Digital vs. Analog Data

Digital data: bits.

$\longrightarrow$ discrete signal

$\longrightarrow$ both in time and amplitude

Analog "data": audio/voice, video/image

$\longrightarrow$ continuous signal

$\longrightarrow$ both in time and amplitude

Both forms used in today's network environment.

$\longrightarrow$ burning CDs

$\longrightarrow$ audio/video playback

In broadband networks:

$\longrightarrow$ use analog signals to carry digital data

Important task: analog data is often digitized

$\longrightarrow$ useful: why?

$\longrightarrow$ it's convenient

$\longrightarrow$ use full power of digital computers

$\longrightarrow$ simple form: digital signal processing

$\longrightarrow$ analog computers are not as versatile/programmable

$\longrightarrow$ cf. "Computer and the Brain," von Neumann (1958)

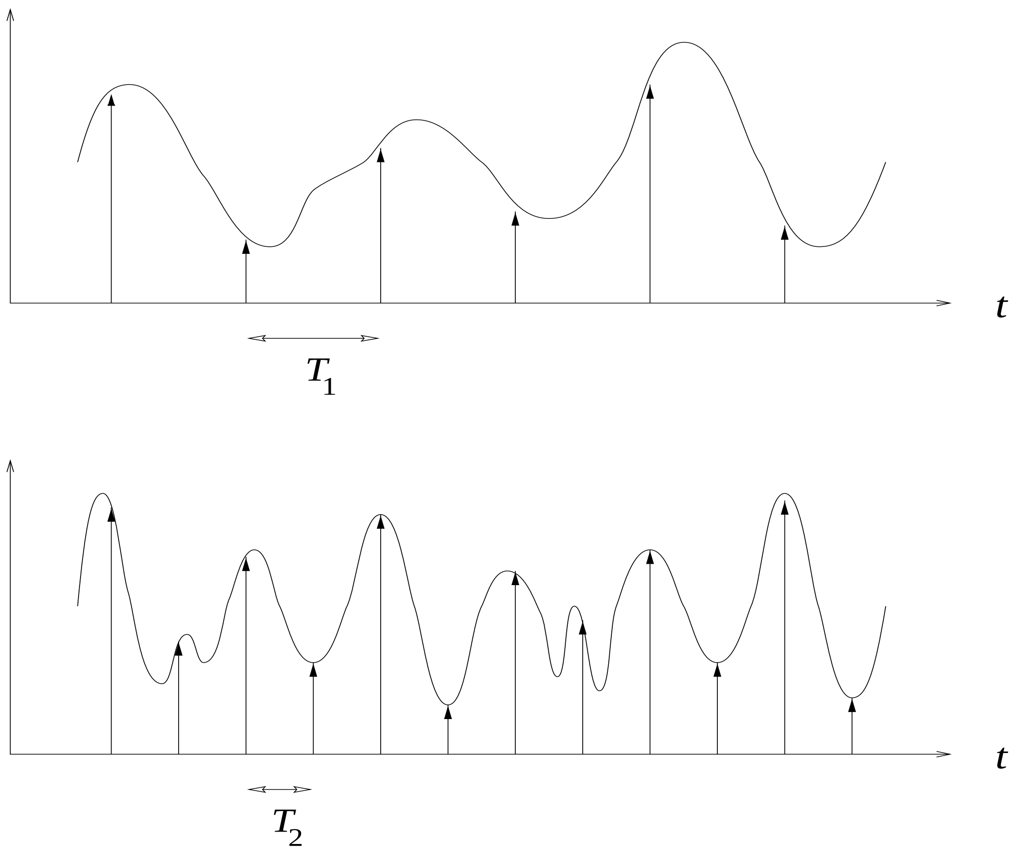
How to digitize such that digital representation is faithful?

