Computer Networks

Physical Layer

Based on slides from **Zoltán Ács ELTE** and D. Choffnes Northeastern U.,
Philippa Gill from StonyBrook University , Revised in 2018 by S. Laki

Last week in Computer Networks

The four goals of reliable transfer

correctness ensure data is delivered, in order, and untouched

timeliness minimize time until data is transferred

efficiency optimal use of bandwidth

fairness play well with concurrent communications

Here is one correct, timely, efficient and fair transport mechanism

ACKing **full information ACK**

retransmission **after timeout**

after k subsequent ACKs

window management **additive increase upon successful delivery**

multiple decrease when timeouts

More details later when we see TCP

Examples

Go-back-N and Selective Repeat

Go-Back-N (GBN)

a simple sliding window protocol using <u>cumulative ACKs</u>

goal **receiver should be as simple as possible**

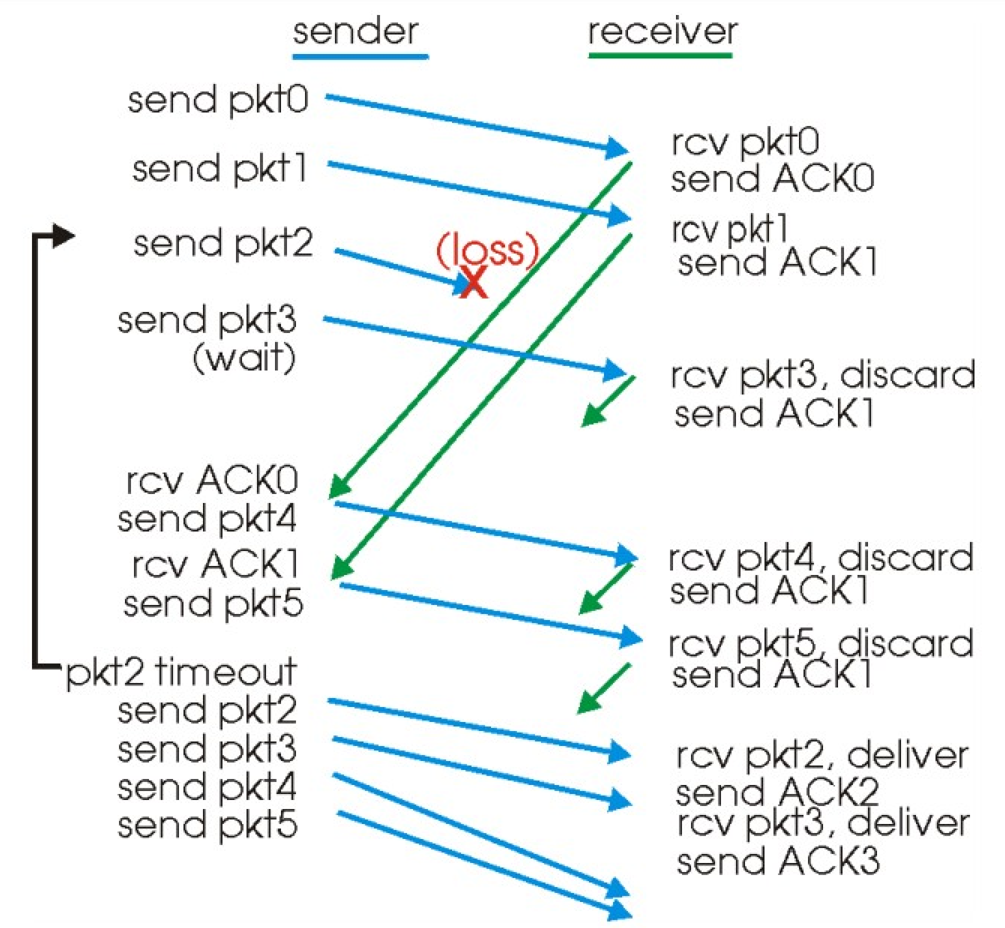
receiver **delivers packets in-order to the upper layer**

receiver wnd size is 1

sender **use a single timer to detect loss, reset at each new ACK**

upon timeout, resend all WND packets
starting with the lost one

GBN in action



Selective Repeat (SR)

avoid unnecessary retransmissions

by <u>using per-packet ACKs</u>

goal **avoids unnecessary retransmissions**

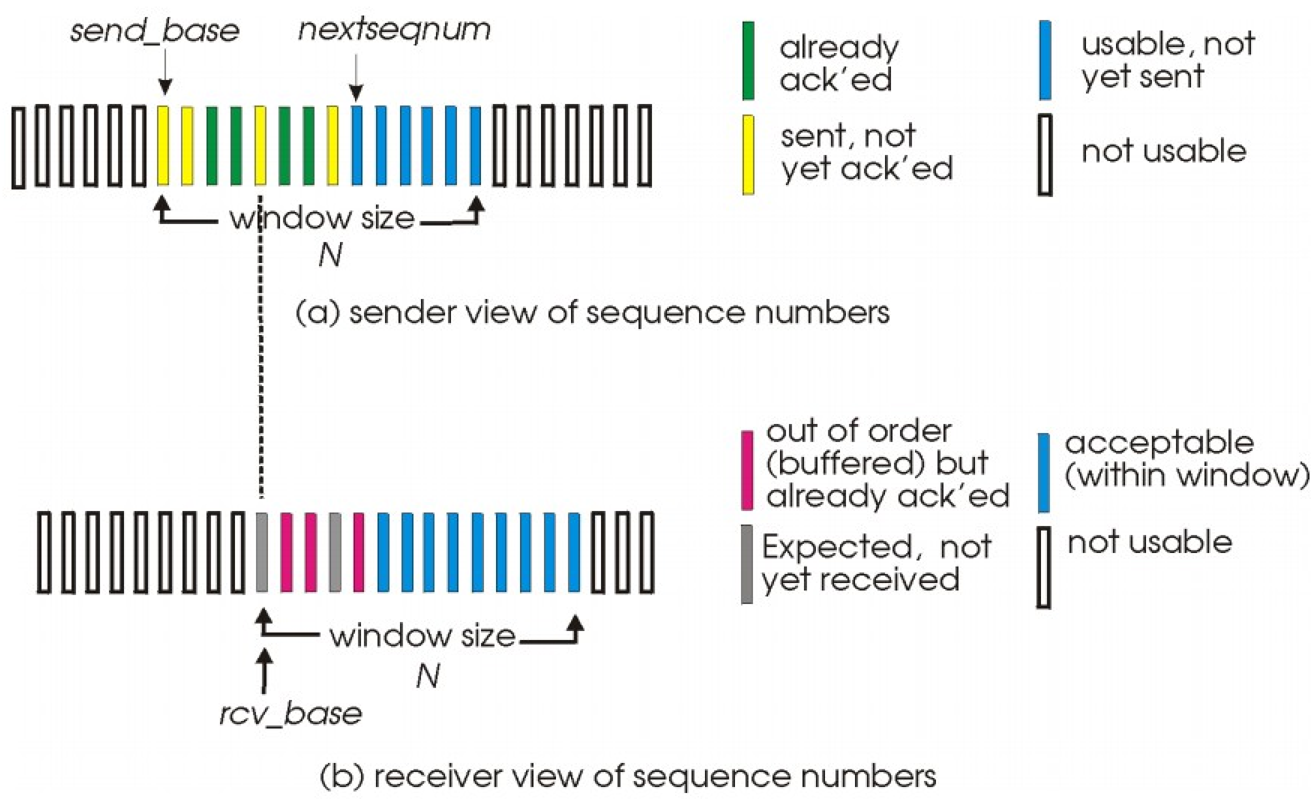
receiver **ACK each packet, in-order or not**

buffer out-of-order packets

sender **use per-packet timer to detect loss**

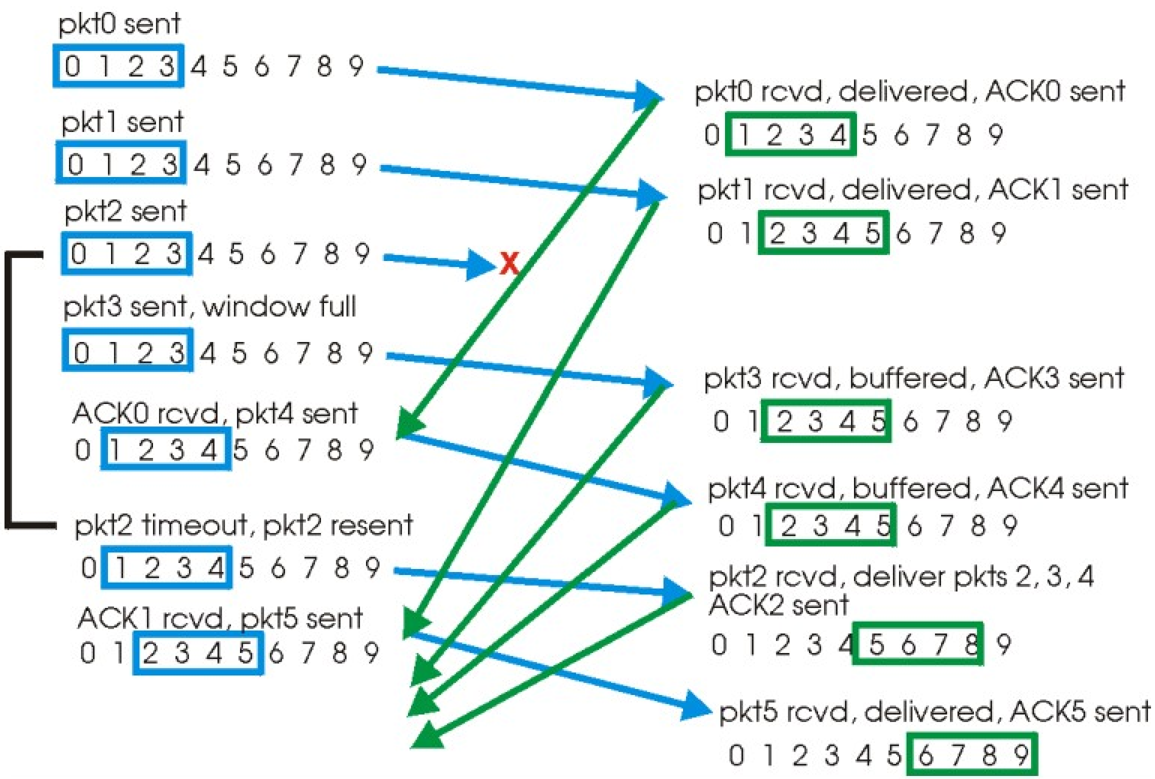
upon loss, only the lost packet

SR - windows

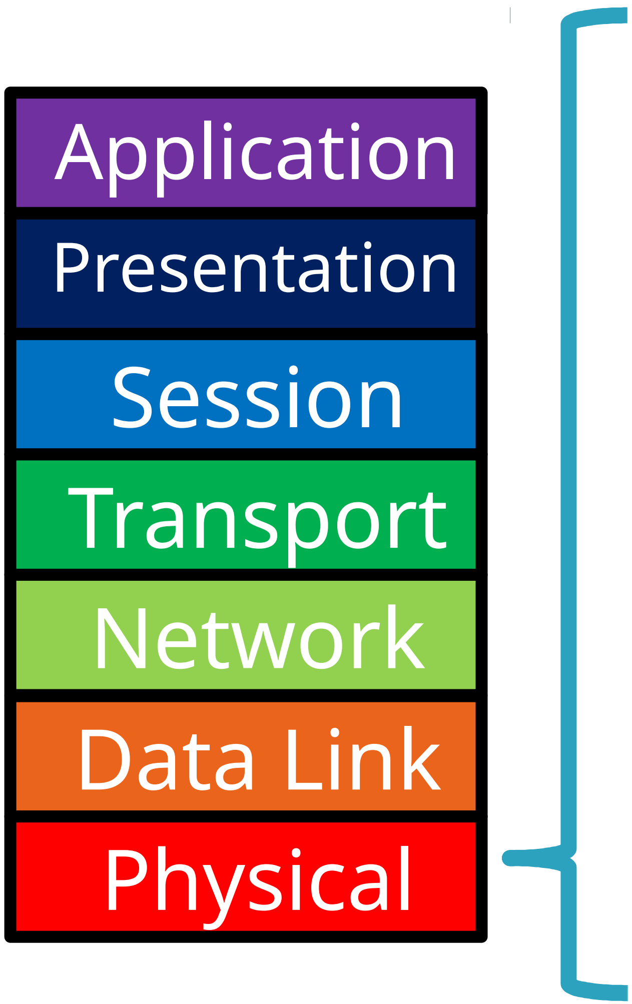


send_base
nextseqnum
already ack'ed
sent, not yet ack'ed
usable, not yet sent
not usable
window size
N
(a) sender view of sequence numbers
out of order (buffered) but already ack'ed
Expected, not yet received
acceptable (within window)
not usable
window size
N
rcv_base
(b) receiver view of sequence numbers

