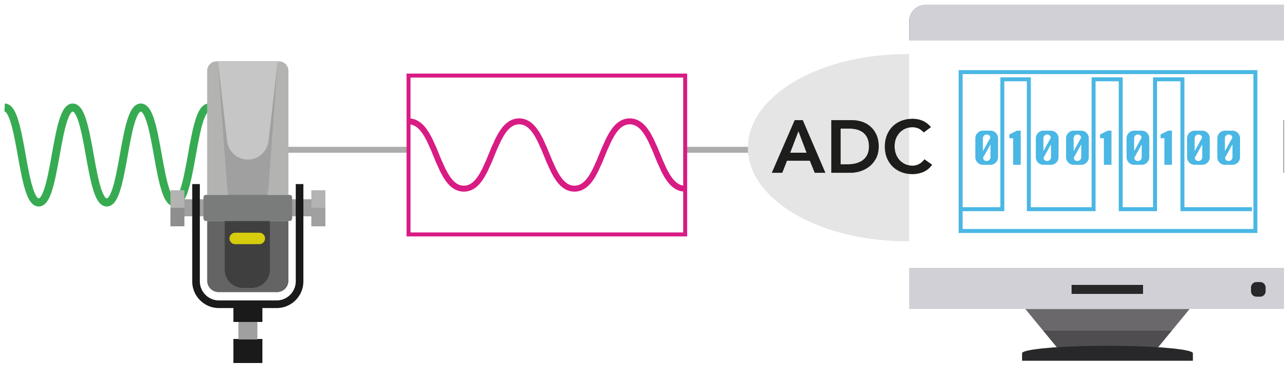
TOMMASO ROSATI
SOUND ART

SAMPLING AND QUANTIZATION

DIGITAL SOUND
SAMPLING
QUANTIZATION


ADC
0100101100
DAC

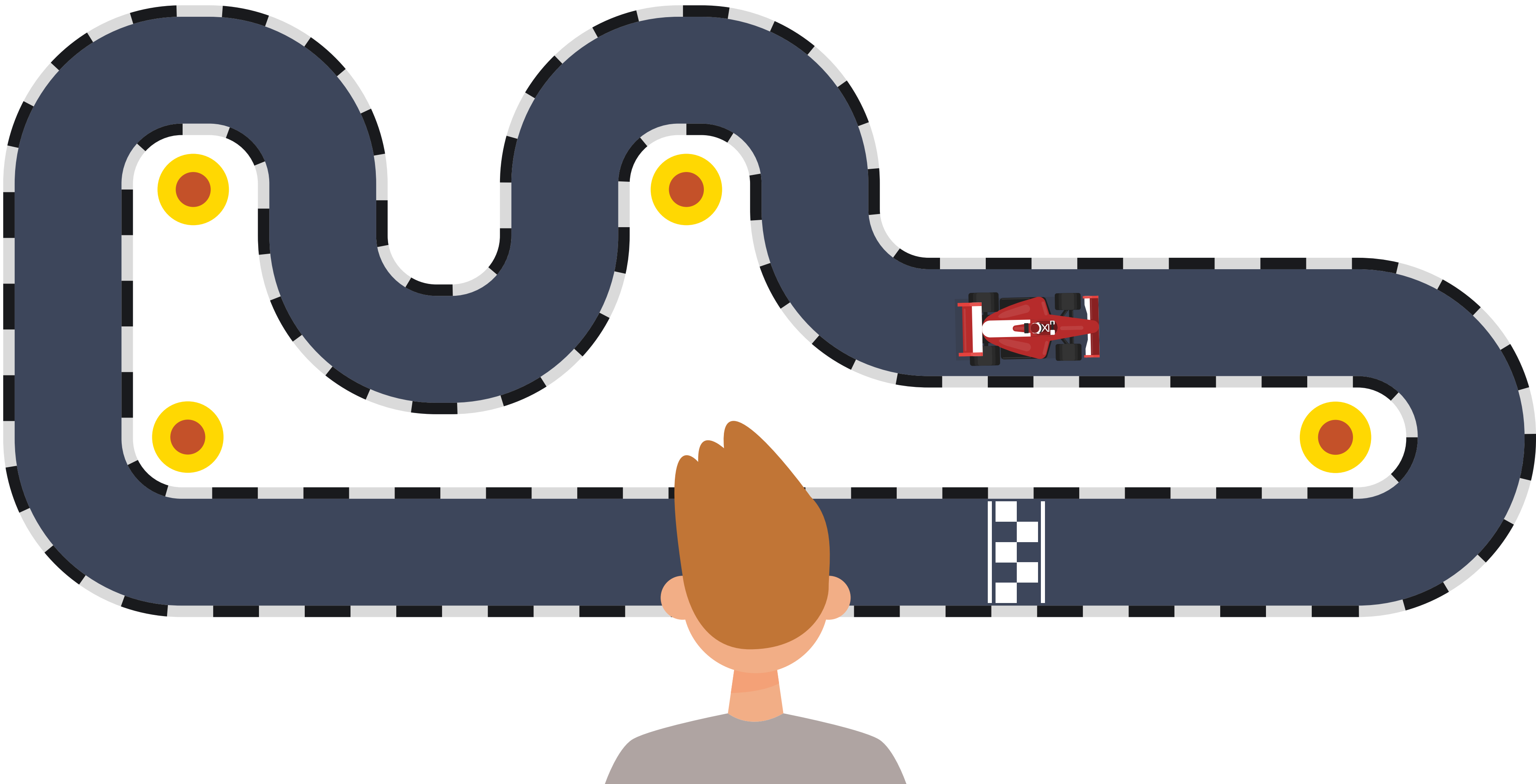
ADC



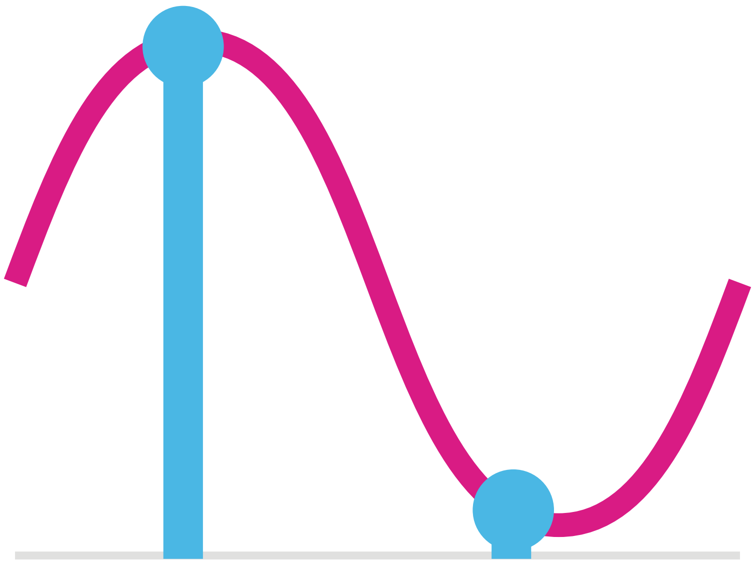
Every sound is made up of a sum of sinusoids: if I can represent the waveform resulting from the sum of these sinusoids, I can then reconstruct the original sound.

But at what **frequency** should I take my samples in order to then reconstruct my original signal without loss of information?

ADC



ADC



1

For each wave cycle, I will need at least two sample points, evenly spaced, to represent the wave in the digital domain.

ADC

2

I consider the worst case, which is the highest frequency we can hear: **20,000 Hz**, and I try to sample that.

To do this, since I need to take 2 points, I will have to sample at twice that frequency: **40,000 Hz**.

