المحور الخامس

Basics of Digital Audio (20^{th} 21^{st})

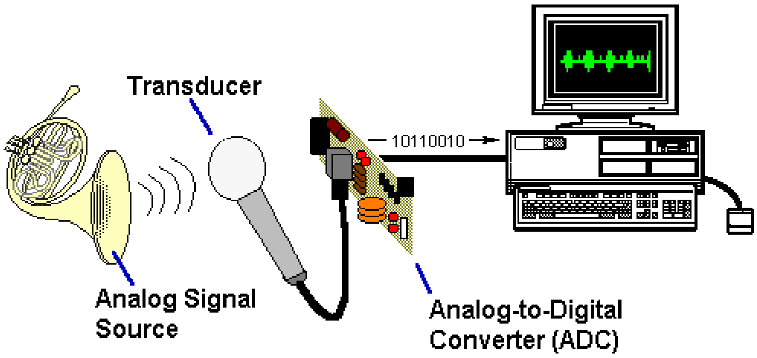
Sound is a wave phenomenon like light, but it is macroscopic and involves molecules of air being compressed and expanded under the action of some physical device.

Sound is a vibration that typically propagates as an audible wave of pressure, through a transmission medium such as a gas, liquid or solid.

The simplest kind of sound wave is a **sine wave**. Pure sine waves rarely exist in the natural world, but they are a useful place to start because all other sounds can be broken down into combinations of sine waves. Even though such pressure waves are longitudinal, they still have ordinary wave properties and behaviors, such as reflection (bouncing), refraction (change of angle when entering a medium with a different density) and diffraction (bending around an obstacle).

1- Detecting Sound: Since sound consists of measurable pressures at any 3D point, we can detect it by measuring the pressure level at a location, using a transducer to convert pressure to voltage levels.

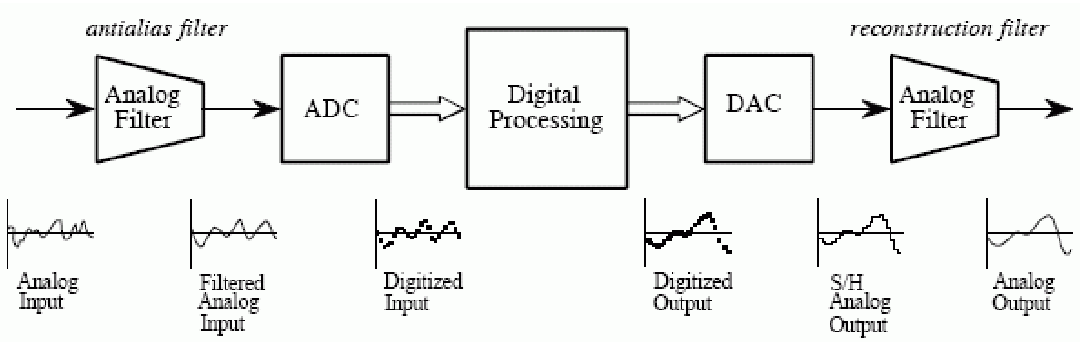
Microphone: Receives sound and converts it to an analog signal.



If we wish to use a digital version of sound waves we must form digitized representations of audio information.

2- The computer needs discrete entities: Analog-to-Digital converter is used Dedicated Hardware (e.g. Sound card). Also known as Digital Sampling.

- Anti-aliasing filters (major part of Analog Conditioning) are needed at the input to remove frequencies above the sampling limit that would result in aliasing. The anti-aliasing filter at the output removes the aliases that result from the sampling.
- After the anti-aliasing filter, the analog/digital converter (ADC) quantises the continuous input into discrete levels.
- After digital processing, the output of the system is given to a digital/analog converter (DAC) which converts the discrete levels into continuous voltages or currents.
- This output must also be filtered with a low pass filter to remove the aliases from the sampling. Subsequent processing can include further filtering, mixing, or other operations.