$\longrightarrow$ sampling

$\longrightarrow$ interface between analog & digital world

Intuition behind sampling:

$\longrightarrow$ slowly vs. rapidly varying signal



If a signal varies quickly, need more samples to not miss details/changes.

$$\nu_1 = 1/T_1 \;\; < \;\; \nu_2 = 1/T_2$$

Sampling criterion for guaranteed faithfulness:

Sampling Theorem (Nyquist): Given continuous bandlimited signal $s(t)$ with $S(\omega) = 0$ for $|\omega| > W$, $s(t)$ can be reconstructed from its samples if

$$\nu \; > \; 2W$$

where ν is the sampling rate.

$\longrightarrow$ ν: samples per second

Remember simple rule: sample twice the bandwidth

Issue of digitizing amplitude/magnitude ignored

$\longrightarrow$ problem of quantization

$\longrightarrow$ possible source of information loss

$\longrightarrow$ exploit limitations of human perception

$\longrightarrow$ logarithmic scale

Compression

Information transmission over noiseless medium

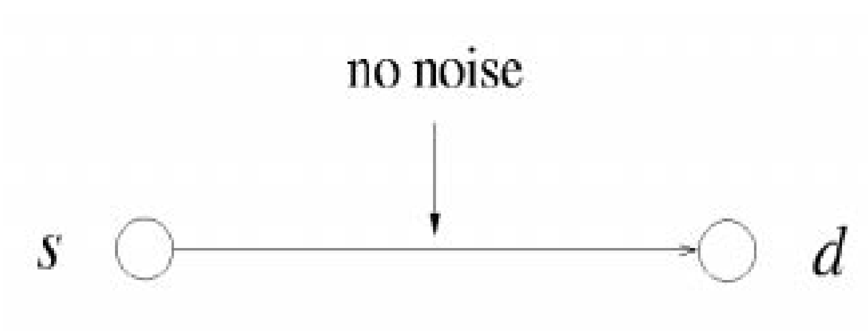
$\longrightarrow$ medium or "channel"

$\longrightarrow$ fancy name for copper wire, fiber, air/space

Sender wants to communicate information to receiver over noiseless channel.

$\longrightarrow$ can receive exactly what is sent

$\longrightarrow$ idealized scenario



Set-up:

$\longrightarrow$ take a system perspective

$\longrightarrow$ e.g., modem manufacturer

Need to specify two parts: property of data source—what are we supposed to send?—and how compression is done.

$\longrightarrow$ need to know what we're dealing with

$\longrightarrow$ if we want to do a good job compressing

Part I. What does the (data) source look like:

- source s emits symbols from finite alphabet set Σ

  $\rightarrow$ e.g., $\Sigma = \{0, 1\}$; $\Sigma =$ ASCII character set

- symbol $a \in \Sigma$ is generated with probability $p_a > 0$

  $\rightarrow$ e.g., books have known distribution for 'e', 'x' ...

  $\rightarrow$ let's play "Wheel of Fortune"

Part II. Compression machinery:

- code book F assigns code word $w_a = F(a)$ for each symbol $a \in \Sigma$

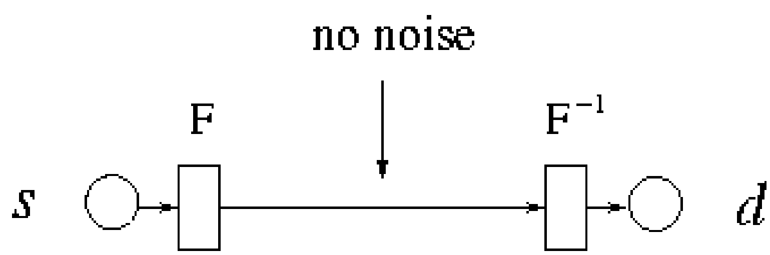
  $\rightarrow$ w_a is a binary string of length $|w_a|$

  $\rightarrow$ F could be just a table

- F is invertible

  $\rightarrow$ receiver d can recover a from w_a

  $\rightarrow$ F^{-1} is the same table, different look-up



Ex.: $\Sigma = \{A, C, G, T\}$; need at least two bits

- F^1: $w_A = 00$, $w_C = 01$, $w_G = 10$, $w_T = 11$
- F^2: $w_A = 0$, $w_C = 10$, $w_G = 110$, $w_T = 1110$

$\longrightarrow$ pros & cons?