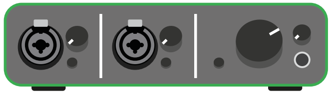
twice the bandwidth of frequencies that I want to represent

$$f_s > 2B$$

sampling frequency

Nyquist-Shannon sampling theorem

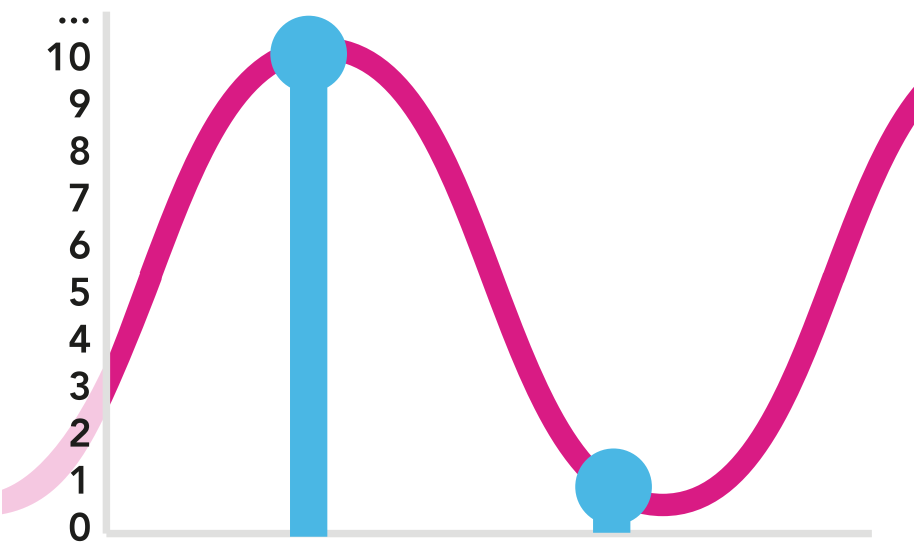
ADC

	CD	44100 Hz	16 bit
	VIDEO	48000 Hz	16/24 bit
	AUDIO INTERFACE	96000 Hz	24 bit

ADC

I found the sampling frequency, but what values can I give to my samples?

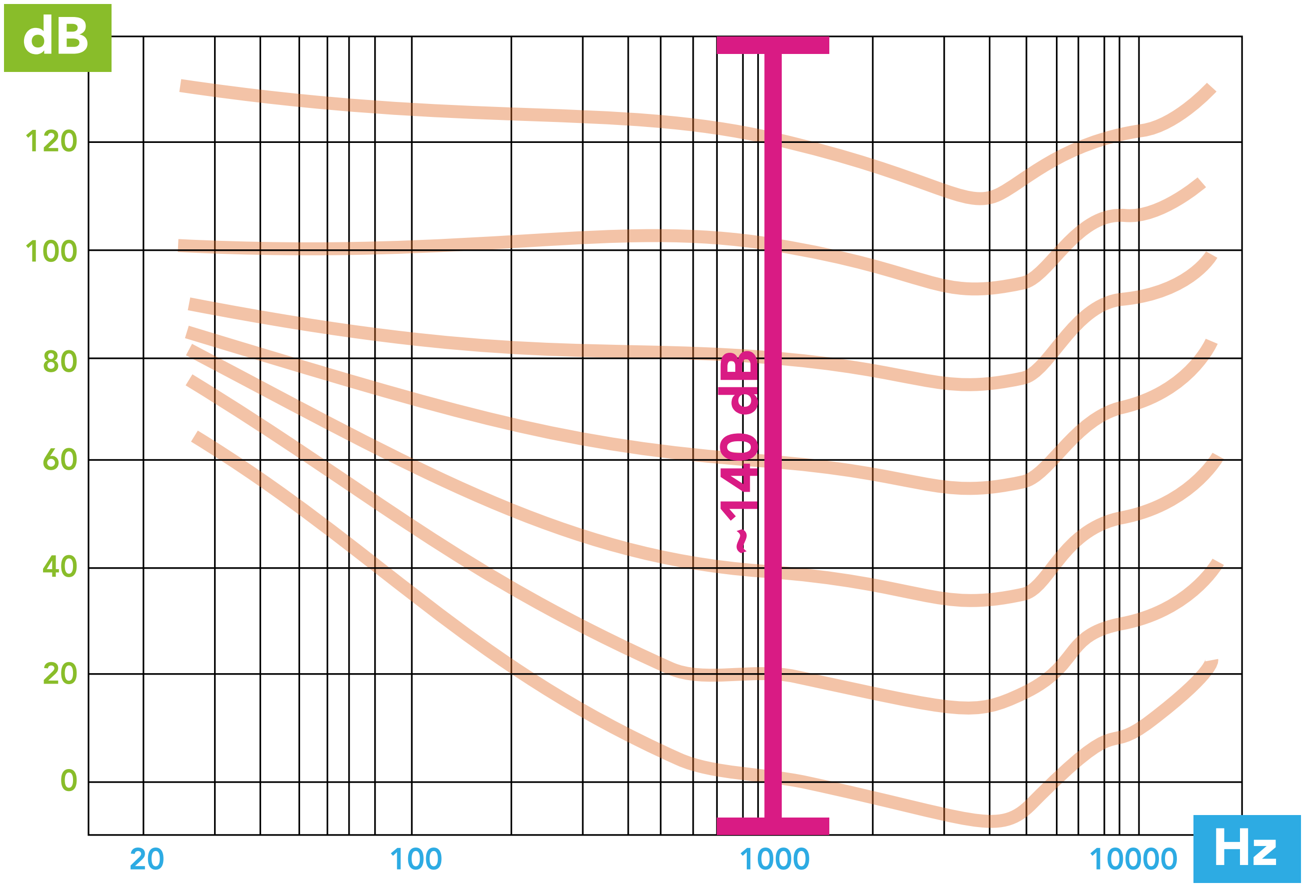
What is the minimum number of values, the level of precision, that I need to be able to accurately reconstruct my original sound?