SR in action


pkt0 sent
0 1 2 3 4 5 6 7 8 9
pkt1 sent
0 1 2 3 4 5 6 7 8 9
pkt2 sent
0 1 2 3 4 5 6 7 8 9
X
pkt3 sent, window full
0 1 2 3 4 5 6 7 8 9
ACK0 rcvd, pkt4 sent
0 1 2 3 4 5 6 7 8 9
pkt2 timeout, pkt2 resent
0 1 2 3 4 5 6 7 8 9
ACK1 rcvd, pkt5 sent
0 1 2 3 4 5 6 7 8 9
pkt0 rcvd, delivered, ACK0 sent
0 1 2 3 4 5 6 7 8 9
pkt1 rcvd, delivered, ACK1 sent
0 1 2 3 4 5 6 7 8 9
pkt3 rcvd, buffered, ACK3 sent
0 1 2 3 4 5 6 7 8 9
pkt4 rcvd, buffered, ACK4 sent
0 1 2 3 4 5 6 7 8 9
pkt2 rcvd, deliver pkts 2, 3, 4
ACK2 sent
0 1 2 3 4 5 6 7 8 9
pkt5 rcvd, delivered, ACK5 sent
0 1 2 3 4 5 6 7 8 9

Physical Layer



- Function:
  - Get bits across a physical medium
- Key challenge:
  - How to represent bits in analog
  - Ideally, want high-bit rate
  - But, must avoid desynchronization

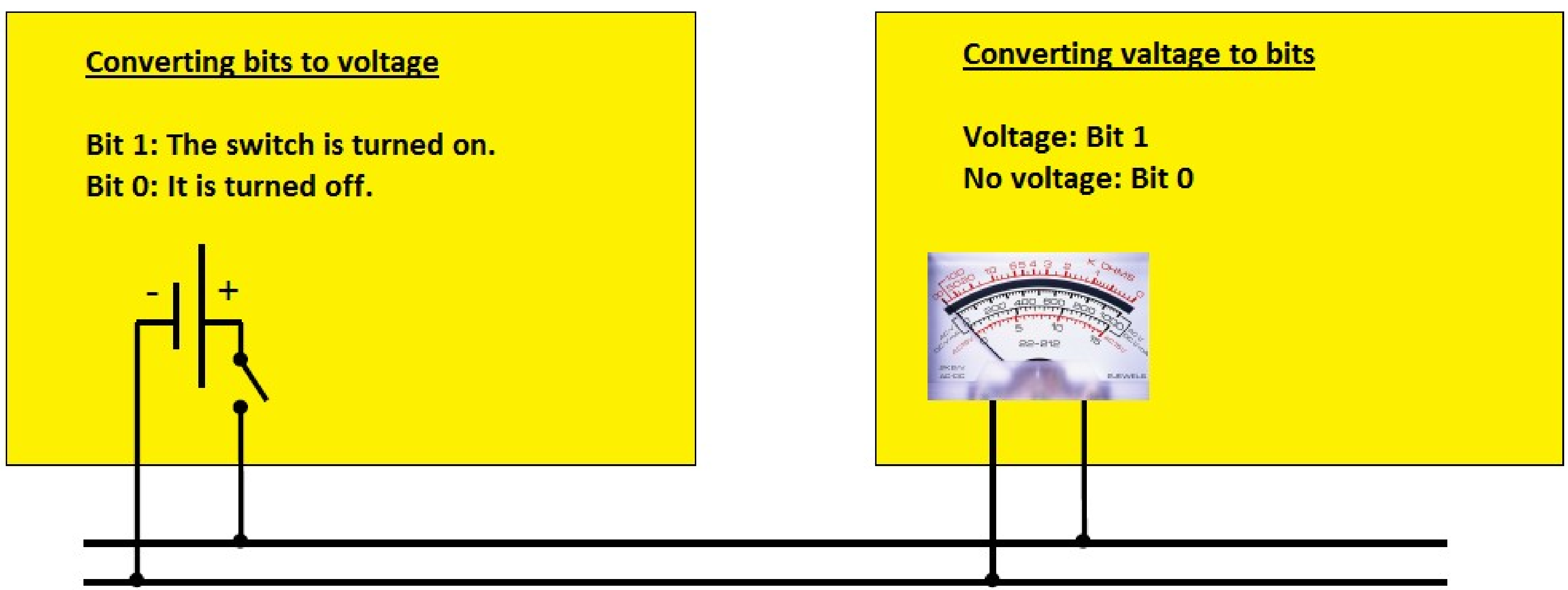
Key challenge

- Digital computers
  - 0s and 1s
- Analog world
  - Amplitudes and frequencies



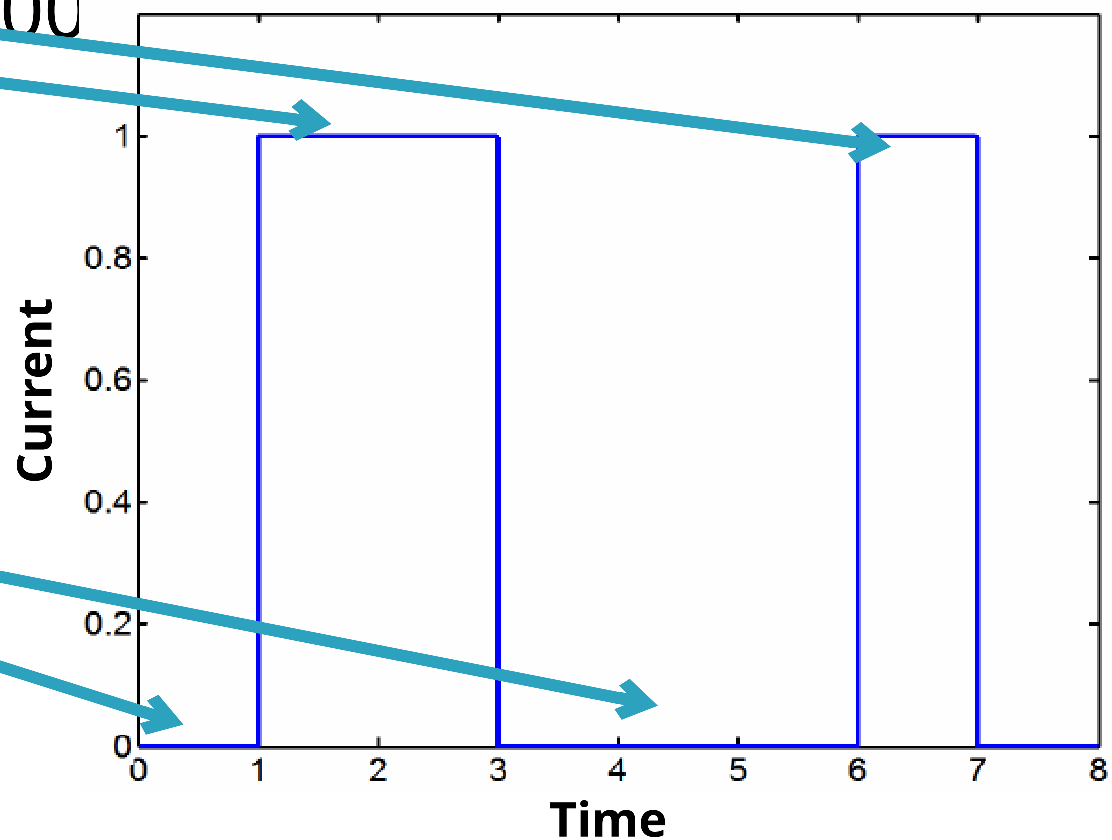

Simple transmission - baseband

- Bit 1: voltage or current strength
- Bit 0: no voltage



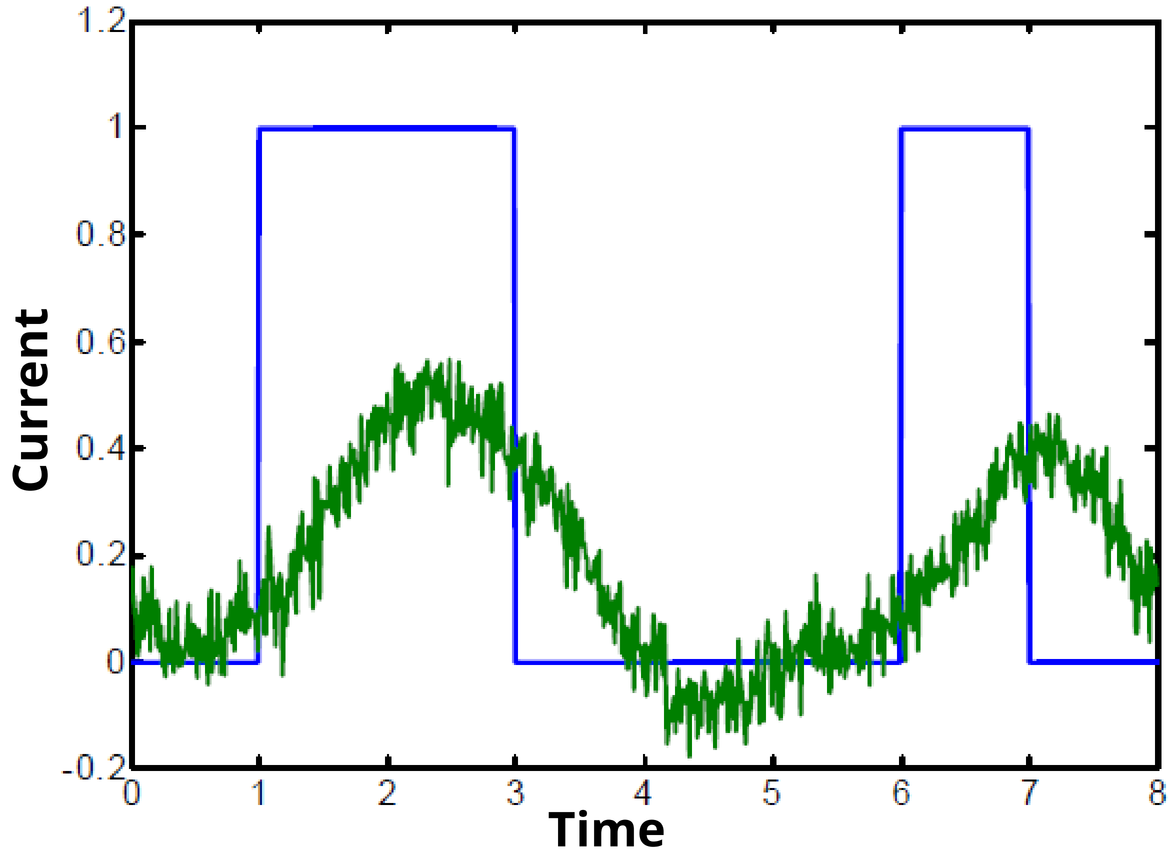
Transmission of „b”

- More than one bit is needed for tranmitting char „b”
- „b” in ASCII: 01100



Transmission of „b" in a real world

- Poor reception – a typical pattern at the receiver



Fundamentals – Singals

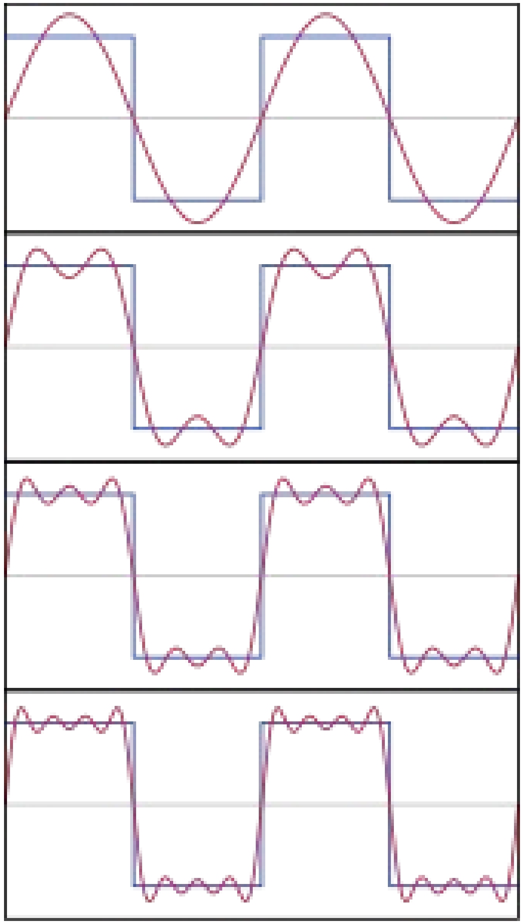
- To understand signal propagation on a physical medium, some background is required how such signals can be analyzed/treated mathematically