Note: code book F is not unique

$\longrightarrow$ find a "good" code book

$\longrightarrow$ when is a code book good?

Performance (i.e., "goodness") measure: average code length L

$$L = \sum_{a \in \Sigma} p_a |w_a|$$

$\longrightarrow$ average number of bits consumed by given F

Ex.: If DNA sequence is 10000 letters long, then require on average $10000 \cdot L$ bits to be transmitted.

$\longrightarrow$ good to have code book with small L

Optimization problem: Given source $\langle \Sigma, \mathbf{p} \rangle$ where $\mathbf{p}$ is a probability vector, find a code book F with least L.

$\longrightarrow$ practically super-important

$\longrightarrow$ shrink-and-send

$\longrightarrow$ lossless shrinkage

A fundamental result on what is achievable to attain small L.

$\longrightarrow$ kind of like speed-of-light

First, define entropy H of source $\langle \Sigma, \mathbf{p} \rangle$

$$H = \sum_{a \in \Sigma} p_a \log \frac{1}{p_a}$$

Ex.: $\Sigma = \{A, C, G, T\}$; H is maximum if $p_A = p_C = p_G = p_T = 1/4$.

$\longrightarrow$ when is it minimum?

Source Coding Theorem (Shannon): For all code books F,

$$H \leq L_F$$

where L_F is the average code length under F.

Furthermore, L_F can be made to approach H by selecting better and better F.

Remark:

- to approach minimum H use blocks of k symbols

  → e.g., treat "THE" as one unit (not 3 separate letters)

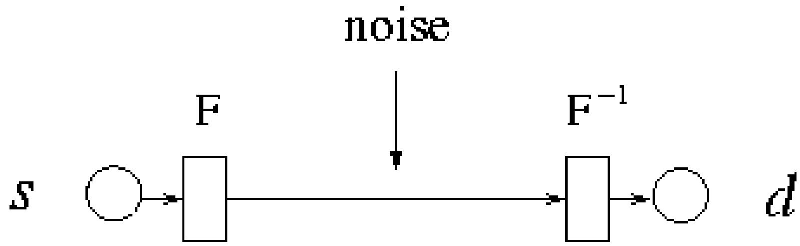
  → called extension code

- entropy is innate property of data source s
- limitation of ensemble viewpoint

  → e.g., sending number $\pi = 3.1415927\ldots$

  → better way?

Information Transmission under Noise



Uncertainty introduced by noise:

$\longrightarrow$ encoding/decoding: $a \mapsto w_a \mapsto w \mapsto [?]$

$\longrightarrow$ w_a gets corrupted, i.e., becomes w

$\longrightarrow$ if $w = w_b$, incorrectly conclude b as symbol

$\longrightarrow$ detect w is corrupted: error detection

$\longrightarrow$ correct w to w_a: error correction

Would like: if received code word $w = w_c$ for some symbol $c \in \Sigma$, then probability that actual symbol sent is indeed c is high

$\longrightarrow$ $\Pr\{\text{actual symbol sent} = c \,|\, w = w_c\} \approx 1$

$\longrightarrow$ noiseless channel: special case (prob $= 1$)

In practice, w may not match any legal code word:

$\longrightarrow$ for all $c \in \Sigma$, $w \neq w_c$

$\longrightarrow$ good or bad?

$\longrightarrow$ what's next?

Shannon showed that there is a fundamental limitation to reliable data transmission.

$\rightarrow$ the noisier the channel, the smaller the reliable throughput

$\rightarrow$ overhead spent dealing with bit flips

Definition of channel capacity C: maximum achievable reliable data transmission rate (bps) over a noisy channel (dB) with bandwidth W (Hz).

