# Perceptual Coding

- Lossless vs. lossy compression
- Perceptual models
- Selecting info to eliminate
- Quantization and entropy encoding

- Part II wrap up

# Lossless vs. Lossy Compression

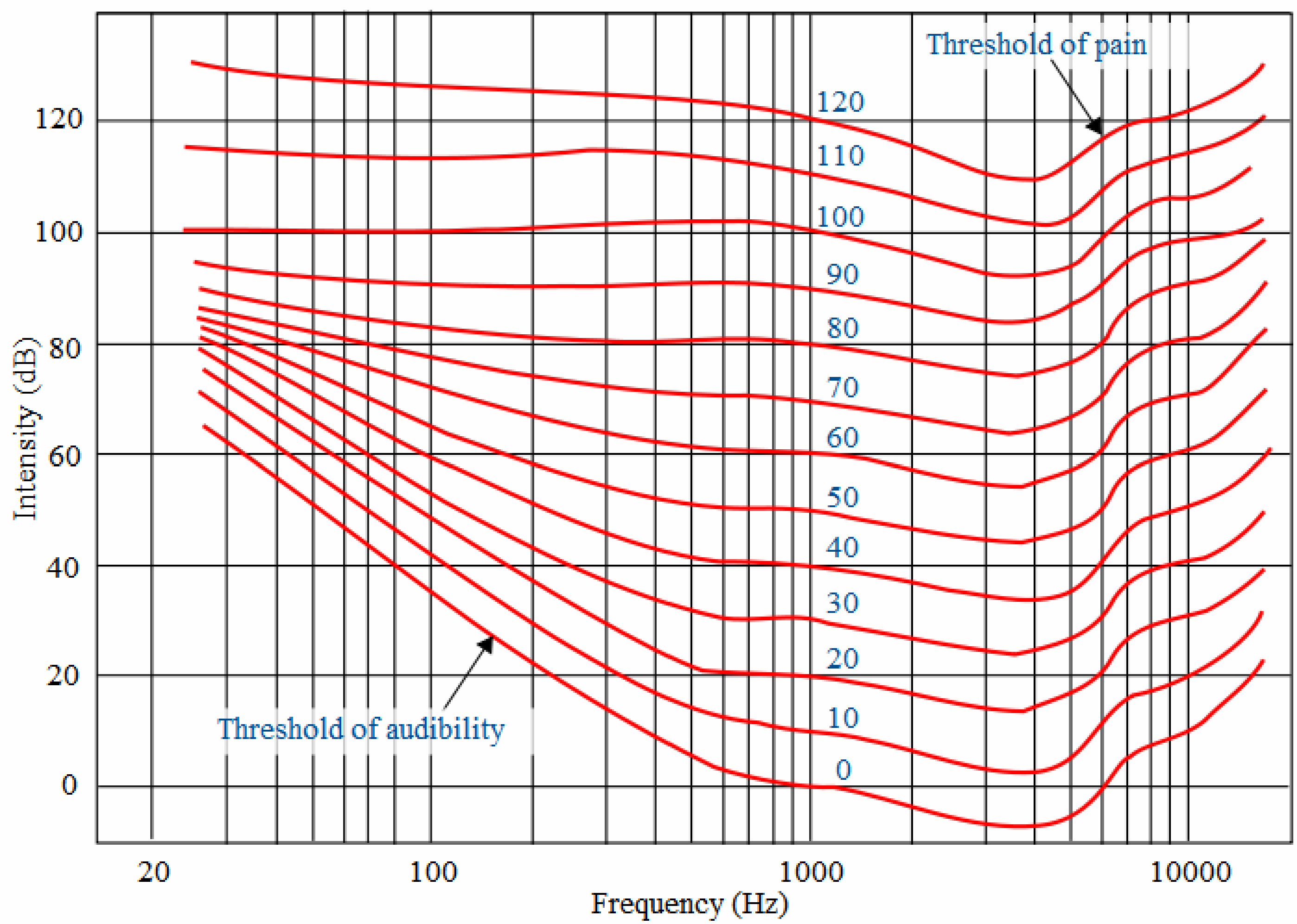

- Huffman and LZW encodings are *lossless*, i.e., we can reconstruct the original bitstream exactly: $bits_{OUT} = bits_{IN}$.
  - What we want for "naturally digital" bitstreams (documents, messages, datasets, …)
- Any use for *lossy* encodings: $bits_{OUT} \approx bits_{IN}$?
  - "Essential" information preserved
  - Appropriate for sampled bitstreams (audio, video) intended for human consumption via imperfect sensors (ears, eyes).