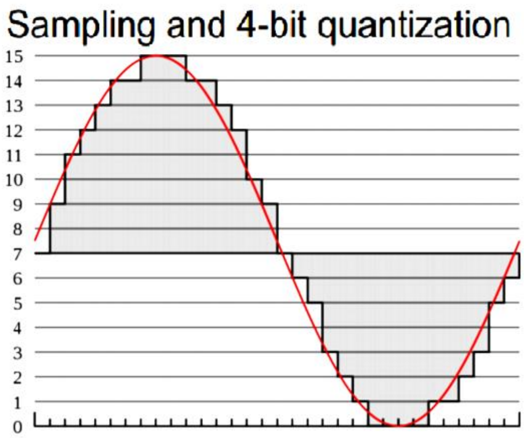


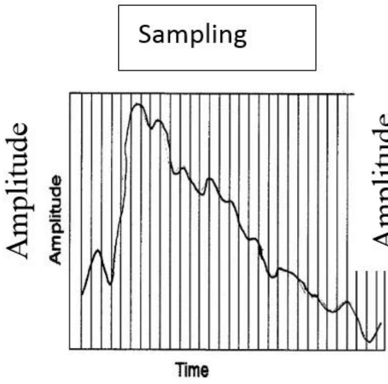
Digitization means conversion to a stream of numbers, and preferably these numbers should be integers for efficiency.

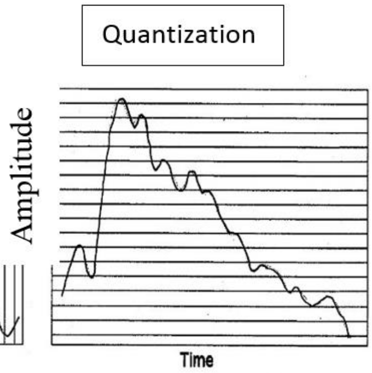
Sampling is digitizing the analog signal in the time dimension;

Quantization is digitizing the analog signal in the amplitude dimension.

16 bits = 65,536 levels (Quantization)



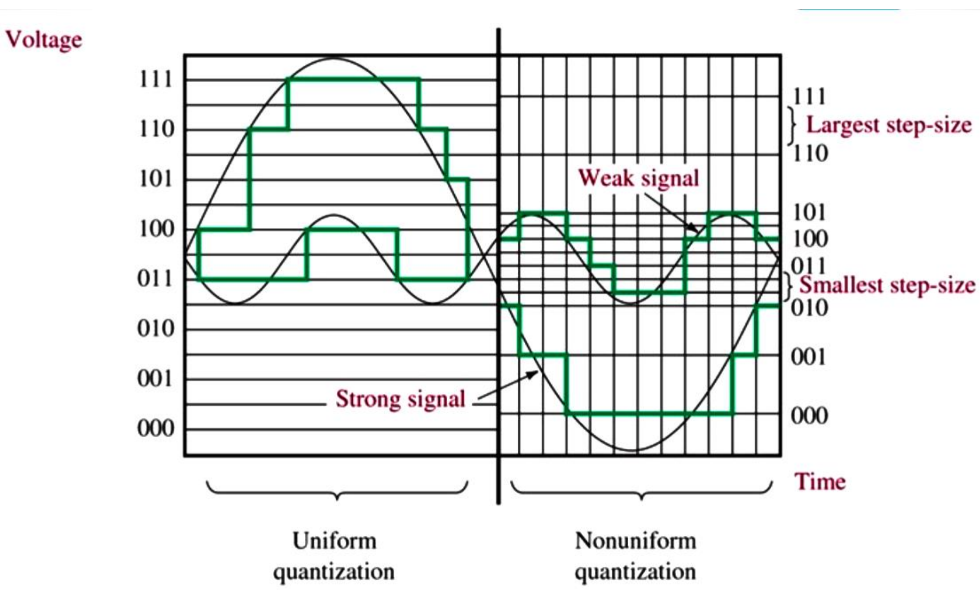




Sampling	Quantization
Sampling rate: Number of samples per second (measured in Hz)	3-bit quantization gives 8 possible sample values
E.g., CD standard audio uses a sampling rate of 44,100 Hz (44100 samples per second)	E.g., CD standard audio uses 16-bit quantization giving 65536 values.

We have discussed only **uniform sampling**, with equally spaced sampling intervals. Typical uniform quantization rates are 8-bit and 16-bit; 8-bit quantization divides the vertical axis into 256 levels, and 16-bit divides it into 65,536 levels.