Channel Coding Theorem (Shannon): Given bandwidth W, signal power P_S, noise power P_N, channel subject to white noise,

$$C \;=\; W \log\left(1 + \frac{P_S}{P_N}\right) \text{ bps.}$$

P_S/P_N: signal-to-noise ratio (SNR)

$\longrightarrow$ upper bound achieved by using longer codes

$\longrightarrow$ detailed set-up/conditions omitted

Increasingly important for modern day networking:

- Power control (e.g., pocket PCs)

  $\longrightarrow$ trade-off w.r.t. battery power

  $\longrightarrow$ trade-off w.r.t. multi-user interference

  $\longrightarrow$ signal-to-interference ratio (SIR)

- Recent trend: software radio

  $\longrightarrow$ hardware-to-software migration

  $\longrightarrow$ kind of like cordless phones (e.g., 2.4 GHz)

  $\longrightarrow$ configurable: make it programmable

Signal-to-noise ratio (SNR) is expressed as

$$\text{dB} \;=\; 10 \log_{10}(P_S/P_N).$$

Example: Assuming a decibel level of 30, what is the channel capacity of a telephone line?

Answer: First, $W = 3000$ Hz, $P_S/P_N = 1000$. Using Channel Coding Theorem,

$$C \;=\; 3000 \log 1001 \;\approx\; 30 \text{ kbps}.$$

$\longrightarrow$ compare against 28.8 kbps modems

$\longrightarrow$ what about 56 kbps modems?

$\longrightarrow$ DSL lines?

Digital vs. Analog Transmission

Two forms of *transmission*:

- digital transmission: data transmission using square waves
- analog transmission: data transmission using all other waves

Four possibilities to consider:

- analog data via analog transmission

  → "as is" (e.g., radio)
- analog data via digital transmission

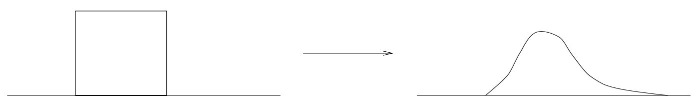
  → sampling (e.g., voice, audio, video)
- digital data via analog transmission

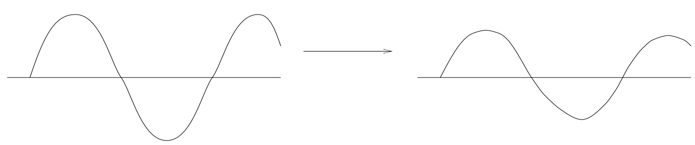
  → broadband & wireless ("high-speed networks")
- digital data via digital transmission

  → baseband (e.g., Ethernet)

Why consider digital transmission?

Common to both: problem of *attenuation*.





- decrease in signal strength as a function of distance
- increase in attenuation as a function of frequency

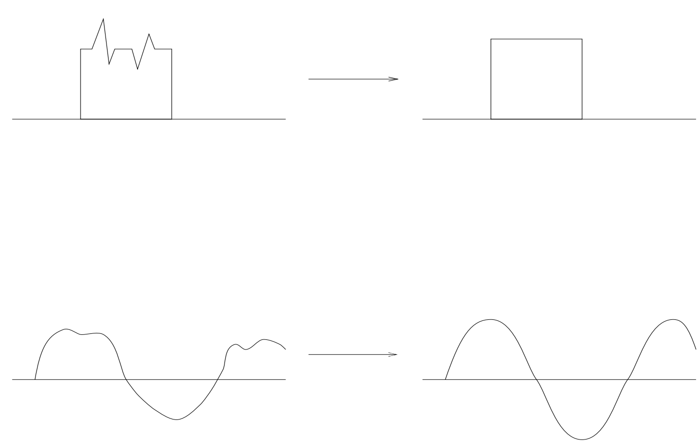
Rejuvenation of signal via amplifiers (analog) and repeaters (digital).

Delay distortion: different frequency components travel at different speeds.

Most problematic: effect of noise

$\longrightarrow$ thermal, interference, ...

