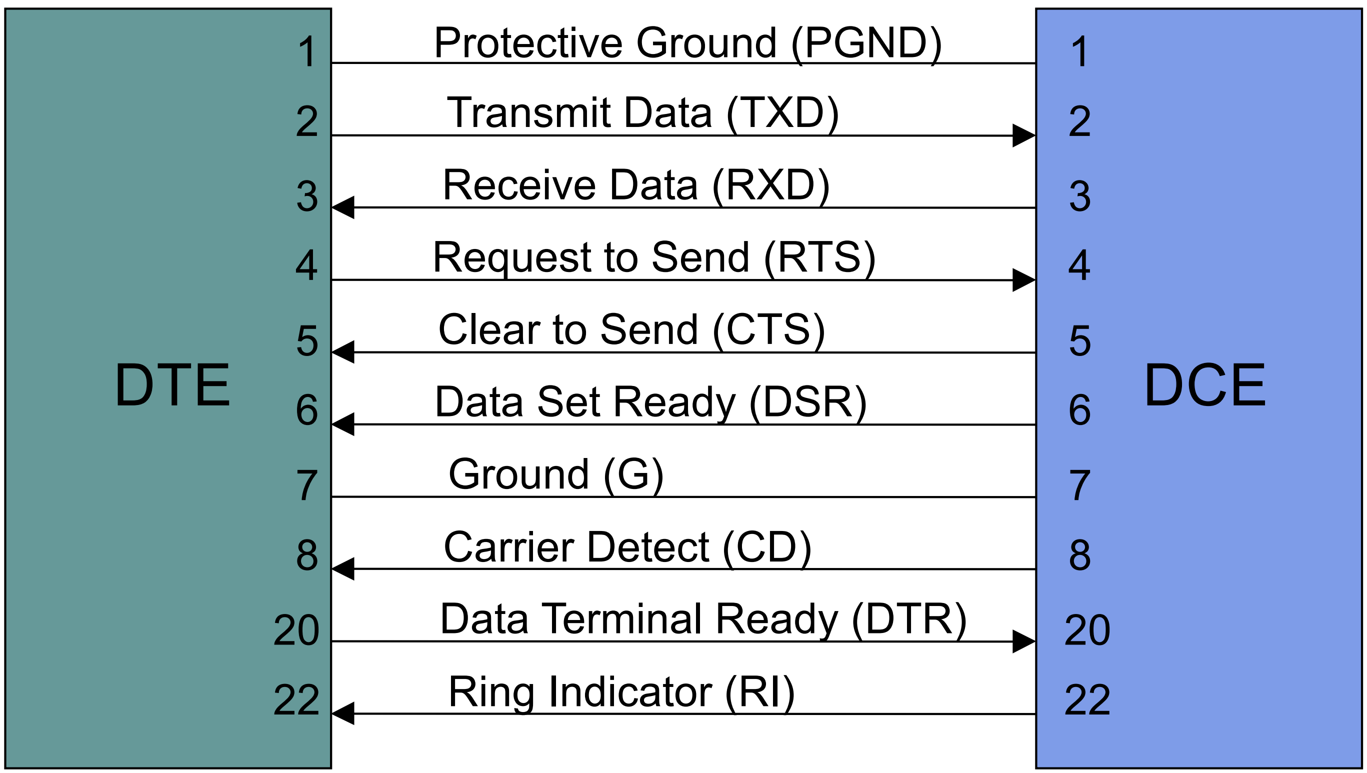
- Serial line interface between computer and modem or similar device
- Data Terminal Equipment (DTE): computer
- Data Communications Equipment (DCE): modem
- Mechanical and Electrical specification

Pins in RS-232 connector

(a)