First: Fourier's theorem

Any periodic function *g(t)* (with period T) can be written as a (possibly infinite) sum of sine and cosine functions; the frequencies of these functions are integer multiples of the base frequency f = 1/T.

$$g(t) = \frac{1}{2}c + \sum_{n=1}^{\infty} a_n \sin(2\pi n f t) + \sum_{n=1}^{\infty} b_n \cos(2\pi n f t),$$

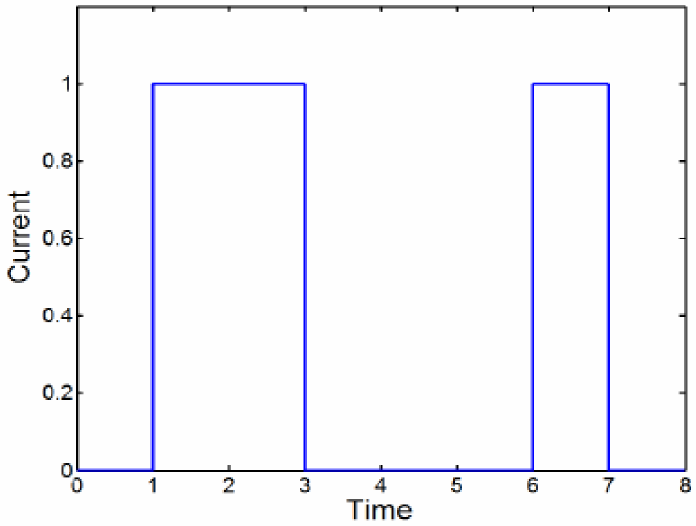
where $f = \frac{1}{T}$ is the base frequency, a_n and b_n are constants, representing the amplitudes of nth sine and cosine harmonics. C is a constant.



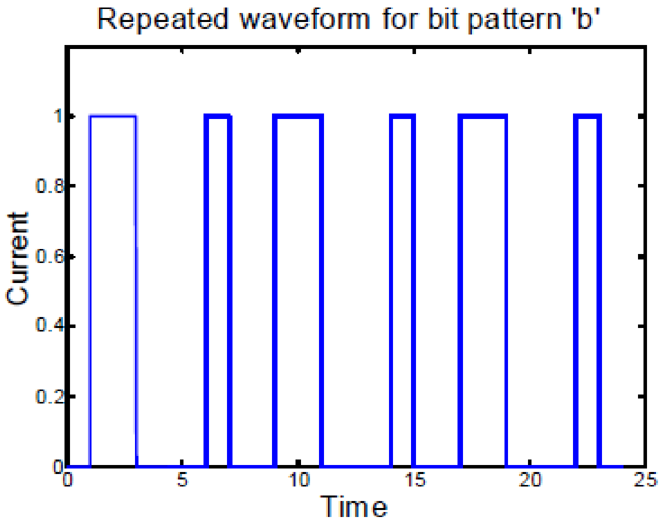
Fundamental – Terms of the Fourier series

Application

- A digital signal is not periodic
  - E.g. the ASCII code of „b" is 8 bits long



- Use a trick: Suppose waveform is repeated infinitely often,
- For „b", resulting in a periodic waveform with period 8 bit times

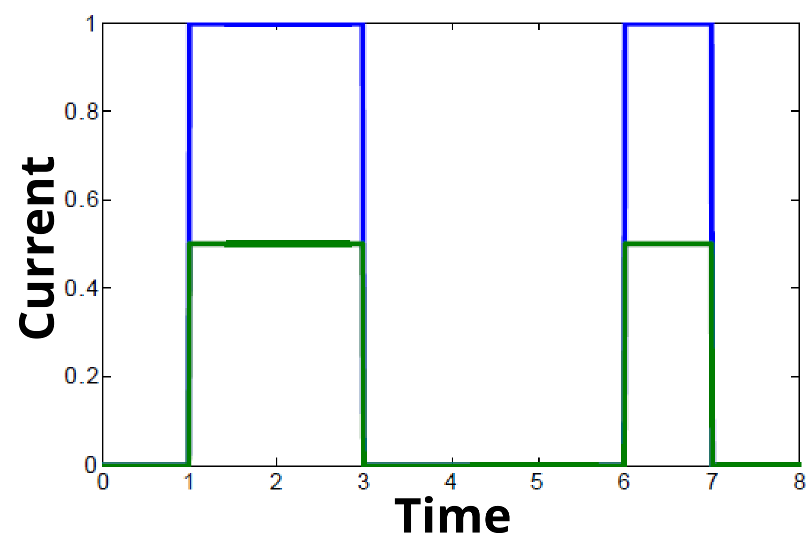


Fundamentals - Attenuation

- Attenuation: α
  - Ratio of transmitted (P_0) and received (P_1) power
  - High attenuation = little power arrives at receiver
    - Making the understanding of signal difficult
  - Typically given in deciBel

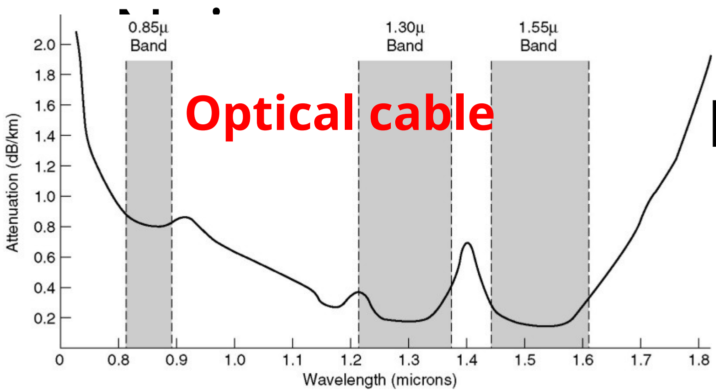
$$\alpha[in\ dB] = 10 \times \log_{10}\frac{P_0}{P_1} \text{ (deciBel [dB])}$$

- It depends on
  - Physical medium
  - Distance between sender and reciever
  - … others

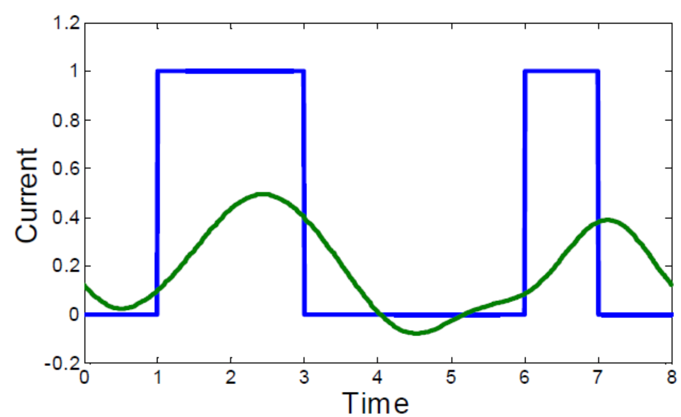


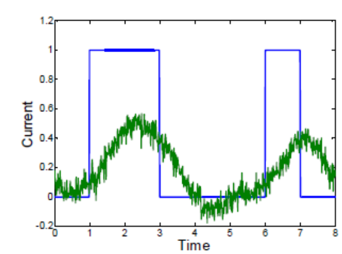
Fundamentals - Attenuation

- In reality
  - Attenuation is not uniform, depends on frequency
  - Not all frequencies pass through a medium
  - Phase shifting
    - Different frequencies have different signal propagation speed
    - Frequency-based disortion



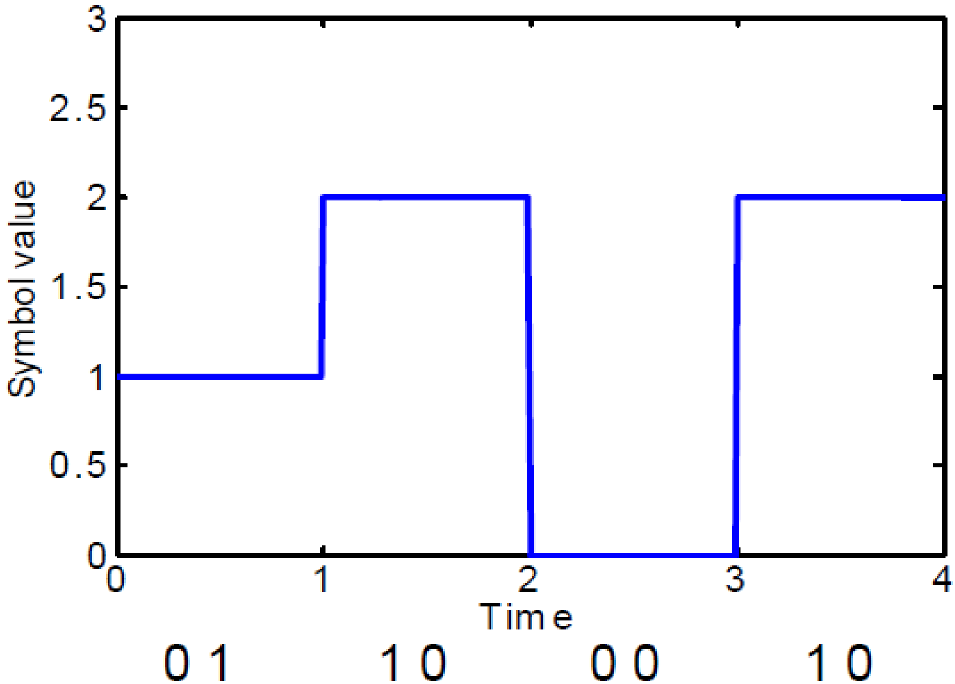
k





Symbols and bits

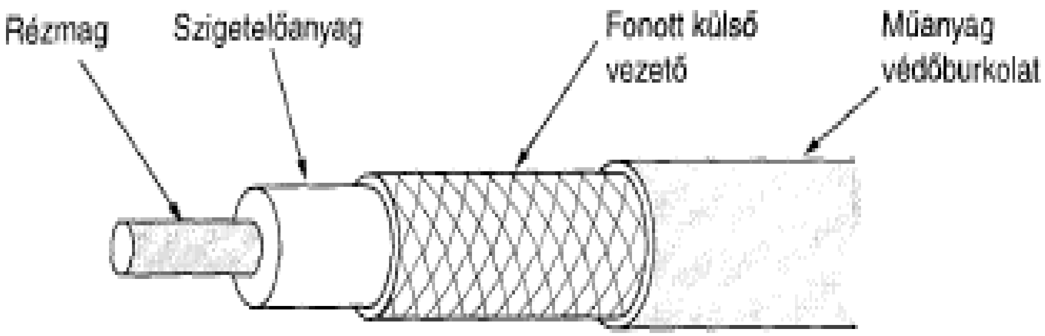
- Use more symbols than 0 and 1 in the channel
- Example:
  - Having 4 symbols: A(00),B(01),C(10),D(11)
  - Symbol rate: (BAUD)
    - Transmitted symbols per sec
  - Data rate (bps):
    - Transmitted bits per sec



Example:
A 600 Baud modem with 16 symbols, one can reach data rate of 2400 bps.