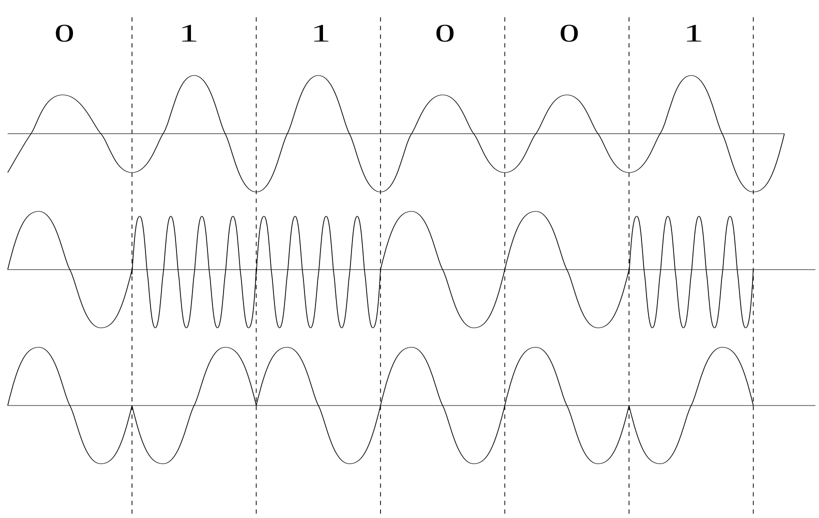
- Analog: Amplification also amplifies noise—filtering out just noise, in general, is a complex problem.
- Digital: Repeater just generates a new square wave; more resilient against ambiguitity.



Analog Transmission of Digital Data

Three pieces of information to manipulate: amplitude, frequency, phase.

- Amplitude modulation (AM): encode bits using amplitude levels.
- Frequency modulation (FM): encode bits using frequency differences.
- Phase modulation (PM): encode bits using phase shifts.



FM radio uses . . . FM!

AM radio uses . . . AM!

iPod & radio experiment uses . . . ?

Why is FM radio clearer ("high fidelity") than AM radio?

Broadband uses . . . ?

Baud Rate vs. Bit Rate

Baud rate: Unit of time within which carrier wave can be altered for AM, FM, or PM.

$\longrightarrow$ signalling rate

$\longrightarrow$ e.g., clock

Not synonymous with bit rate: e.g., AM with 8 levels, PM with 8 phases

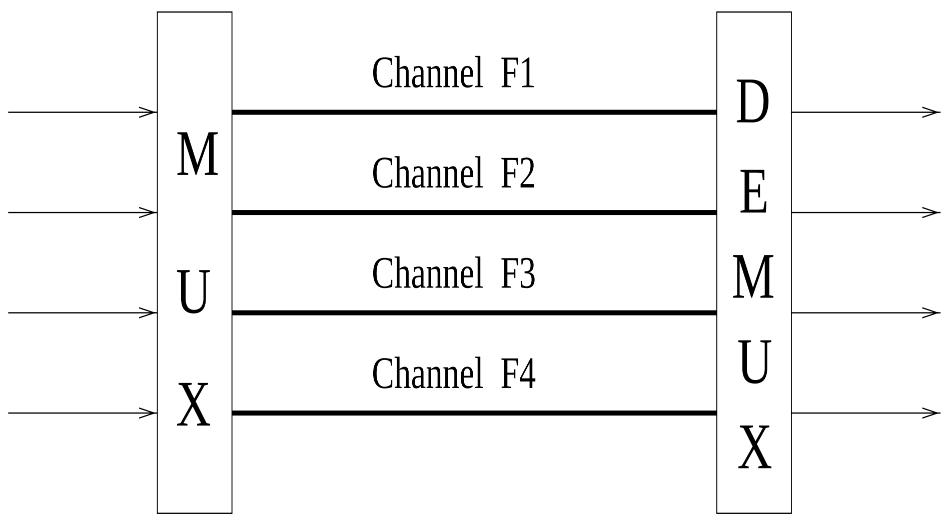
$\longrightarrow$ bit rate (bps) = 3 $\times$ baud rate

. . . less than one bit per baud?