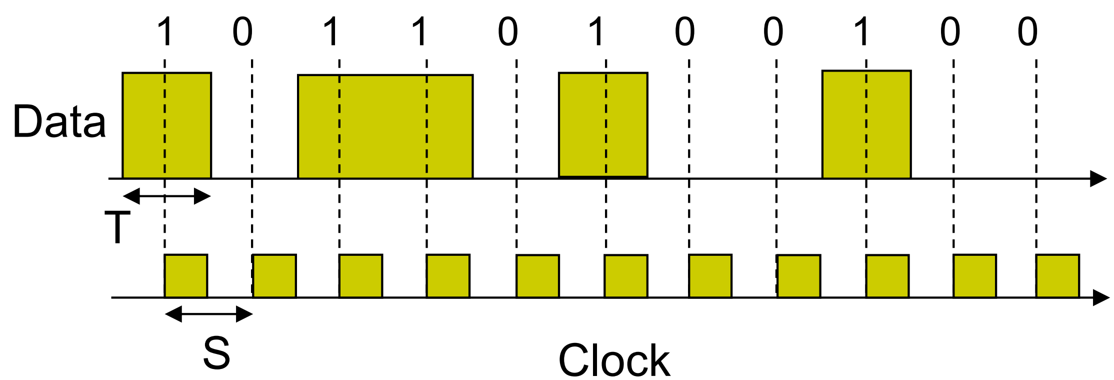


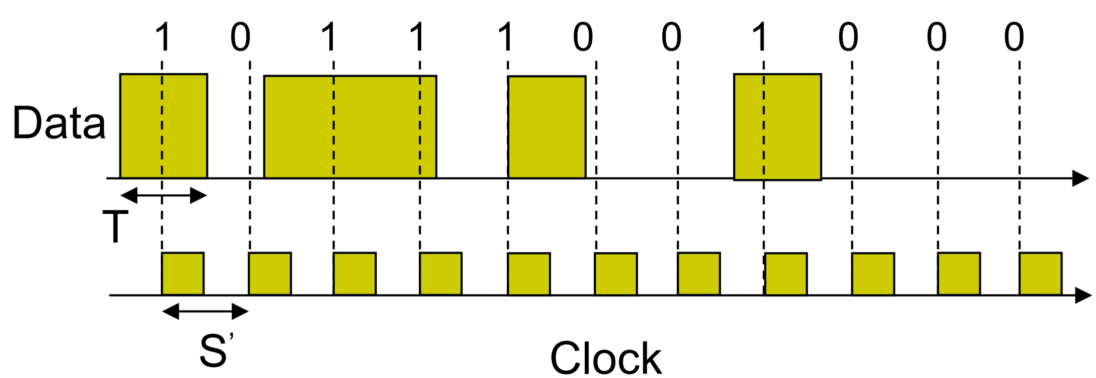
(b)



Synchronization

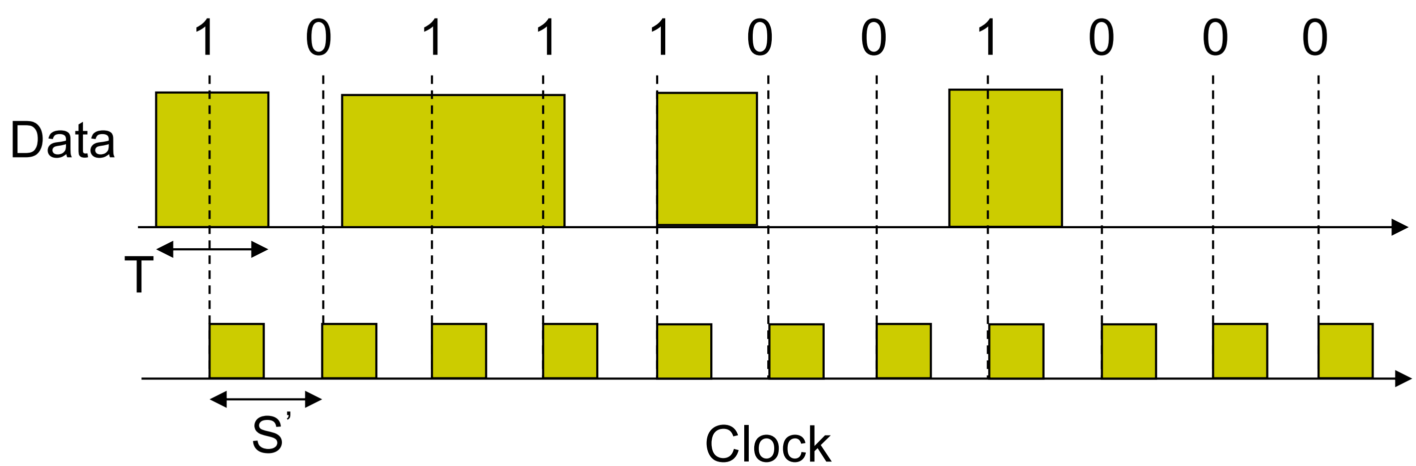
- Synchronization of clocks in transmitters and receivers.
  - clock drift causes a loss of synchronization
- Example: assume '1' and '0' are represented by V volts and 0 volts respectively
  - Correct reception
  - Incorrect reception due to incorrect clock (slower clock)