ADC

1

The maximum range that our auditory system can perceive is 140 dB relative to a reference of 0, which is silence.



ADC

2

In digital, data is stored in bits, which are memory cells that can assume a state of **0** or **1**.

I therefore need to find the number of bits that allows me to represent 100-140 dB.

A certain number of bits has a precise number of possible states, given by the calculation:

$$2^N$$

possible states of 1 bit (0 or 1)

N number of bits

ADC

3

In dB, doubling the amplitude is calculated by adding 6 dB because dB is a logarithmic scale.

In 1 bit, there are two possible states: 0 or 1. If I consider 0 as silence and 1 as sound, I can say that 1 is double the amplitude of my silence (6 dB).

And if I now want to double the number of possible states, I need to increase by one bit.

With 2 bits, I have 4 possible states, so I can represent 12 dB (double the amplitude of 6 dB)

With 3 bits, I have 8 states and represent double 12 dB, which is 18 dB

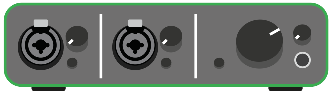
etc.

ADC

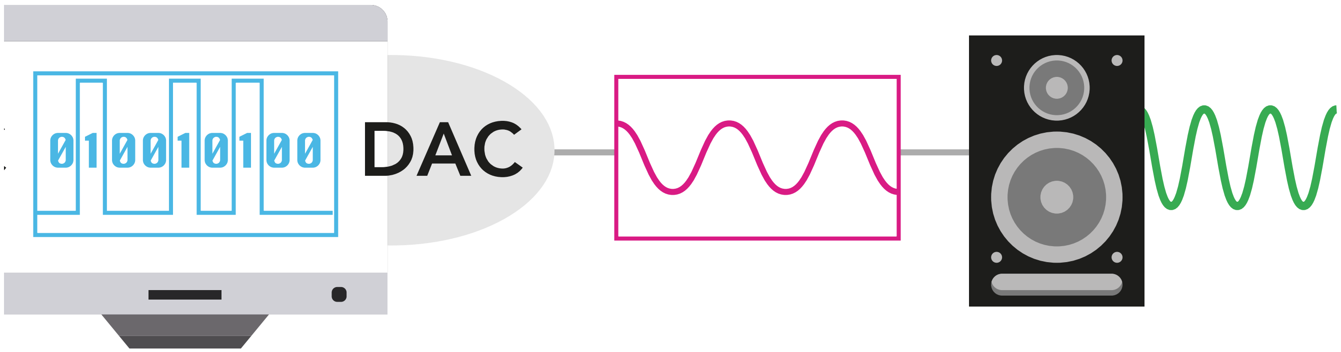
+1 bit = a gain of 6dB in signal-to-noise ratio

bit	states	signal-to-noise ratio (dB)
1	2	6
2	4	12
3	8	18
4	16	24
5	32	30
8	256	48
16	65,536	96
24	16,777,216	144