Broadband vs. Baseband

Presence or absence of carrier wave: allows many channels to co-exist at the same time

$\longrightarrow$ frequency division multiplexing (FDM)



Ex.: AM radio (535 kHz–1705 kHz)

$\longrightarrow$ tuning to specific frequency: Fourier transform

$\longrightarrow$ coefficient (magnitude) carries bit information

Ex.: FM radio

$\longrightarrow$ 88 MHz–108 MHz

$\longrightarrow$ 200 kHz slices

$\longrightarrow$ how does it work?

$\longrightarrow$ better or worse than AM?

Ex.: Digital radio

$\longrightarrow$ digital audio radio service

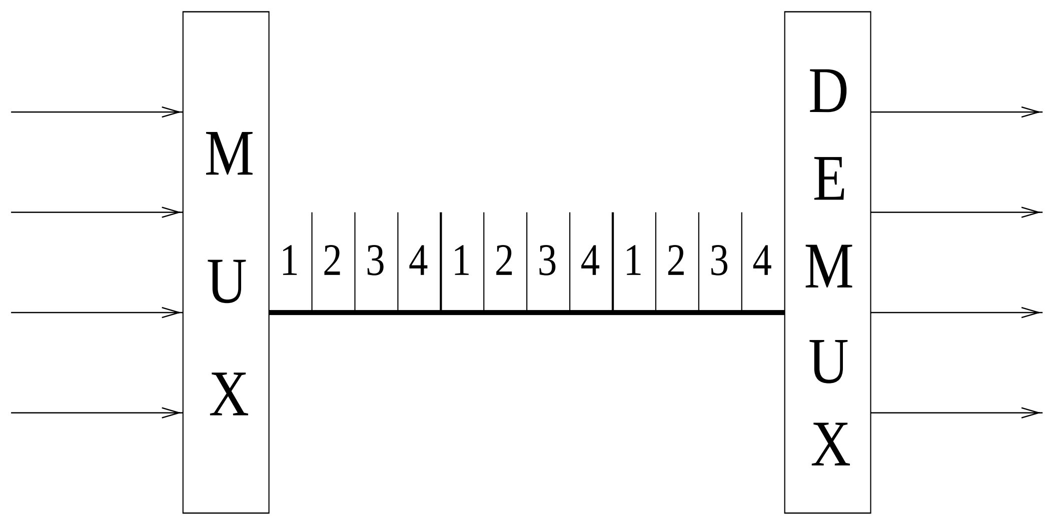
$\longrightarrow$ GEO satellites (a.k.a. satellite radio)

$\longrightarrow$ uses 2.3 GHz spectrum (a.k.a. S-band)

$\longrightarrow$ e.g., XM, Sirius

In the absence of carrier wave, can still use multiplexing:

$\longrightarrow$ time-division multiplexing (TDM)