# Perceptual Coding

- Start by evaluating input response of bitstream consumer (eg, human ears or eyes), i.e., how consumer will perceive the input.
  - Frequency range, amplitude sensitivity, color response, …
  - Masking effects (more on this later)
- Identify information that can be removed from bitstream without noticeable effect, e.g.,
  - Sounds outside frequency range, or masked sounds
  - Visual detail below resolution limit (color, spatial detail)
  - Info beyond maximum allowed output bitrate
- Encode remaining information efficiently
  - Use DCT-based transformations
  - Quantize DCT coefficients
  - Entropy code (eg, Huffman encoding) results

# Perceptual Coding Example: Audio

## Input response of human ear

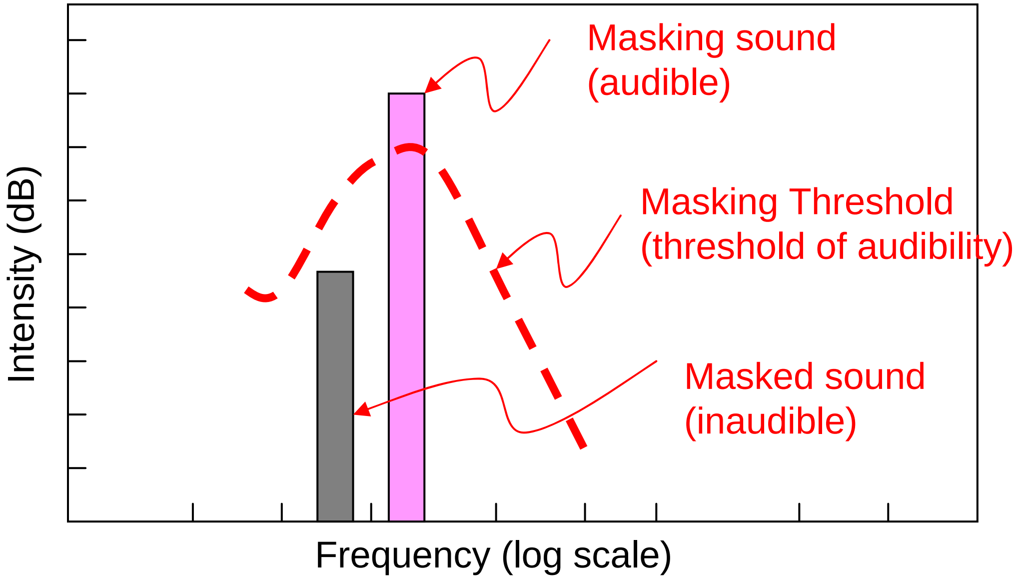



See http://en.wikipedia.org/wiki/Psychoacoustics

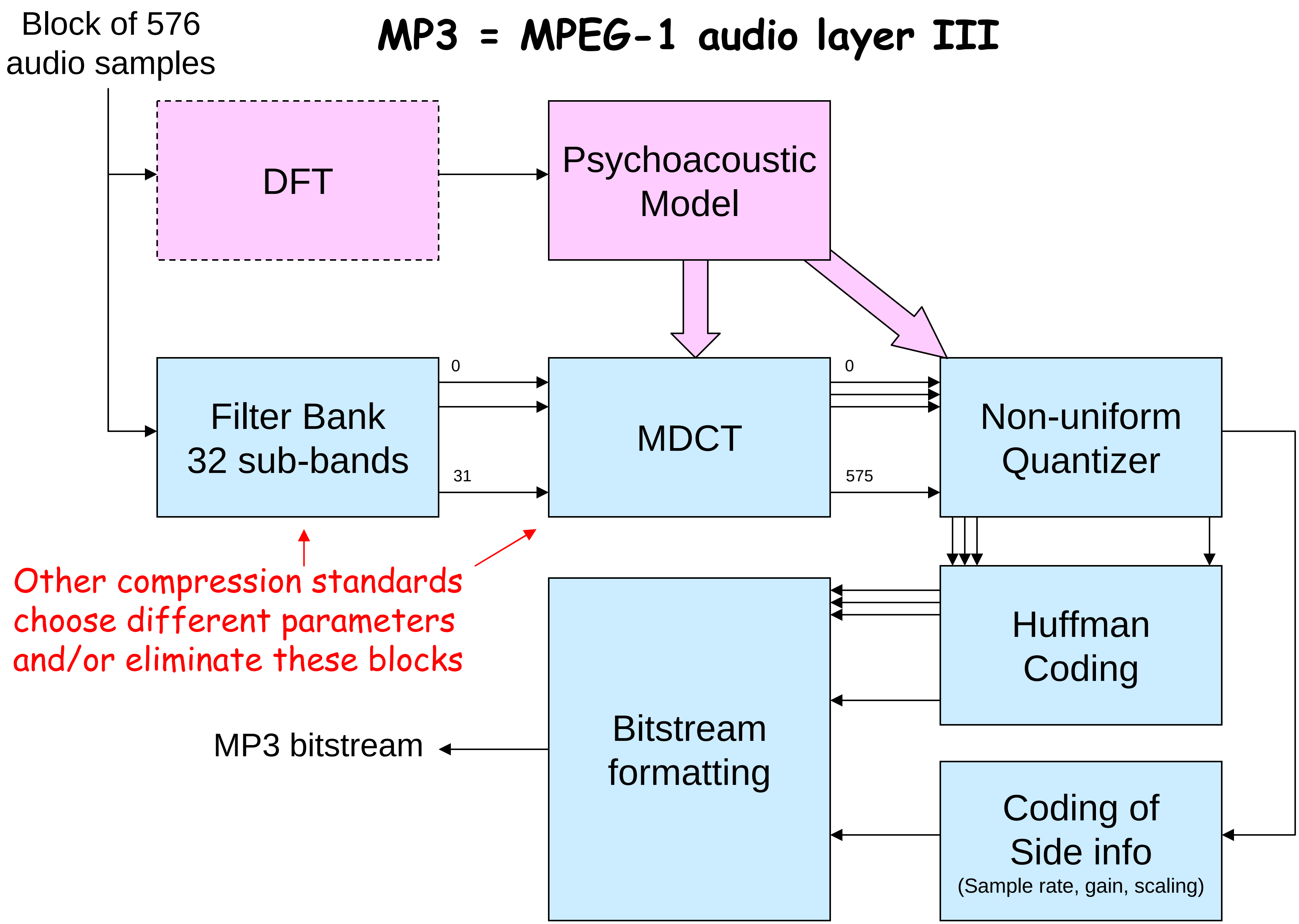
# Psychoacoustic Model

**Bark scale identifies critical bands, which are frequency ranges in which one pitch will mask another pitch. The masking phenomenon occurs because any loud sound will distort the absolute threshold of audibility, making quieter, otherwise perceptible sounds inaudible. Masking can be "simultaneous" (nearby in frequency) and "temporal" (nearby in time).**

24 bands with edges at 0, 100, 200, 300, 400, 510, 630, 770, 920, 1080, 1270, 1480, 1720, 2000, 2320, 2700, 3150, 3700, 4400, 5300, 6400, 7700, 9500, 12000, 15500 Hz.