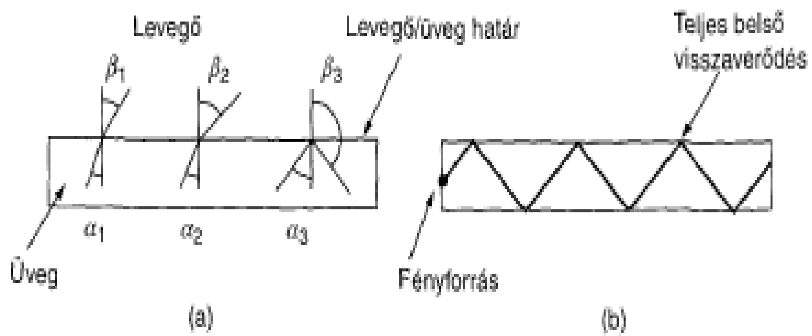
Physical media – wired 1/1

- **Magnetic storage** – e.g. never underestimate the power of a truck of hard disks
- **Twisted pair** – telephone networks; double copper wire, both analog and digital; UTP and STP
- **Coaxial cable** – Higher speed and larger distance than with twisted pair; analog (*75 Ω*) and digital (*50 Ω*)



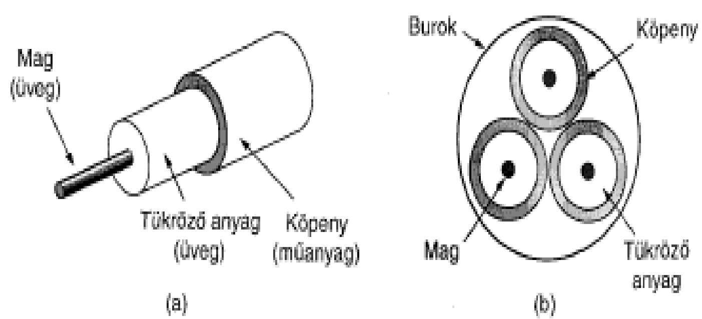
Physical media – wired 2/2

- **Optical cable** – parts: light source, media and detector; light impulse = 1 bit, no light impulse = 0 bit;



(Tanenbaum)

- **Optical cables**:



Fundamentals – wireless transmission

- **Frequency:** the rate per second of a vibration constituting an electromagnetic wave.
  - Notation: f
  - Measured in: Hertz (Hz)

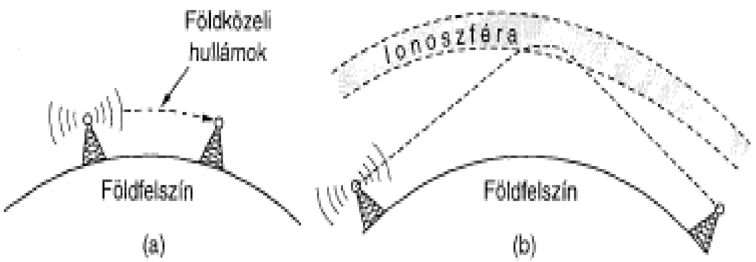
- **Wavelength:** the distance between successive crests of a wave
  - Notation: λ

- **Speed of light:** signal propagation speed of electric signals in a physical media
  - Notation c
  - In vacuum: kb. $3 * 10^8 \frac{m}{s}$
  - In copper or optical cable: 2/3 x c(vacuum)

- **Relationship: $\lambda f = c$**

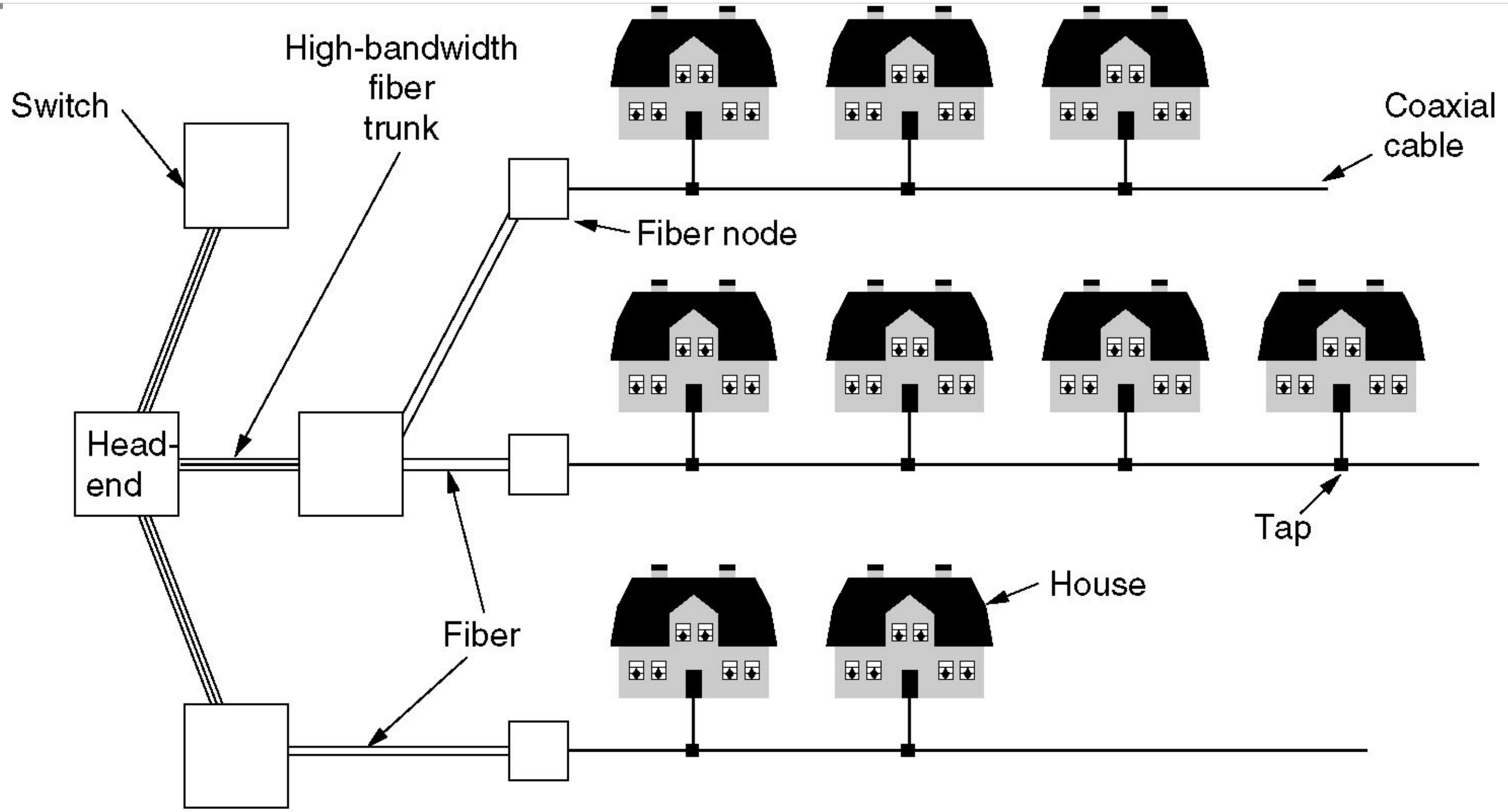
Funamentals – wireless

- **Radio frequency transmission** – simple; large distances; indoor and outdoor; frequency-dependent propagation properties



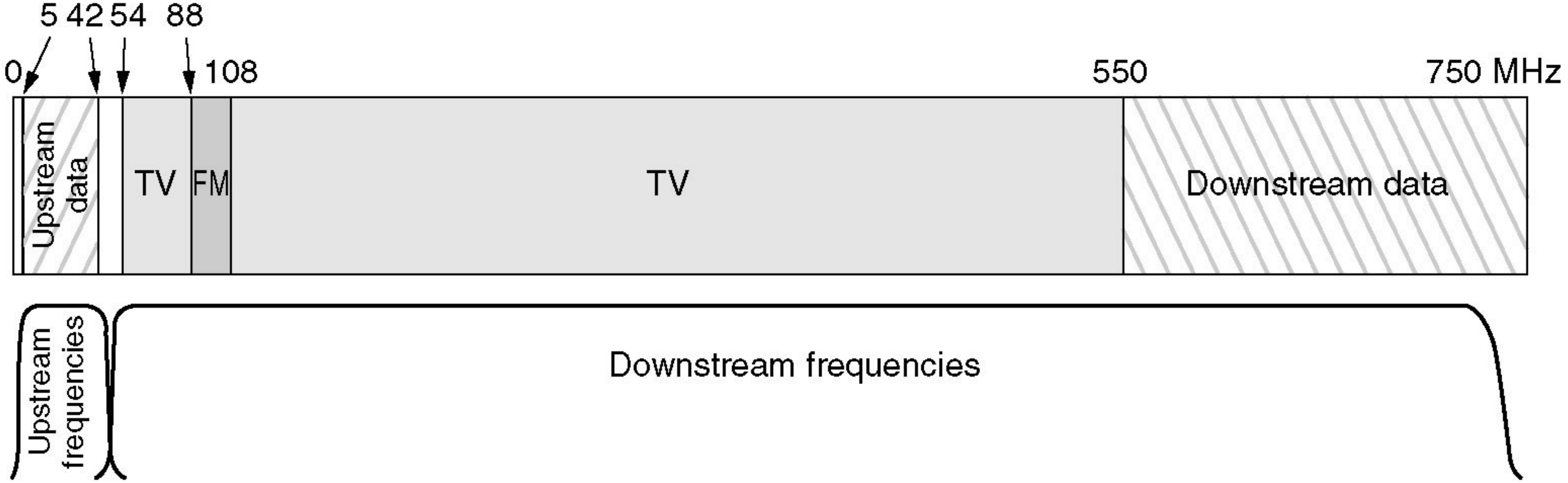
- **Microwave transmission** – propagation along a straight line; attenuation; cheap
- **Infrared and millimeter-wave** – small distances; cannot go through objects
- **Visible light** – laser; high speed, cheap; weather conditions;

Internet in a cable TV network



(a)

Internet in a cable TV network

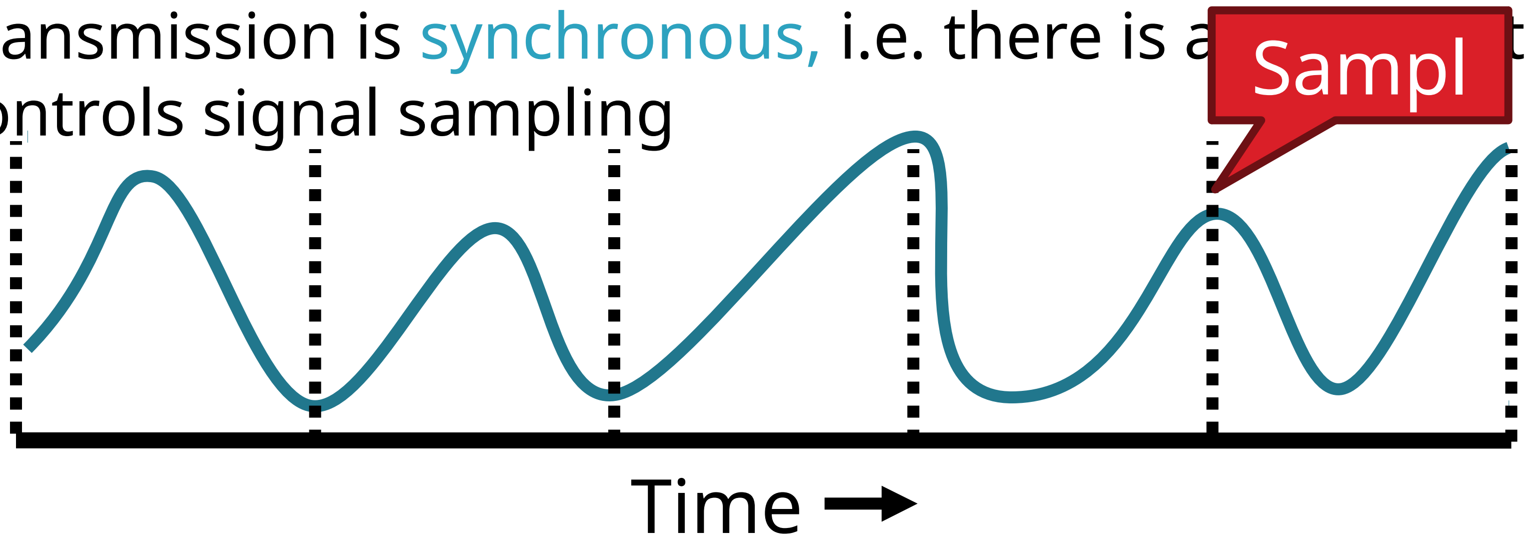


Already discussed...