Non-uniform sampling is also possible. This is not used for sampling in time, but is used for quantization (the μ-law). We call it non-linear If its logarithmic. The non-linear scale is used because small amplitude signals are more likely to occur than large amplitude signals, and they are less likely to mask any noise.



Data rate = sample rate * quantization * channel

Q) Compare rates for CD vs. mono audio?

Mono audio: ***Data rate =*** 8000 samples/second * 8 bits/sample * 1 channel

= 7.8 kBytes / second

CD: ***Data rate =*** 44,100 samples/second * 16 bits/sample * 2 channels

= 172.26 KBytes / second ~= 10MB / minute

File size = (Data rate * time) + Header file

= sample rate * quantization * channel* time in second

Q) what is file size for CD data rate with 10 minutes

File size= 172.26 Kbytes * 60_{second} * $10_{minunts}$= 100.9Mbyts

Audio Quality vs. Data Rate

Quality	***Sample Rate (kHz)***	***Bits per Sample***	***Mono/ Stereo***	***Data Rate (kBytes/sec) (uncompressed)***	***Frequency Band***
Telephone	8	8	Mono	8	200-3400 Hz
AM Radio	11.025	8	Mono	11.0	540-1700KHz
FM Radio	22.050	16	Stereo	88.2	
CD	44.1	16	Stereo	176.4	20-20000 Hz
DAT	48	16	Stereo	192.0	20-20000 Hz

Nyquist Theorem (22nd)

As a simple illustration, Fig. (1,a) shows a single sinusoid: it is a single, pure, frequency (only electronic instruments can create such boring sounds). Now if the sampling rate just equals the actual frequency, we can see from Fig. (1,b) that a false signal is detected: it is simply a constant, with zero frequency. On the other hand, we sample at 1.5 times the frequency, Fig. (1,c) shows that we obtain an **incorrect (*alias*) frequency** that is lower than the correct one. it is half the correct one (the wavelength, from peak to peak, is double that of the actual signal).

An alias is any artifact that does not belong to the original signal. Thus, for correct sampling we must use a sampling rate equal to at least **twice the maximum frequency** content in the signal. This is called the **Nyquist rate**.

$$f_s \geq 2f_c$$

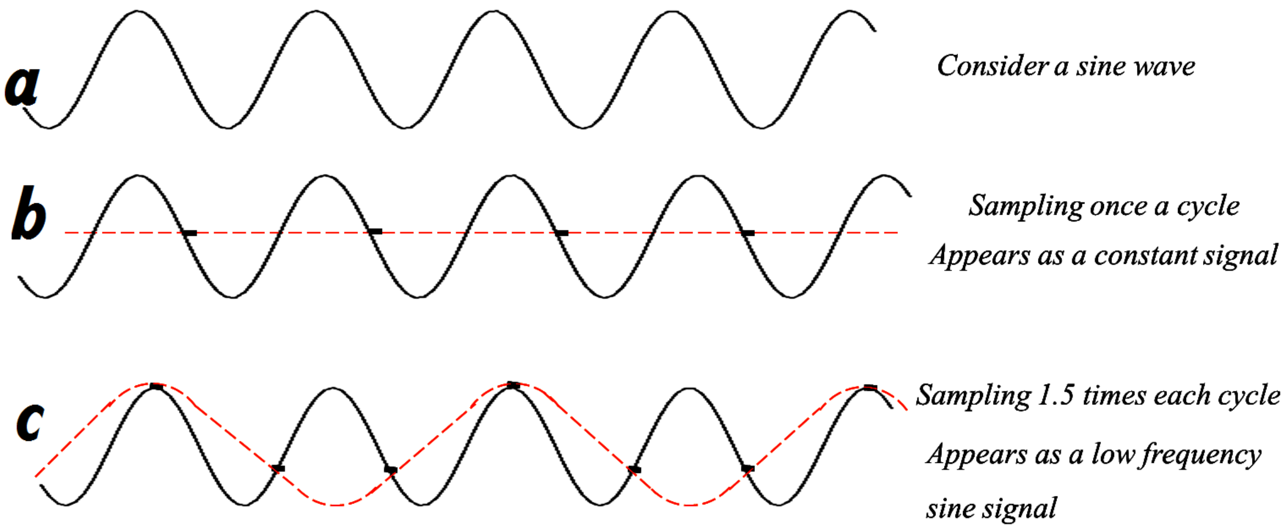


Fig. 1: Aliasing:

(a) a single frequency;

(b) sampling at exactly the frequency produces a constant;

(c) sampling at 1.5 times per cycle produces an alias frequency that is perceived

What will happen if we get Nyquist Sampling Wrong?