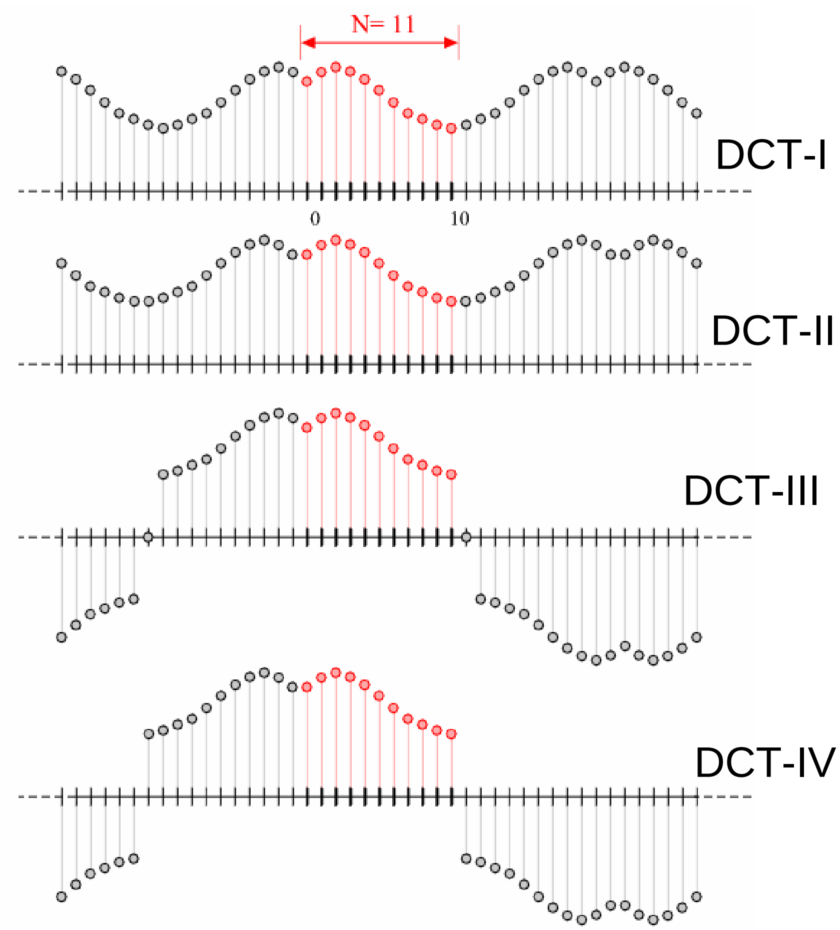
# Overview of MP3 audio compression



# Discrete Cosine Transform (DCT)

**DCT-IV:**

$$X_k = \sum_{n=0}^{N-1} x_n \cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)\left(k+\frac{1}{2}\right)\right]$$




DFTs imply periodic extension of functions, but DCTs imply an even extension (no discontinuities at boundaries between blocks).

Figure at left shows implicit extensions of DCT input data for N=11 points (red dots) for common types of DCT.

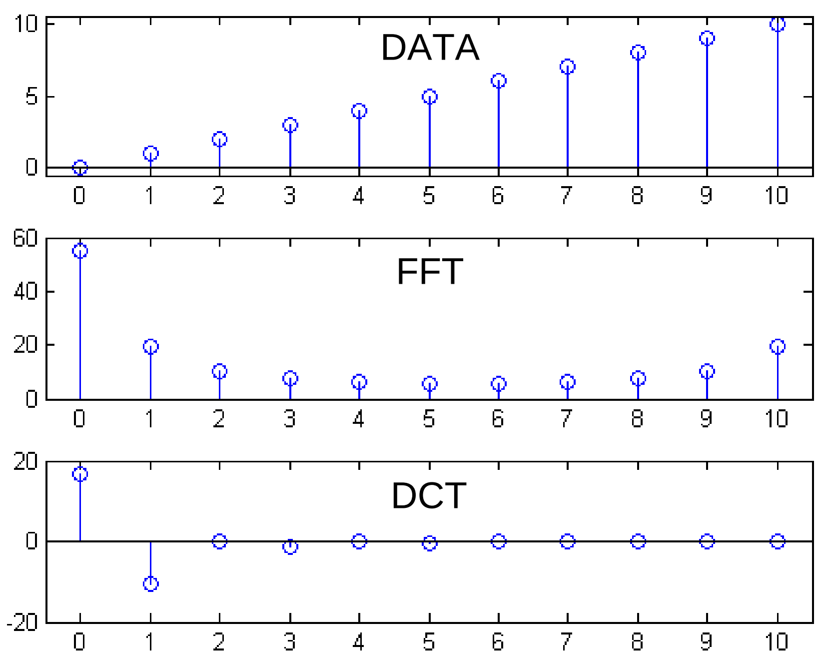
MDCT = variant of DCT-IV where data blocks overlap (for better continuity between blocks)