Data transmission

Assumptions

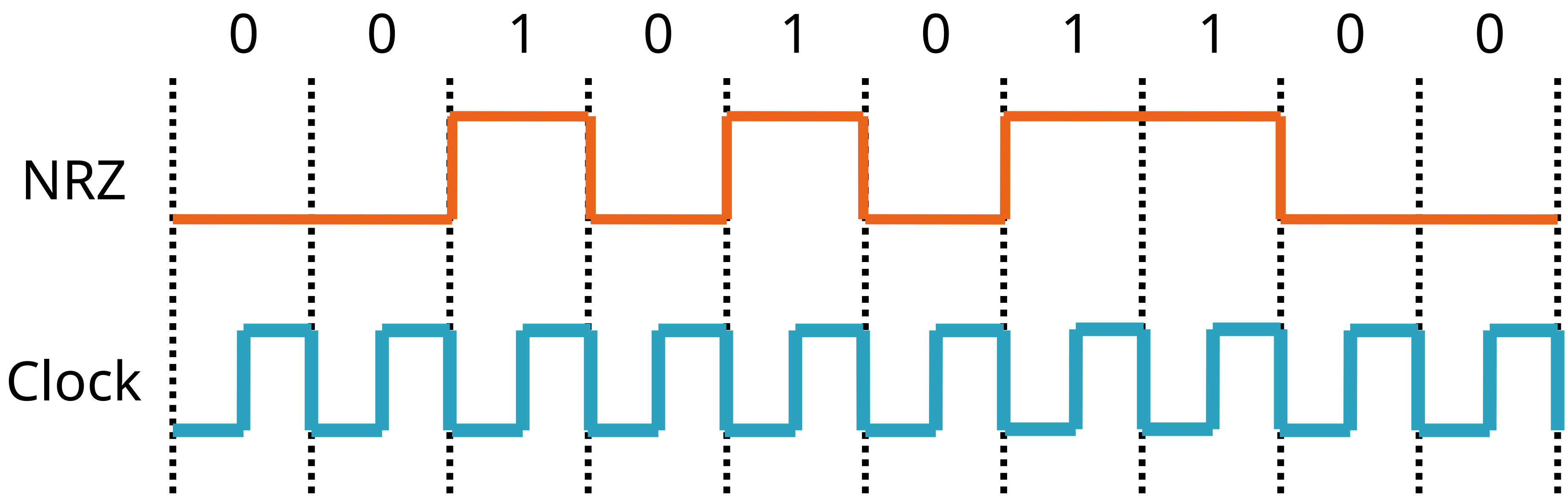
- We have two discrete signals, high and low, to encode 1 and 0
- Transmission is synchronous, i.e. there is a t controls signal sampling



- Amplitude and duration of signal must be significant

Non-Return to Zero (NRZ)

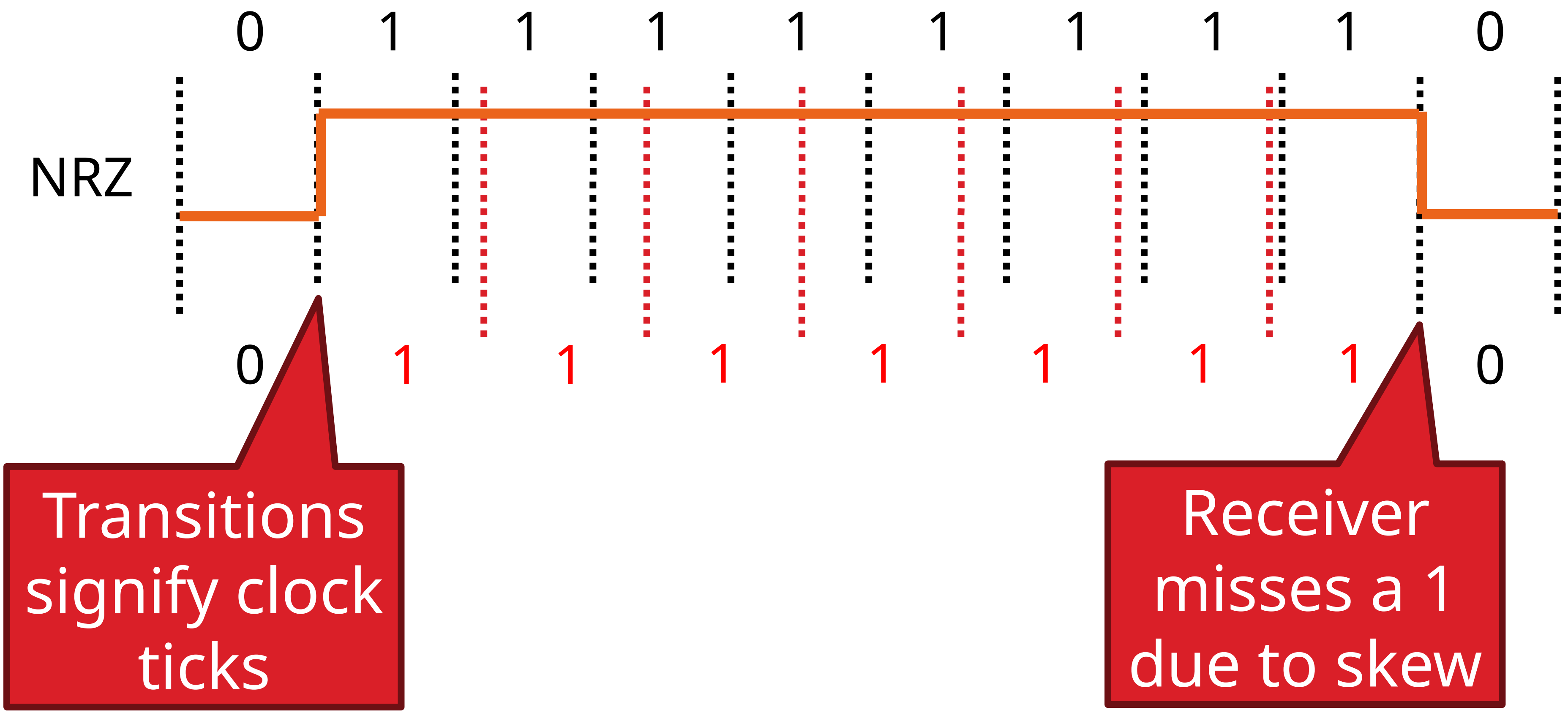
- 1 → high signal, 0 → low signal



- Problem: long strings of 0 or 1 cause desynchronization
  - How to distinguish lots of 0s from no signal?
  - How to recover the clock during lots of 1s?

Desynchronization

- Problem: how to recover the clock during sequences of 0's or 1's?



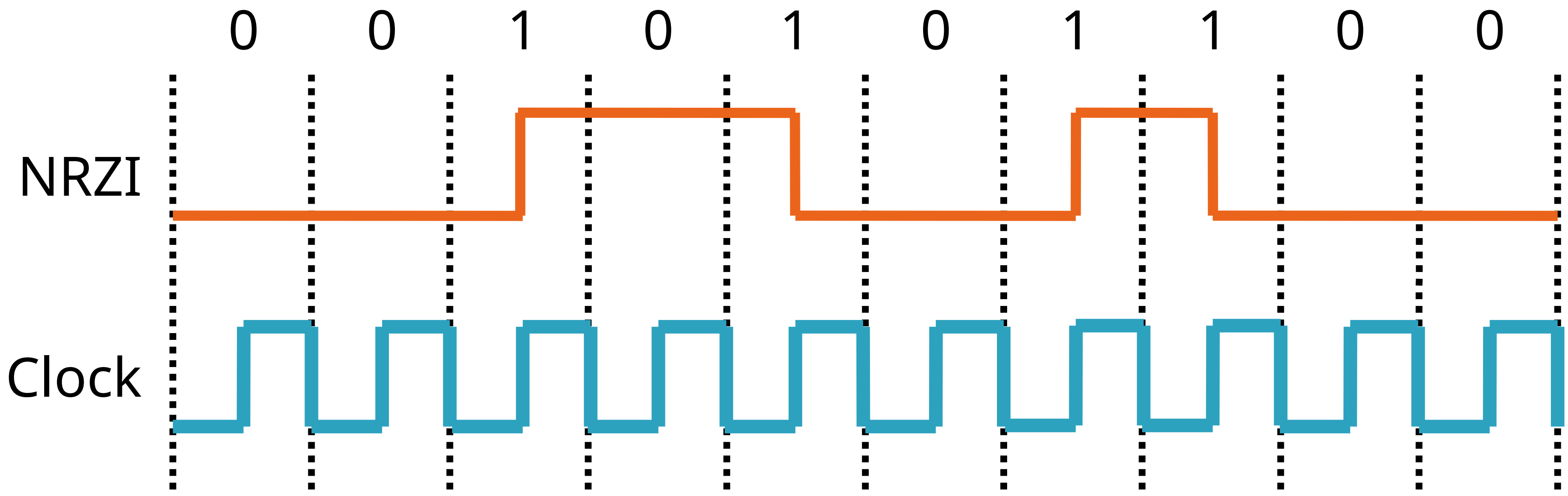
- Clock drift is major problem – two different clocks never stay in perfect synchrony

Options to tell the receiver when to sample

- Relying on permanently synchronized clocks does not work
  1. Explicit clock signal
     - Needs parallel transmission over some additional channel
     - Must be in synch with the actual data, otherwise pointless !
     - Useful only for short-range communication
  2. Synchronize the receiver at crucial points (e.g., start of a character or of a block)
     - Otherwise, let the receiver clock run freely
     - Relies on short-term stability of clock generators (do not diverge too quickly)
  3. Extract clock information from the received signal itself
     - *Self-clocked signals*
     - Put enough information into the data signal itself so that the receiver can know immediately when a bit starts/stop

Non-Return to Zero Inverted (NRZI)

- 1 make transition, 0 remain the same



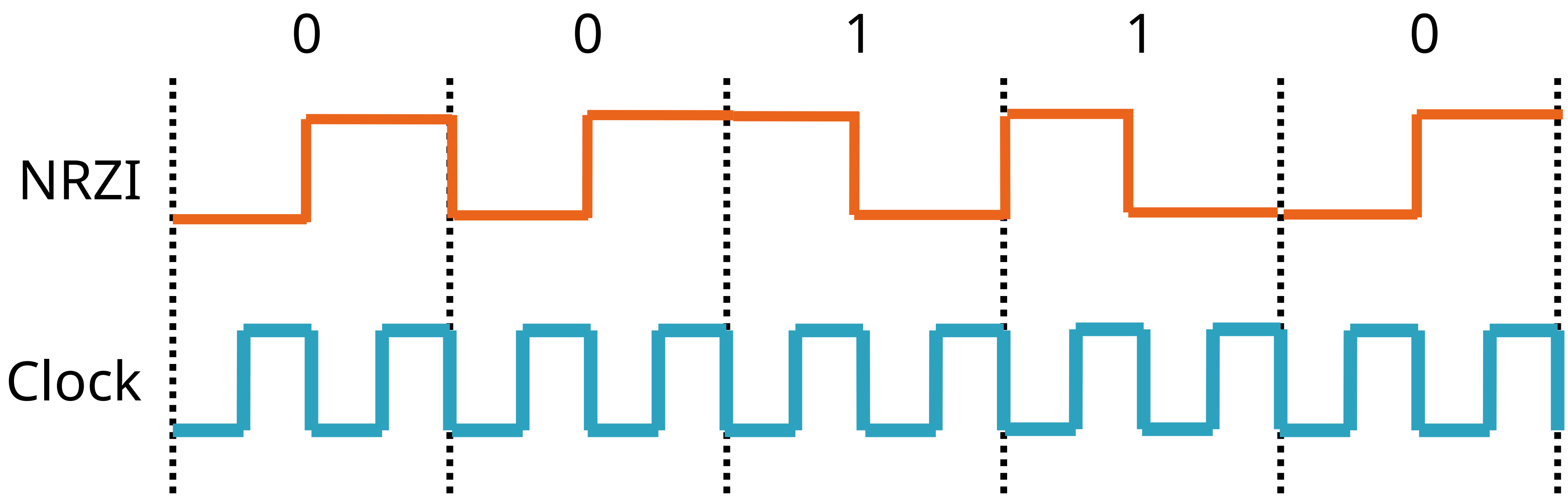
- Solves the problem for sequences of 1s, but not 0s

Ethernet examples:

- 10BASE-TX
- 100BASE-TX

Manchester – used by 10BASE-TX

- 1 → high-to-low, 0 → low-to-high



- Good: Solves clock skew (every bit is a transition)
- Bad: Halves throughput (two clock cycles per bit)

4-bit/5-bit (100 Mbps Ethernet)

- Observat n: NRZI works as long as no sequences of 0
- Idea: all 4-bit sequences as 5-bit sequences