Digital Sampling Artifacts Arise - Effect known as Aliasing which affects Audio, Generally, if a signal is band-limited—that is, if it has a **lower limit *f1*** and an **upper limit *f2*** of frequency components in the signal—then we need a sampling rate of at least **2(*f2* − *f1*).**

<u>Nyquist theorem is used to calculate the optimum sampling rate in order to obtain good audio quality</u>.

For audio, typical sampling rates are from 8 kHz (8,000 samples per second) to 48 kHz. This range is determined by the Nyquist theorem. Humans have a range of hearing from 20 Hz (low) to 20,000 Hz (high)

Q: Why are CD Sample Rates 44.1 KHz?

The upper range of human hearing is around 20Hz - 22 KHz. Apply Nyquist Theorem ***(2× frequencies)***. Therefore, sampling at twice the maximum frequency (44 KHz) we could have achieved good audio quality.

Suppose we have a fixed sampling rate. Since it would be impossible to recover frequencies higher than half the sampling rate in any event, most systems have an anti-aliasing filter that restricts the frequency content of the sampler's input to a range at or below half the sampling frequency. The frequency equal to half the Nyquist rate is called the Nyquist frequency.

$$\boxed{f_{alias} = f_{sampling} - f_{true}, \quad \text{for} \quad f_{true} < f_{sampling} < 2 \times f_{true}}$$

Example,

If the true frequency is 5.5 kHz and the sampling frequency is 8 kHz, then the alias frequency is 2.5 kHz: So, if again the sampling frequency is less than twice the true frequency and is less than the true frequency, then the alias frequency equals ***n*** times the sampling frequency minus the true frequency, where ***n*** is the lowest integer that makes ***n*** times the sampling frequency larger than the true frequency.

For example, when the true frequency is between 1.0 and 1.5 times the sampling frequency, the alias frequency equals the true frequency minus the sampling frequency.

In general, the apparent frequency of a sinusoid is the lowest frequency of a sinusoid that has exactly the same samples as the input sinusoid.

Data Level

A digital signal has eight levels. How many bits are needed per level?

$$\boldsymbol{Number\ of\ bits\ per\ level\ = log_2 8 = 3}$$

Noiseless - Nyquist Theorem

Nyquist gives the upper bound for the bit rate of a transmission system by calculating the bit rate directly from the number of bits in a symbol (or signal levels) and the bandwidth of the system.

Nyquist theorem states that for a noiseless channel:

$$\boxed{C = 2\, B\, \log_2 2^n}$$

C= capacity in bps, B = bandwidth in Hz, n= number of bits

Example: ***Consider a noiseless channel with a bandwidth of 3000 Hz transmitting a signal with two signal levels.***

Number of bits per level= $\log_2 2= 1$

$C = 2 \text{ X } (3000) \log_2 2^1= 6000$ bps.

Example: Consider the same noiseless channel transmitting a signal with four signal levels

(for each level, we send 2 bits).

Number of bits per level= $\log_2 4 = 2$

$C = 2 \text{ X } (3000) \log_2 2^2 = 1200$ bps.

Example: ***Consider the telephone channel having bandwidth B = 4 kHz. Assuming there is no noise, determine channel capacity for the following encoding levels: (i) 2, and (ii) 128.***

(i) $C = 2B = 2 \times 4000 = 8$ Kbits/s

(ii) $C = 2 \times 4000 \times \log_2 128 = 8000 \times 7 = 56$ Kbits/s

Example: **Television channels are 6 MHz wide. How many bits/sec can be sent if four-level digital signals are used? Assume a noiseless channel.**