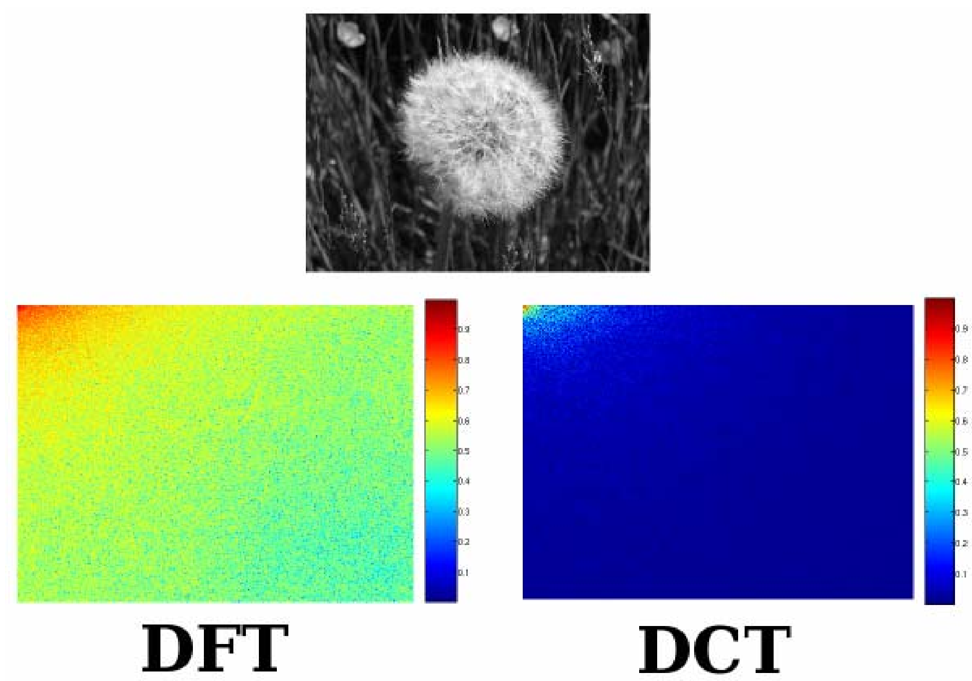


See http://en.wikipedia.org/wiki/Discrete_Cosine_Transform

# DCT vs. FFT

- DCTs produce real instead of complex outputs
- DCT has a strong "energy compaction" property: most of the signal information tends to be concentrated in a few low-frequency components of the DCT:



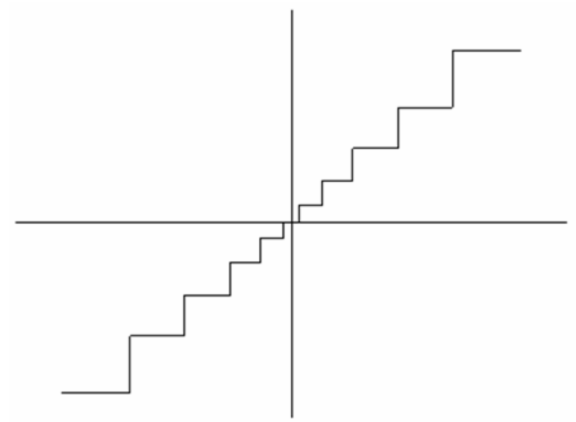






See http://en.wikipedia.org/wiki/Discrete_Cosine_Transform

# Non-Uniform Quantization & Coding

- Quantization: map coefficients into smallish number of discrete values.
- Non-Uniform: select number of levels according to amplitude of coefficients
  - Small amplitude
    → smaller steps
    → greater accuracy
    → better SNR
  - Just keep quantization noise below audible threshold (scaling)



- Probability of quantized values is not uniform
  - Huffman coding is effective at compression
  - Different codes for 3 different frequency bands
  - Adjust number of quantization levels so that Huffman data size < available bits (global gain)

# Other opportunities for compression

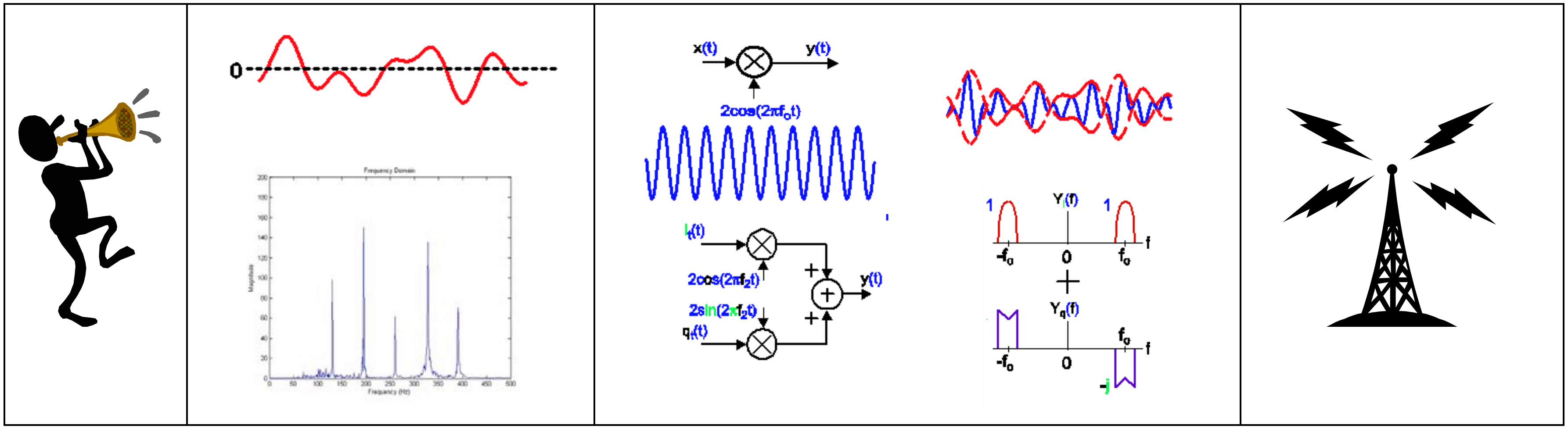
- Variable bit rate encoding
  - Adjust bitrate to media stream complexity
- Take advantage of omnidirectionality of low-frequency sounds
  - Only do stereo encoding at higher frequencies
- Voice-only
  - "radio quality audiobooks": mono, reduced sample rate, low-bit rate