8-bit / 10-bit used in Gigabit

two trailing 0

4-bit	5-bit	4-bit	5-bit
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

- Tradeoff: efficiency drops to 80%

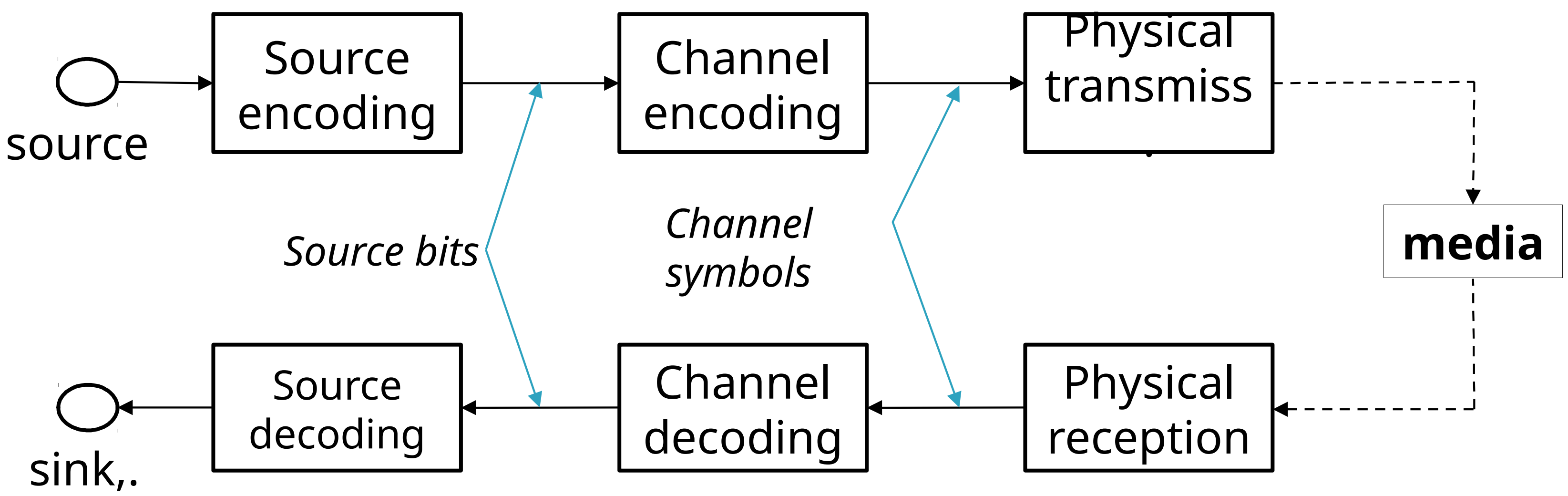
Signal transmission

Baseband VS broadband transmission

- *baseband*
  - Baseband transmission directly puts the digital symbol sequences onto the wire
  - At different levels of current, voltage, … essentially, direct current (DC) is used for signaling
  - Baseband transmission suffers from the problems discussed above
    - Limited bandwidth reshapes the signal at receiver
    - Attenuation and distortion depend on frequency and baseband transmissions have many different frequencies because of their wide Fourier spectrum

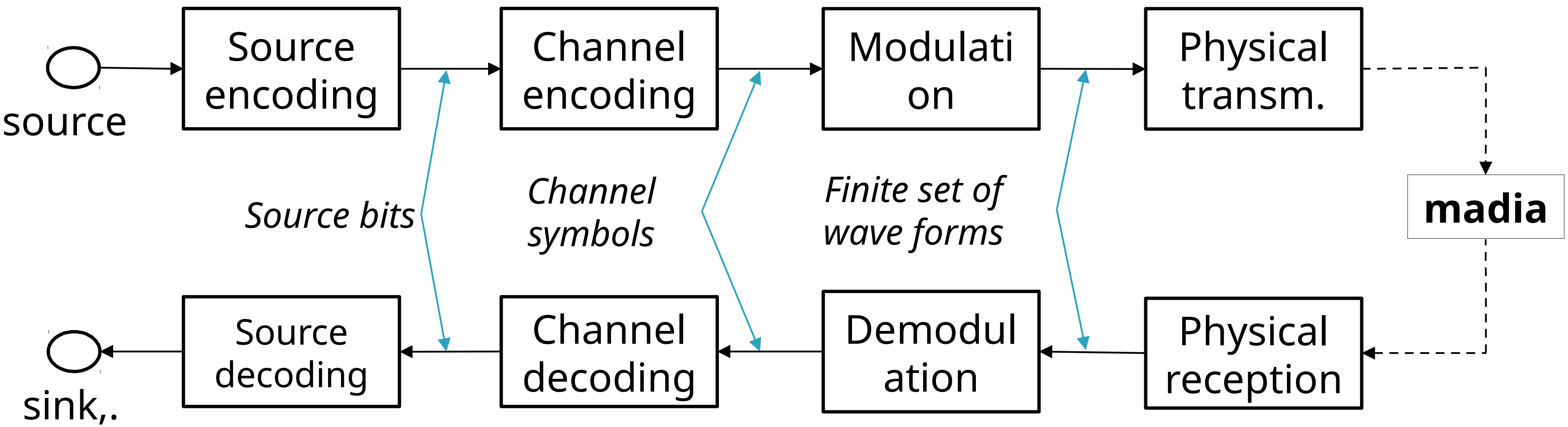
- *broadband*
  - Idea: get rid of the wide spectrum needed for DC transmission
  - Use a sine wave as a carrier for the symbols to be transmitted
    - Typically, the sine wave has high frequency
    - But only a single frequency!
    - Pure sine waves has no information, so its shape has to be influenced according to the symbols to be transmitted
  - The carrier has to be modulated by the symbols (widening the spectrum)
    - Three parameters that can be influenced Amplitude, Frequency, Phase

Digital baseband transmission



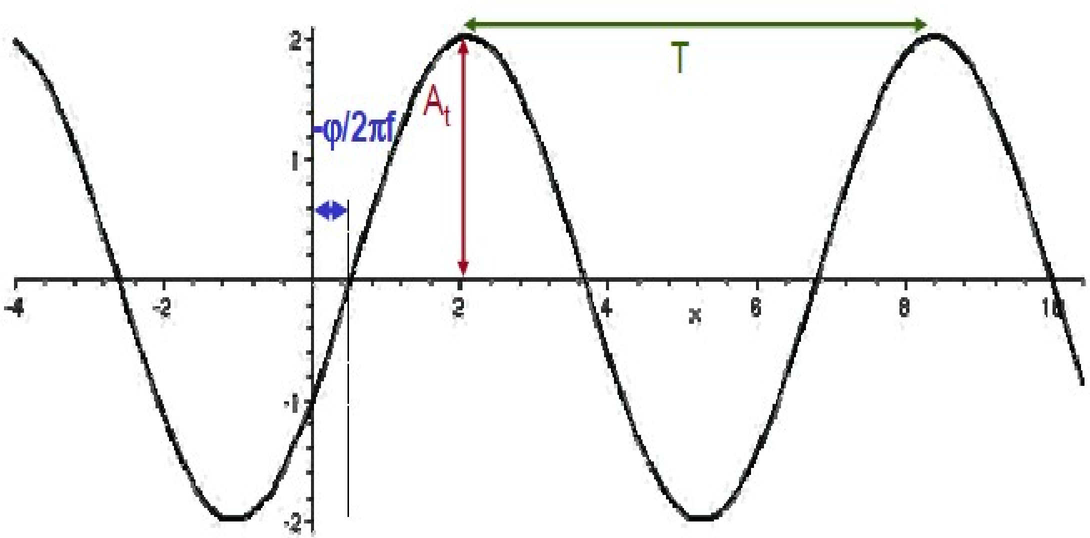
- Bring source information in digital form
  - E.g., sample and quantize an analog voice signal, represent text as ASCII
- Source encode: Remove redundant or irrelevant data
  - E.g., lossy compression (MP3, MPEG 4); lossless compression (Huffmann coding, runlength coding)
- Channel encode: Map source bits to channel symbols
  - Potentially several bits per symbol
  - May add redundancy bits to protect against errors
  - Tailored to channel characteristics
- Physical transmit: Turn the channel symbols into physical signals
- At receiver: Reverse all these steps

Digital broadband transmission



Three key properties used to carry information

- $s(t) = A\sin(2\pi f t + \varphi)$, where A is the amplitude, f the frequency and φ the phase.

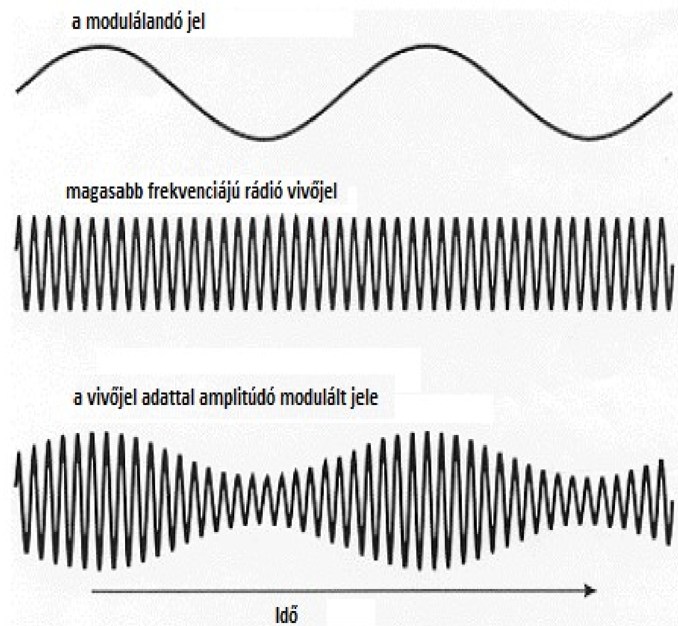


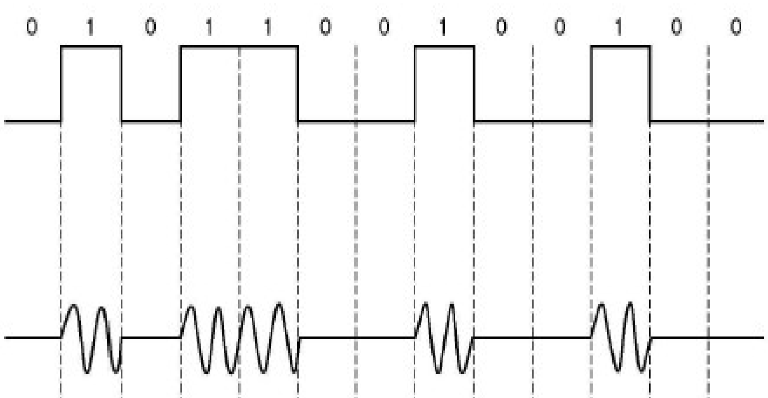
Amplitude modulation

- The time-varying s(*t*) signal is encoded into the amlitude of the sine wave (carrier):

$$f_A(t) = s(t) * \sin(2\pi f t + \varphi)$$

  - *Analog signal*: amplitude modulation
  - *Digital signal*: amplitude keying or on/off keying
  - *Digital signal*: amplitude keying or on/off keying (s(t) takes discrete values)

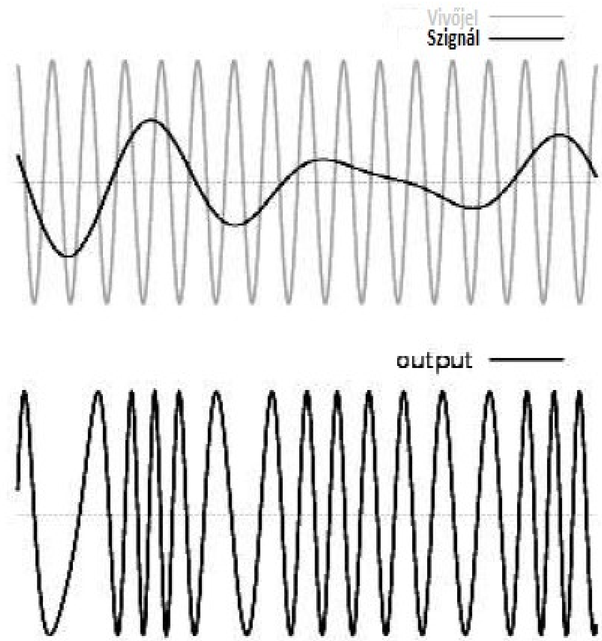




Frequency modulation

- The time-varying s(t) signal is encoded into the frequency of the sine wave:
  $f_F(t) = a * \sin(2\pi s(t)t + \varphi)$
  - *analog signal*: frequency modulation
  - *Digital signal*: frequency-shift keying

- The time-varying s(t) signal is encoded into the frequency of the sine wave:
  - *analog signal*: frequency modulation
  - *Digital signal*: frequency-shift