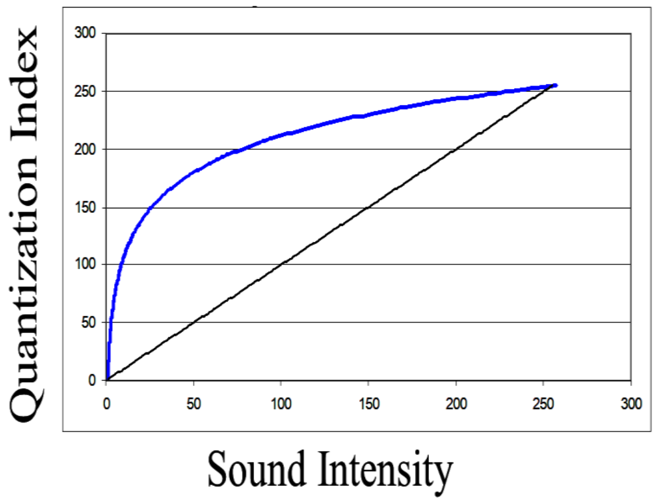
$C = 2 \text{ X } (6 \text{ X } 10^6) \log_2 4 = 24$ Mbps.

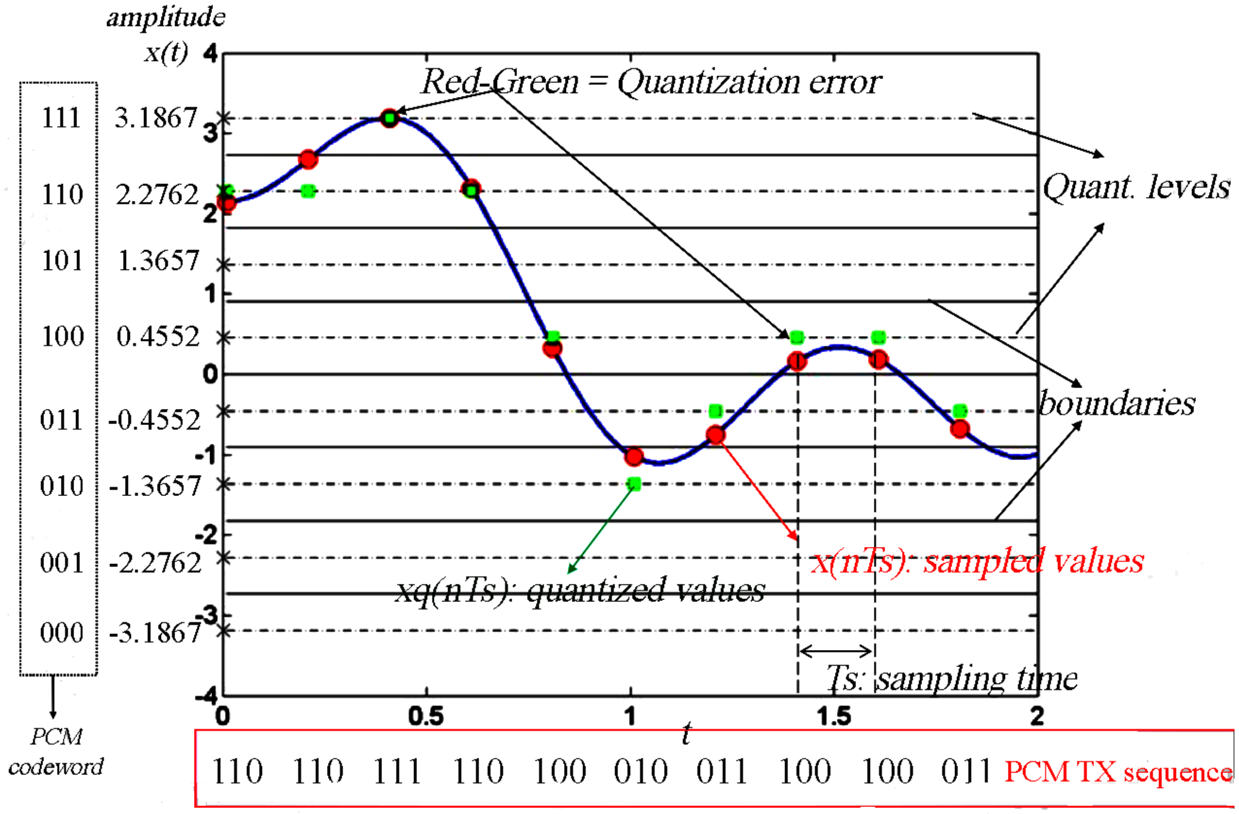
Synthetic Sound (23rd)

Quantization and transmission of audio (24th)

Quantization

- Since we quantize, we may choose to create either an accurate or less accurate representation of sound magnitude values.
- To compress the data, by assigning a bit stream that uses fewer bits for the most prevalent signal values.
- Quantization process introduces a certain amount of error or distortion into the signal samples.
- Perceptual Quantization (u-Law)
- Want intensity values logarithmically mapped over N quantization units





Quantization and transformation of data are collectively known as ***coding of the data***. For audio, the µ-law technique for companding audio signals is usually combined with a simple algorithm that exploits the temporal redundancy present in audio signals.