# Summary

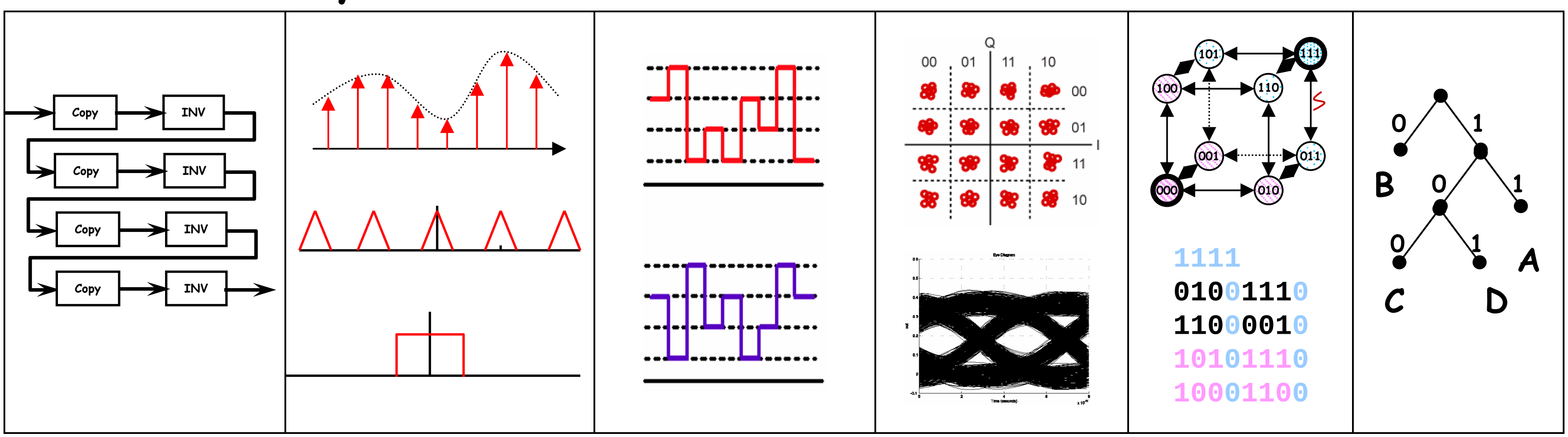
- Perceptual model can identify information that can be removed without being missed
- DCT is well-suited for media compression
  - Boundary conditions for block-to-block continuity
  - Energy compaction properties
- Non-uniform quantization
  - Adjust steps according to amplitude
  - Differing probabilities $\rightarrow$ Huffman coding
  - Scale factor: quantization noise < perception threshold
  - Global gain: Huffman data < available bit rate

# 6.082 up until now…

## Part I: volt-by-volt




## Part II: bit-by-bit



# Big Ideas

**Multiple representations**
- time vs. frequency

**Communicating Information**
- managing signal vs. noise
- encode as voltages, amplitude, phase
- power, bandwidth, channel capacity

**Digital Abstraction**
- errors accumulate
- continuous → discrete
- restoration of values

**Information & Encodings**
- info can be measured: entropy
- add redundancy → correct errors
- remove redundancy → compression

**Systems engineering**
- composable behaviors
- uniform representations
- layered architectures