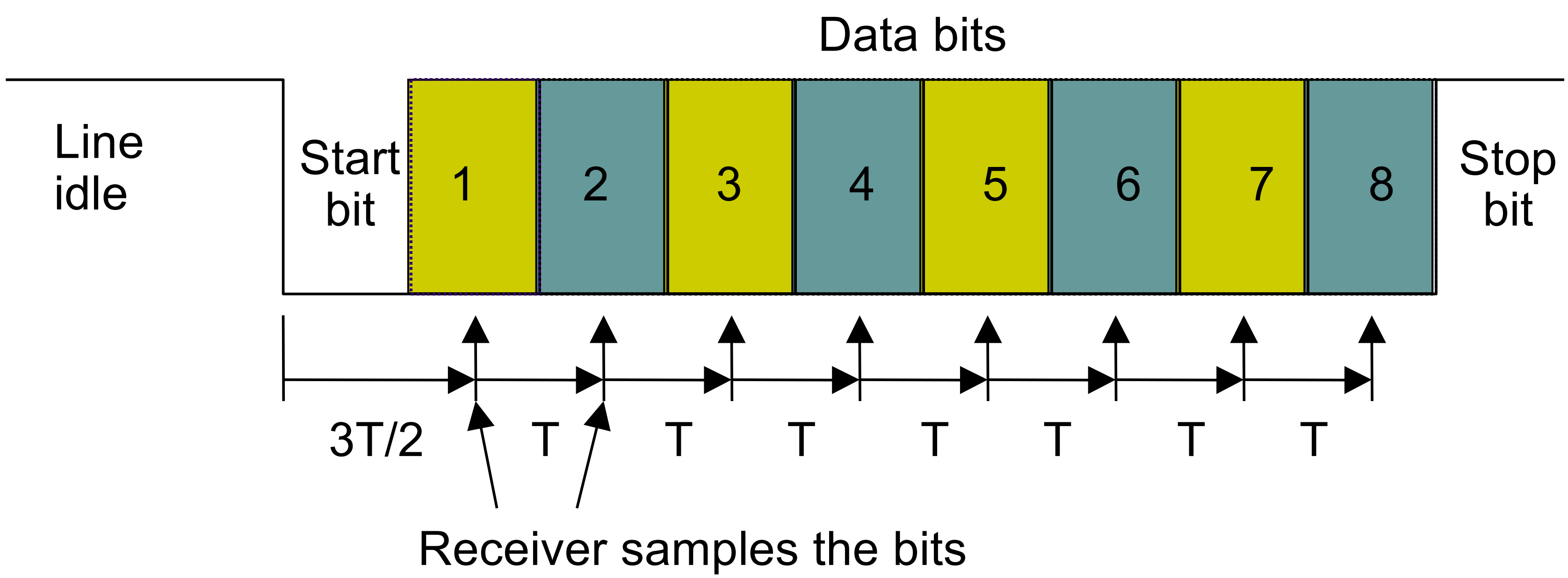
Synchronization (cont'd)

- Incorrect reception (faster clock)
- How to avoid a loss of synchronization?
  - Asynchronous transmission
  - Synchronous transmission



Asynchronous Transmission

- Avoids synchronization loss by specifying a short maximum length for the bit sequences and resetting the clock in the beginning of each bit sequence.
- Accuracy of the clock?



Synchronous Transmission

- Sequence contains data + clock information (line coding)
  - i.e. Manchester encoding, self-synchronizing codes, is used.
- R transition for *R* bits per second transmission
- R transition contains a sine wave with *R* Hz.
- *R* Hz sine wave is used to synch receiver clock to the transmitter's clock using PLL (phase-lock loop)