Example: CD audio, which uses 16-bit samples at a 44,100 Hz sampling rate. There are two parallel streams, one for each channel, to produce stereo. What is the transmission rate of CD-quality audio?

As long as you understand the terms involved, this is a straightforward math problem. For each of the two channels, there are 44,100 samples per second. Each of these samples requires 16 bits. Transmission rates are normally described in terms of the number of bits per second that must flow from the source to the destination. In our case:

44100 samples per second * 16 bits per sample * 2 channels = 1411.2 kbps

5-different Voice Digitization - techniques:

1) **PAM** = pulse amplitude modulation
2) **PDM** = pulse duration modulation
3) **PPM** = Pulse position modulation
4) **PCM** = Pulse code modulation
5) **ADPCM** = Adaptive differential PCM

In general, ***producing quantized sampled output for audio is called Pulse Code Modulation, or PCM.***

- PCM is an extension of PAM wherein each analog sample is quantized into a discrete value for representation as a digital code word.
- PAM system can be converted to PCM if we add ADC at the source and DAC at the destination.

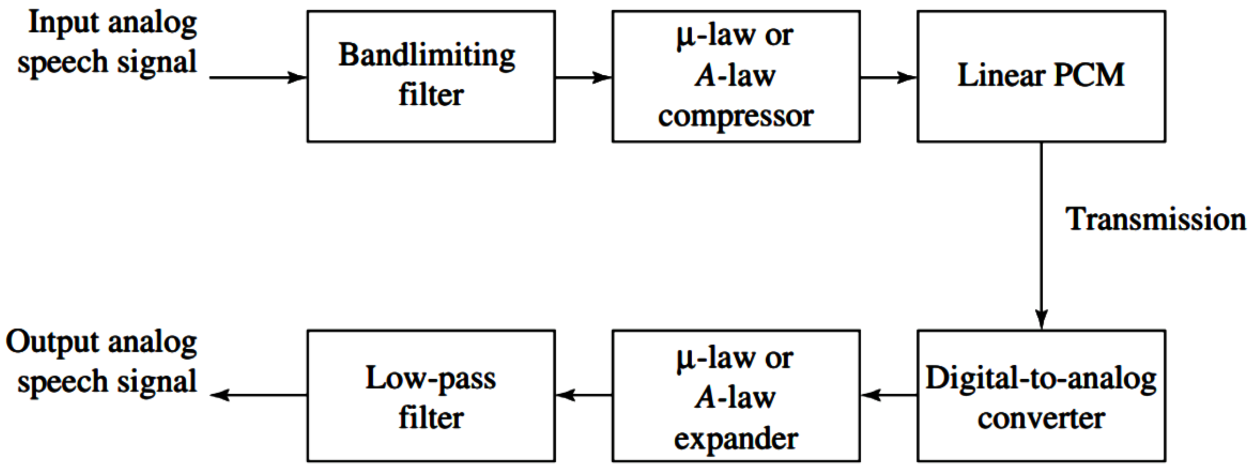


Figure: PCM signal encoding and decoding

Differential Pulse Code Modulation (DPCM)

What if we look at sample differences, not the samples themselves?

$$d_t = x_t - x_{t-1}$$

Differences tend to be smaller. Use 4 bits instead of 12 bits. Changes between adjacent samples small.

Value uses full bits, changes use fewer bits

Example:

(a) Full bits:-> 220, 218, 221, 219, 220, 221, 222, 218,... Result: originally for encoding sequence 0-255 numbers need 8 bits;

(b) Changes:-> (Difference sequence) 220, -2, +3, -2, +1, +1, +1, -4....

Difference coding: need only 3 bits

Compression of Audio (25th)

The Nyquist theorem states how frequently we must sample in time to be able to recover the original sound.