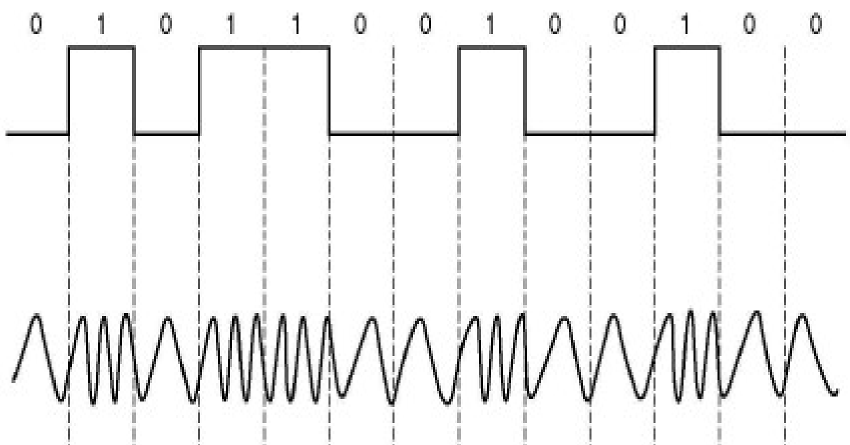
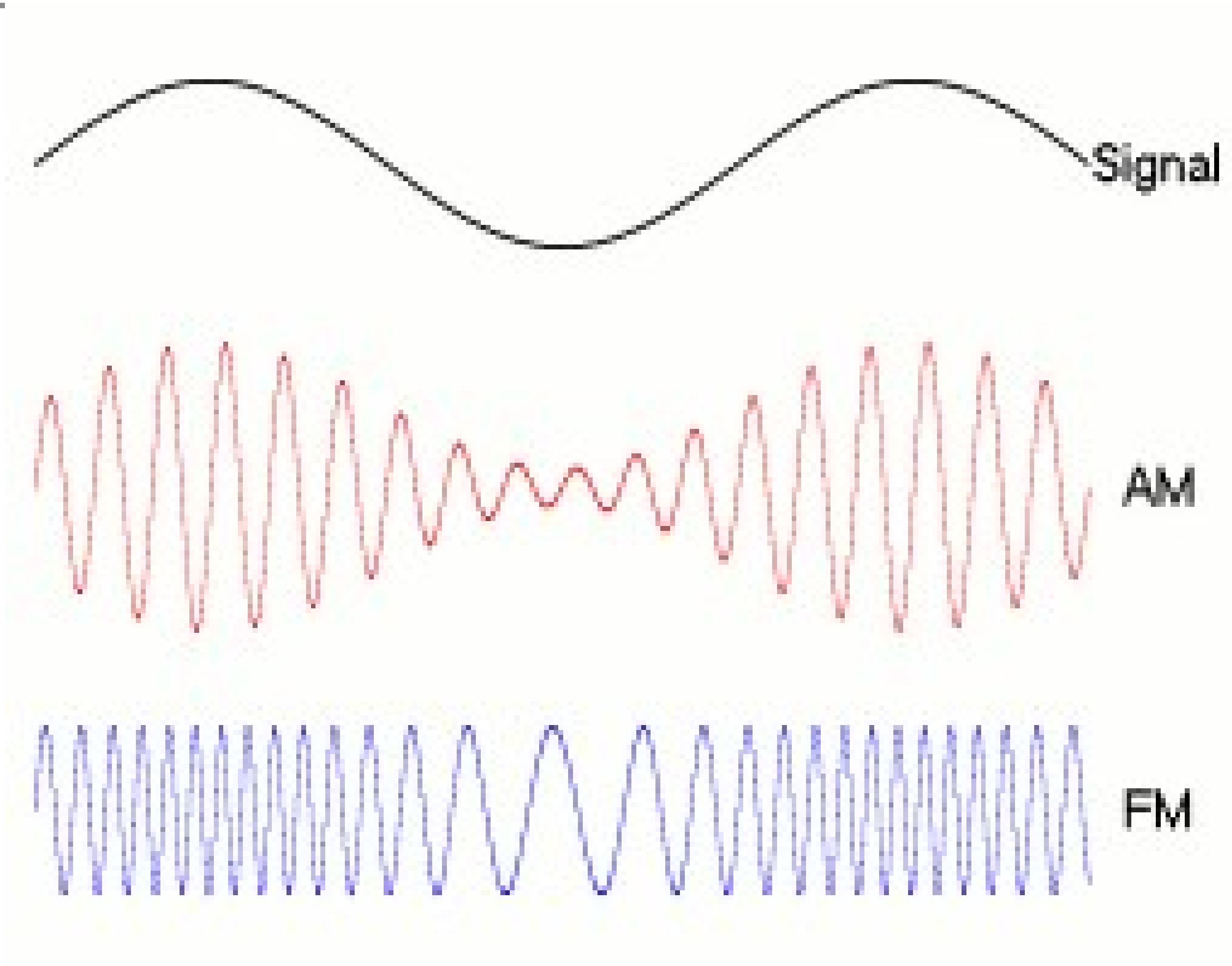


Illustration - AM & FM for analog signals

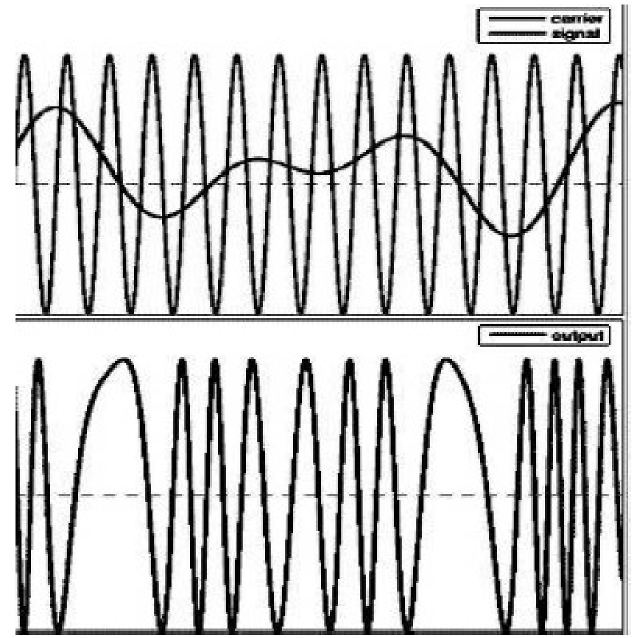


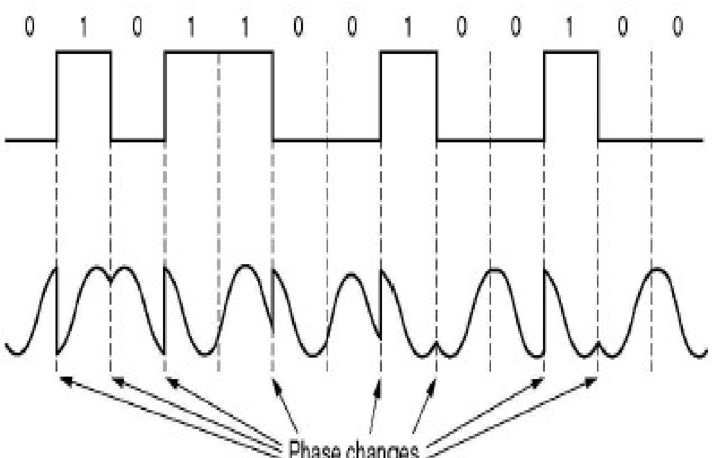
Phase modulation

- The signal $s(t)$ is encoded in the phase of the sine wave:

$$f_P(t) = a * \sin(2\pi f t + s(t))$$

  - *Analog signal*: phase modulation (not really used)
  - *Digital signal*: phase-shift keying (discrete set of phase changes)

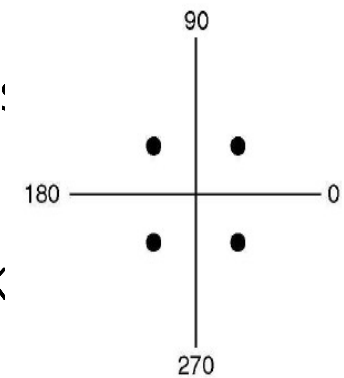




Usage of multiple symbols

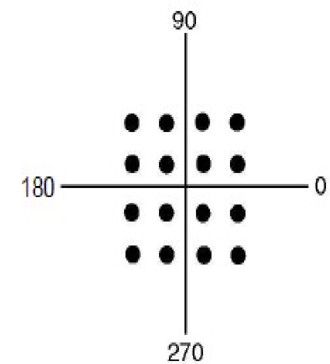
PSK with multiple values

- A receiver can usually quite well distinguish phase shifts
  - 4 symbols/values: $\frac{\pi}{4}, \frac{3\pi}{4}, \frac{5\pi}{4}, \frac{7\pi}{4}$
  - Result: Data rate is twice the symbol rate
  - Technique is called Quadrature Phase Shift Keying (QPSK)



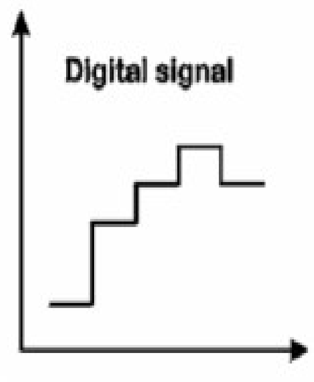
Amplitude + Phase modulation

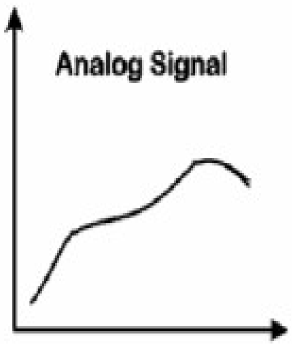
- Methods can be combined
- Symbols are encoded by a discrete set of amlitude, phase values
  - E.g. 16 symbols
  - Four times higher data rate than the symbol rate
  - Called as **Q**uadrature **A**mplitude **M**odulation-16



Digital VS analog signals

- A sender has two principal options what types of signals to generate
  - It can choose from a finite set of different signals – digital transmission
  - There is an infinite set of possible signals – analog transmission
- Simplest example: Signal corresponds to current/voltage level on the wire
  - In the digital case, there are finitely many voltage levels to choose from
  - In the analog case, any voltage is legal
- More complicated example: finite/infinitely many sinus functions
  - In both cases, the resulting wave forms in the medium can well be continuous functions of time!
- Advantage of digital signals: There is a principal chance that the receiver can precisely reconstruct the transmitted signal





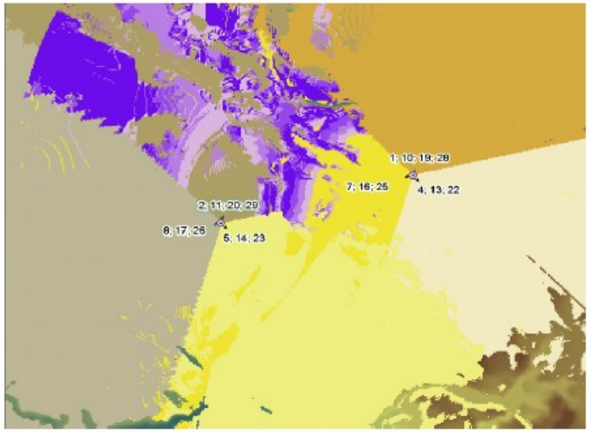
Static Channel Allocation

Multiplexing

- Enabling multiple signals to travel through the same media at the same time
- To this end, the channel is split into multiple smaller subchannels
- A special device (multiplexer) is needed at the sender, transmitting signals to the proper subchannel

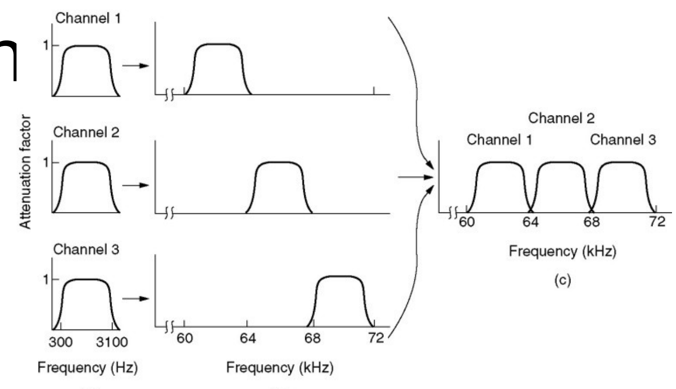
Space-Division Multiplexing

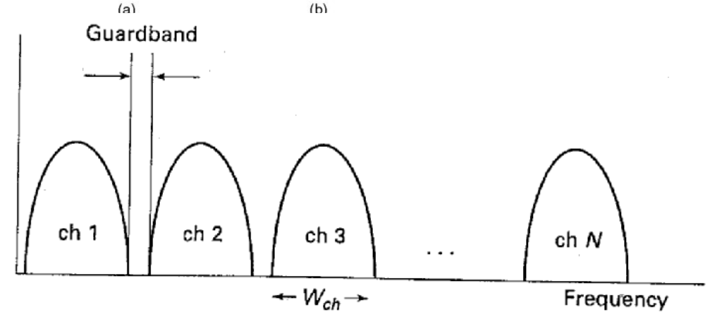
- Simplest way of multiplexing
- Wired example: point-to-point wire for each subchannel
- Wireless example: Different antennas for the subchannels