DAC

	CD	44100 Hz	16 bit
	VIDEO	48000 Hz	16/24 bit
	AUDIO INTERFACE	96000 Hz	24 bit

DAC

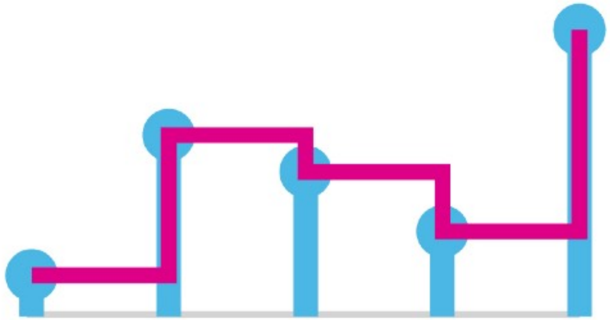


In the phase of reconstructing the analog signal, I have a series of samples and need to construct what is between one sample and the next.

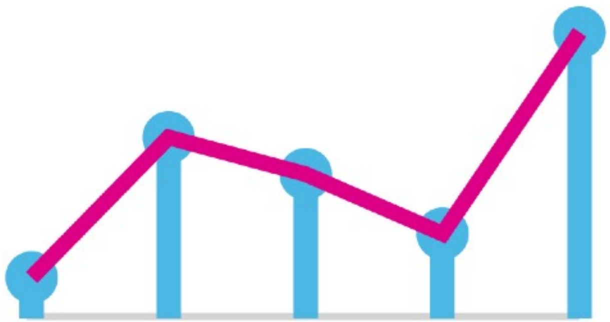
What **function** creates an interpolation between the samples that faithfully brings me back to the original sound?

DAC

In history, various tests have been conducted before arriving at the most accurate function:

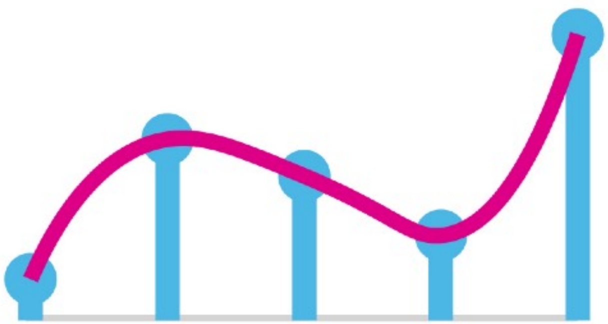


Sample and Hold
takes the value of a sample and holds that value until we get to the next sample. However, these abrupt and sharp "steps" can create unwanted harsh timbres.



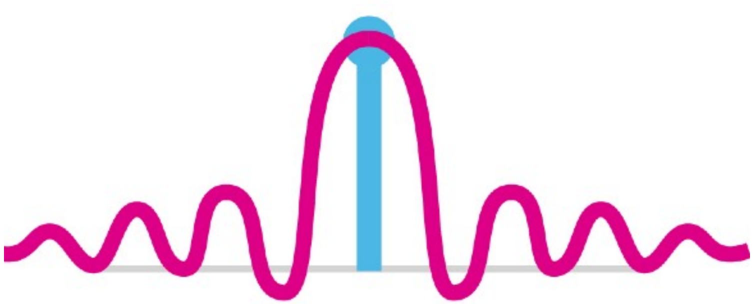
Linear interpolation
directly connects one sample's value to the next sample's value using a straight line with a constant slope. This method also creates sharp angles that can result in a harsh sound.