- digital transmission of analog data

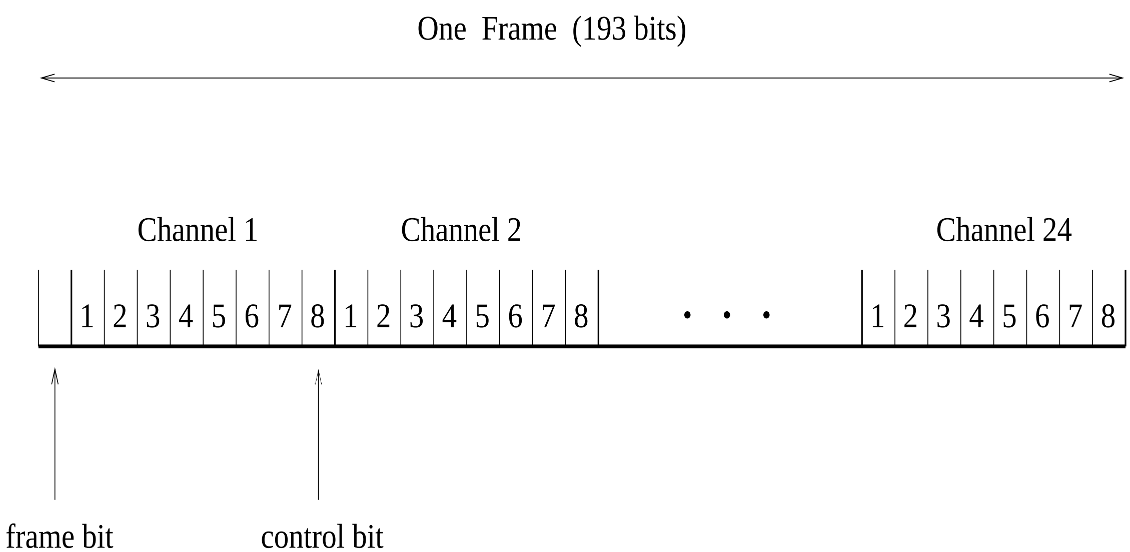
  → first digitize

  → PCM (e.g., PC sound cards), modem

- digital transmission of digital data

  → e.g., telephony backbone network

Example: T1 carrier (1.544 Mbps)



- 24 simultaneous users
- 7 bit quantization

Assuming 4 kHz telephone channel bandwidth, Sampling Theorem dictates 8000 samples per second.

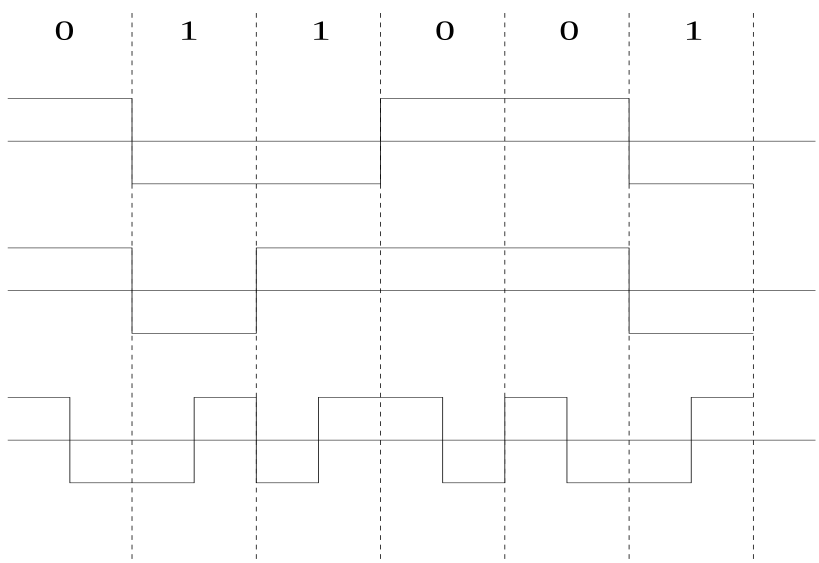
$\longrightarrow$ 125 μsec inter-sample interval

Bandwidth = $8000 \times 193 = 1.544$ Mbps

Digital Transmission of Digital Data

Direct encoding of square waves using voltage differentials; e.g., -15V–+15V for RS-232-C.

- NRZ-L (non-return to zero, level)
- NRZI (NRZ invert on ones)
- Manchester (biphase or self-clocking codes)



$\longrightarrow$ baud rate vs. bit rate of Manchester?

Trade-offs:

- NRZ codes—long sequences of 0's (or 1's) causes synchronization problem; need extra control line (clock) or sensitive signalling equipment.
- Manchester codes—synchronization achieved through self-clocking; however, achieves only 50% efficiency vis-à-vis NRZ codes.

4B/5B code

Encode 4 bits of data using 5 bit code where the code word has at most one leading 0 and two trailing 0's.

$0000 \leftrightarrow 11110$, $0001 \leftrightarrow 01001$, etc.

$\longrightarrow$ at most three consecutive 0's

$\longrightarrow$ efficiency: 80%

Multiplexing techniques:

- TDM
- FDM
- mixture (FDM + TDM); e.g., TDMA
- CDMA (code division multiple access) or spread spectrum

  → wireless communication

  → competing scheme with TDMA