Data compression implies sending or storing a smaller number of bits. Although many methods are used for this purpose, in general these methods can be divided into two broad categories: lossless and Lossy methods. Audio compression can be used for speech or music. For speech, we need to compress a 64 kHz digitized signal, while for music we need to compress a 1.411 MHz signal. Two categories of techniques are used for audio compression: predictive encoding and perceptual encoding.

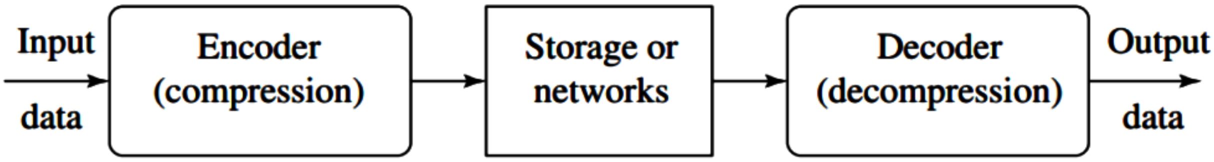


Figure: A general data compression scheme

lossy audio compression

Our eyes and ears cannot distinguish subtle changes. In such cases, we can use a lossy data compression method. These methods are cheaper, they take less time and space when it comes to sending millions of bits per second for images and video. Several methods have been developed using lossy compression techniques. JPEG (Joint Photographic Experts Group) encoding is used to compress pictures and graphics, MPEG (Moving Picture Experts Group) encoding is used to compress video, and MP3 (MPEG audio layer 3) for audio compression.

lossless audio compression

In lossless data compression, the integrity of the data is preserved. The original data and the data after compression and decompression are exactly the same because, in these methods, the compression and decompression algorithms are exact inverses of each other: no part of the data is lost in the process. Redundant data is removed in compression and added during decompression. Lossless compression methods are normally used when we cannot afford to lose any data.

1. μ-Law and A-Law Companding

These two method encode audio samples, by means of nonlinear quantization. The μ-law encoder inputs 14-bit samples and outputs 8-bit codewords. The A- law inputs 13-bit samples and outputs 8-bit codewords. G.711 standard is an 8-bit codewords whose format is shown :-

P	S2	S1	S0	Q3	Q2	Q1	Q0

- Bit P is sign bit of the output.
- Bits S2, S1, and S0 are the segment code.
- Bits Q3 through Q0 are the quantization code.

Algorithm (1): μ-Law encoder

Input : 14 bit samples

Output: 8 bit codewords

Begin:

step1: adding a bias 33to the absolute value of the input sample.

step2: determining the bit position of the most significant 1-bit among bits 5 through 12 of the input.

step3: subtracting 5 from that position.

Step4: The 4-bit quantization code is set to the four bits following the bit position determined in step 2.

Step5: The encoder ignores the remaining bits of the input sample, and it inverts codeword before it is output.

Algorithm (2): μ-Law decoder

Input : 8 bit codewords

Output: 14 bit samples

Begin:

Step1: Multiply the quantization code by 2 and add the bias 33 to result.

Step2: Multiply the result by 2 raised to the power of the segment code.

Step3: Decrement the result by the bias 33.

Step4: Use bit P to determine the sign of the result.

End

Example: input sample -656

P = 1   because sample is negative.

$|-656|+33 = 689$

$689 = 0001010110001_2$

				Q3	Q2	Q1	Q0					
0	0	0	1	0	1	0	1	1	0	0	0	1
12	11	10	9	8	7	6	5	4	3	2	1	0

The most significant 1-bit in positions 5 through 12 is found at position 9. The segment code value = 9-5 = 4.

The quantization code is the four bits 0101 at positions 8-5, and the remaining five bits 10001 are ignored. The 8-bit codeword (which is later inverted) becomes

P	S2	S1	S0	Q3	Q2	Q1	Q0
1	1	0	0	0	1	0	1

μ - Law decoder

1. The quantization code is $101_2 = 5$, so $5 \times 2 + 33 = 43$
2. The segment code is $100_2 = 4$,   so $43 \times 2^4 = 688$.
3. Decrement by the bias 688 — 33 = 655.
4. Bit P is 1, so the final result is —655. Thus, the quantization error (the noise) is 1; very small.