DAC



Spline cubiche

improves the result but is still not "perfect."

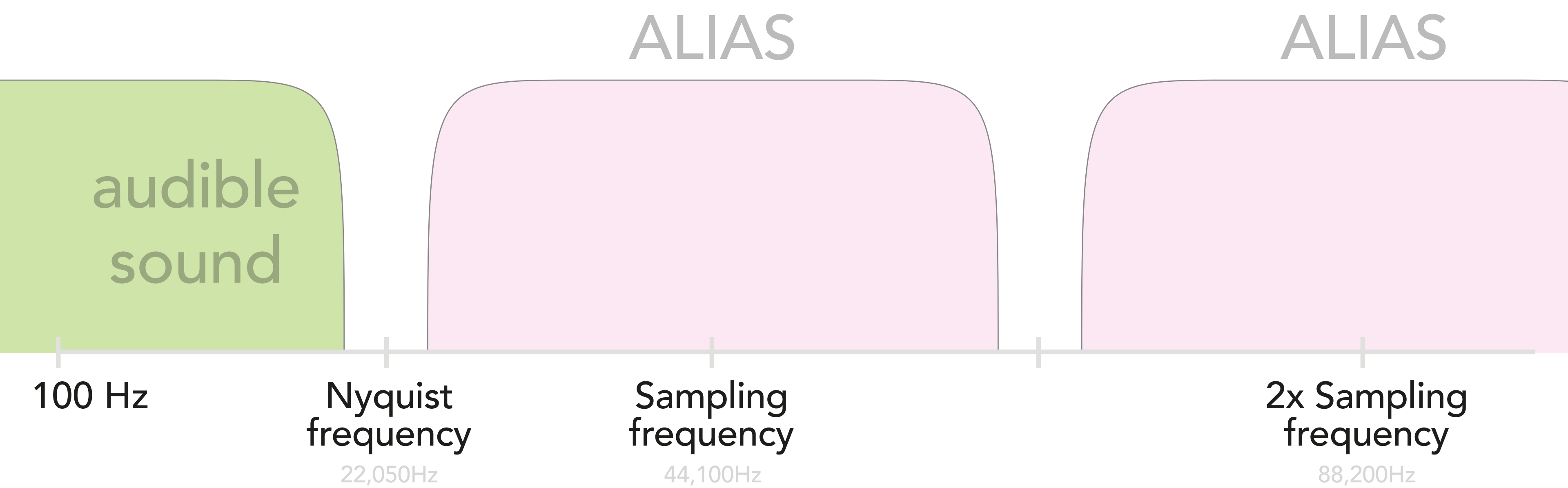


Sinc or Cardinal Sine

This function reconstructs an analog signal that is virtually identical to the original wave.