Frequency-Division Multiplexing

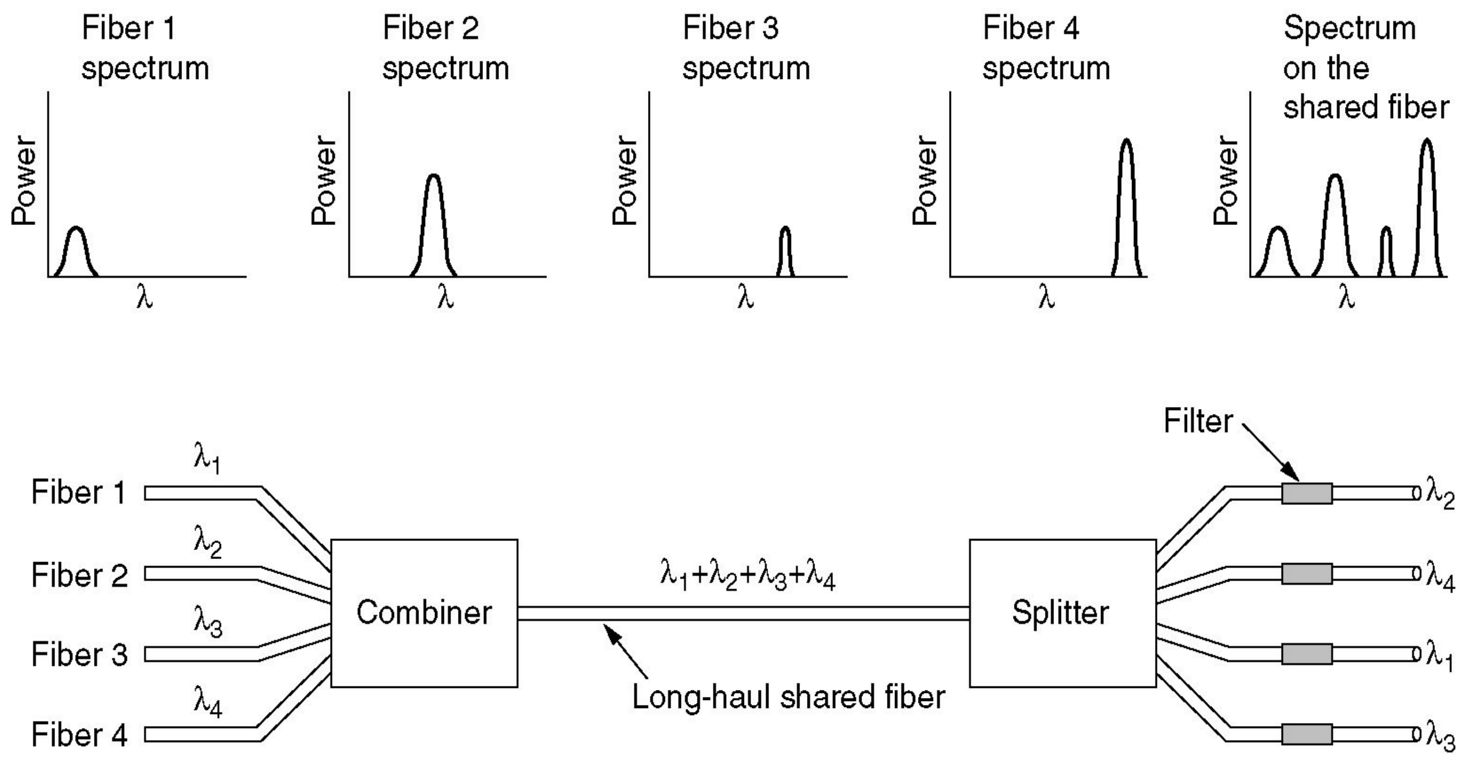
- Multiple signals are combined and transmitted over the channel
- Each signal is transmitted in different frequency ranges
- Typically used for analog transm
- Multiple implementations…





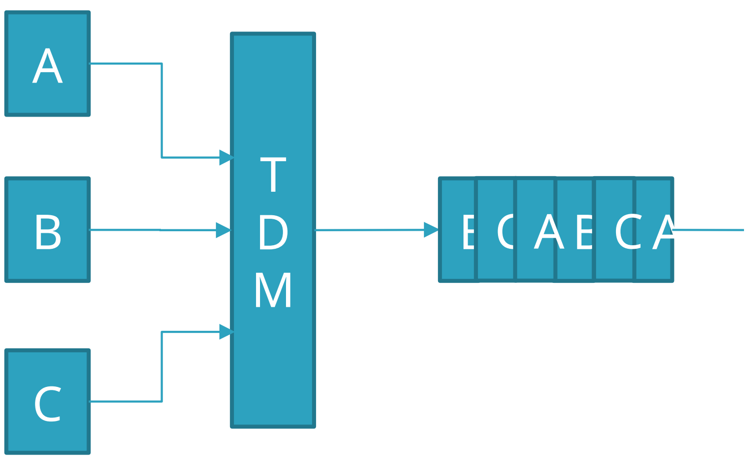
Wavelength-Division Multiplexing

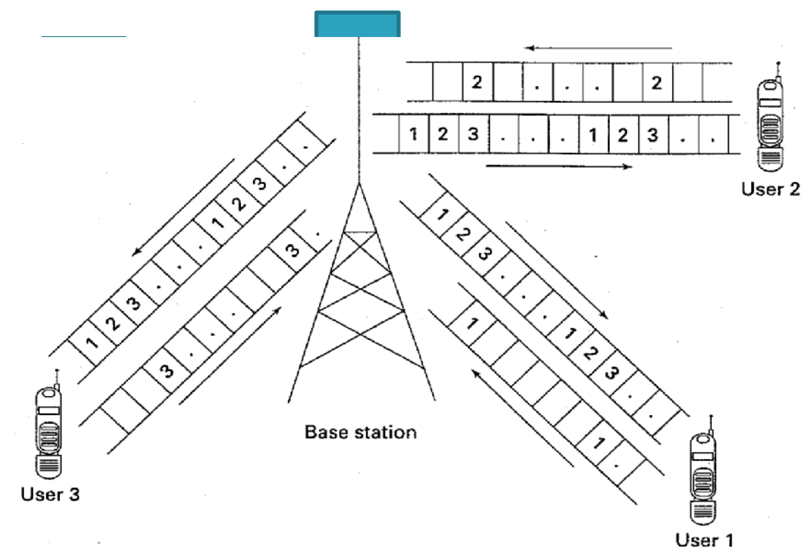
- Used for optical cables
- IR laser rays at different wavelengths



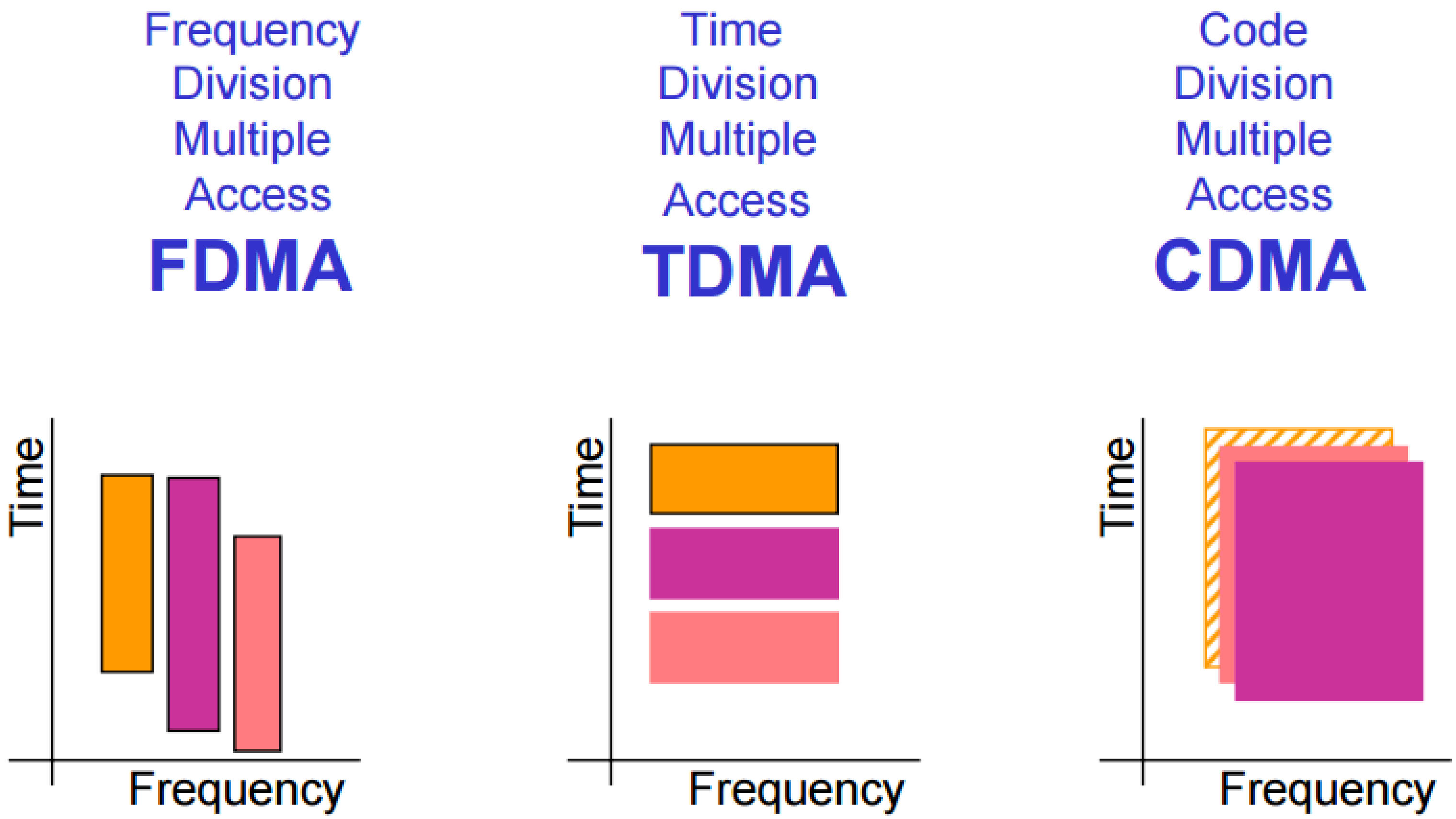
Time-Division Multiplexing

- Time is divided into not overlapping intervals
- Each time slot is assigned to a sender, exlusively.
- Empty slots may happen.





CDMA – Code Division Multiple Access



CDMA Analogy

- 10 people in a room.
  - 5 speak English, 2 speak Spanish, 2 speak Chinese, and 1 speaks Russian.
- Everyone is talking at relatively the same time over the same medium – the air.
- Who can listen to whom and why?
- Who can't you understand?
- Who can't speak to anyone else?

CDMA – Code Division Multiple Access

- Used by 3G and 4G cellular networks
- Each station can broadcast at any time in the full frequency spectrum
- The signals may interfere
  - Resulting in a linear combination of individual signals

- Algorithm
  - We assign a vector of length m to each station: v
    - Pairwise orthogonal vectors!!!
  - Each bit is encoded
    by the chip vector of the sender or it's complement: v or -v
  - If it sends bit 1, it transmits v
  - If it sends bit 0, it transmits -v

- Result is a sequence of vectors of length m

CDMA – Code Division Multiple Access

- Interference
  - A sends a,-a,a,a
  - B sends b,b,-b,-b
  - After interference we receive: a+b,-a+b,a-b,a-b ???

- How to decode?

CDMA – Code Division Multiple Access

- Interference
  - A sends a,-a,a,a
  - B sends b,b,-b,-b
  - After interference we receive: a+b,-a+b,a-b,a-b ???

- Decoding the message of A
  - Take the dot product by the sender’s chip code
    - (a+b)a > 0 => 1
    - (-a+b)a < 0 => 0
    - (a-b)a >0 => 1
    - (a-b)a > 0 => 1

If the dot product is
<0: bit 0 was sent by A
>0: bit 1 was sent by A
=0: nothing was sent by A
the channel is not used by A

Thank you…