ADC-DAC

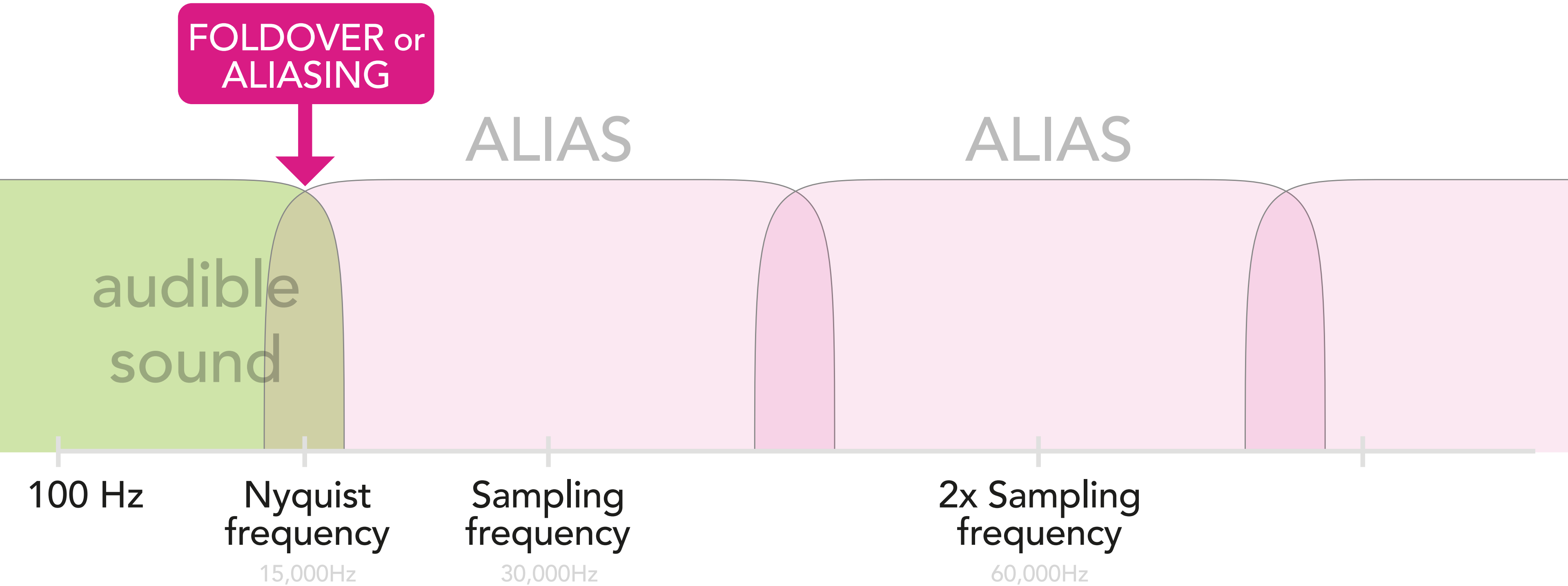
When I sample at a certain sampling frequency, ambiguities are created with the multiples of the sampling frequency.
These ambiguities are called **ALIAS**.



If I sample at a very high sampling frequency, the phenomenon of aliasing does not create major difficulties unless I introduce frequencies much higher than the Nyquist frequency.

ADC-DAC

When I introduce frequencies higher than the Nyquist frequency, these frequencies undergo a ‘**foldover**’ effect, also known as **aliasing**, due to the overlap between the positive part of the real frequency and the negative mirror image of the alias frequency.

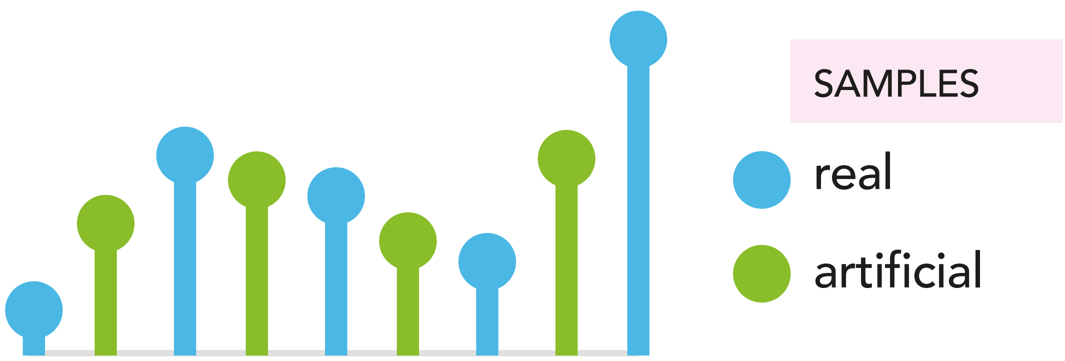


ADC-DAC

There are some methods during sampling or reconstruction that can help us avoid aliasing distortion:
- using **anti-aliasing filters**
    - at the input to prevent the contribution of frequencies that are too high
    - at the output, to attenuate any aliases occurring at very high frequencies
- increasing the **sampling frequency**
- Using **Oversampling**

During reconstruction, adding artificial samples between real ones virtually doubles the sampling rate, helping to avoid foldover distortion.

Oversampling



www.tommasorosati.it