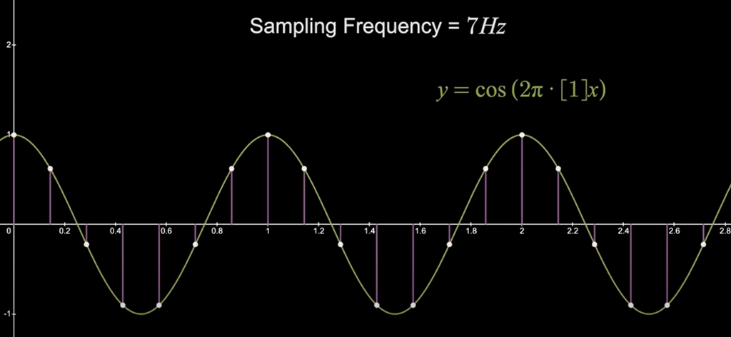
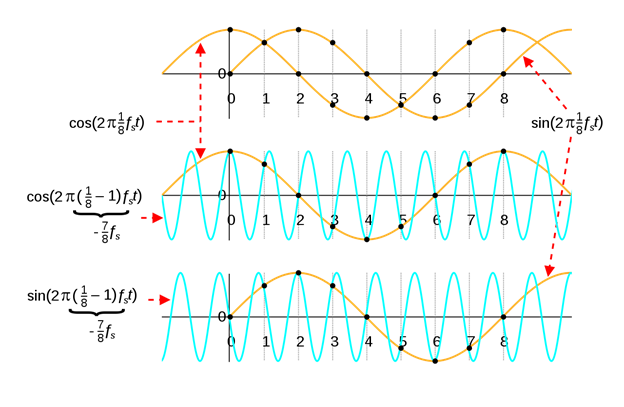
NOTES ON THE COMMUNICATION PROJECT



# Sampling rate:

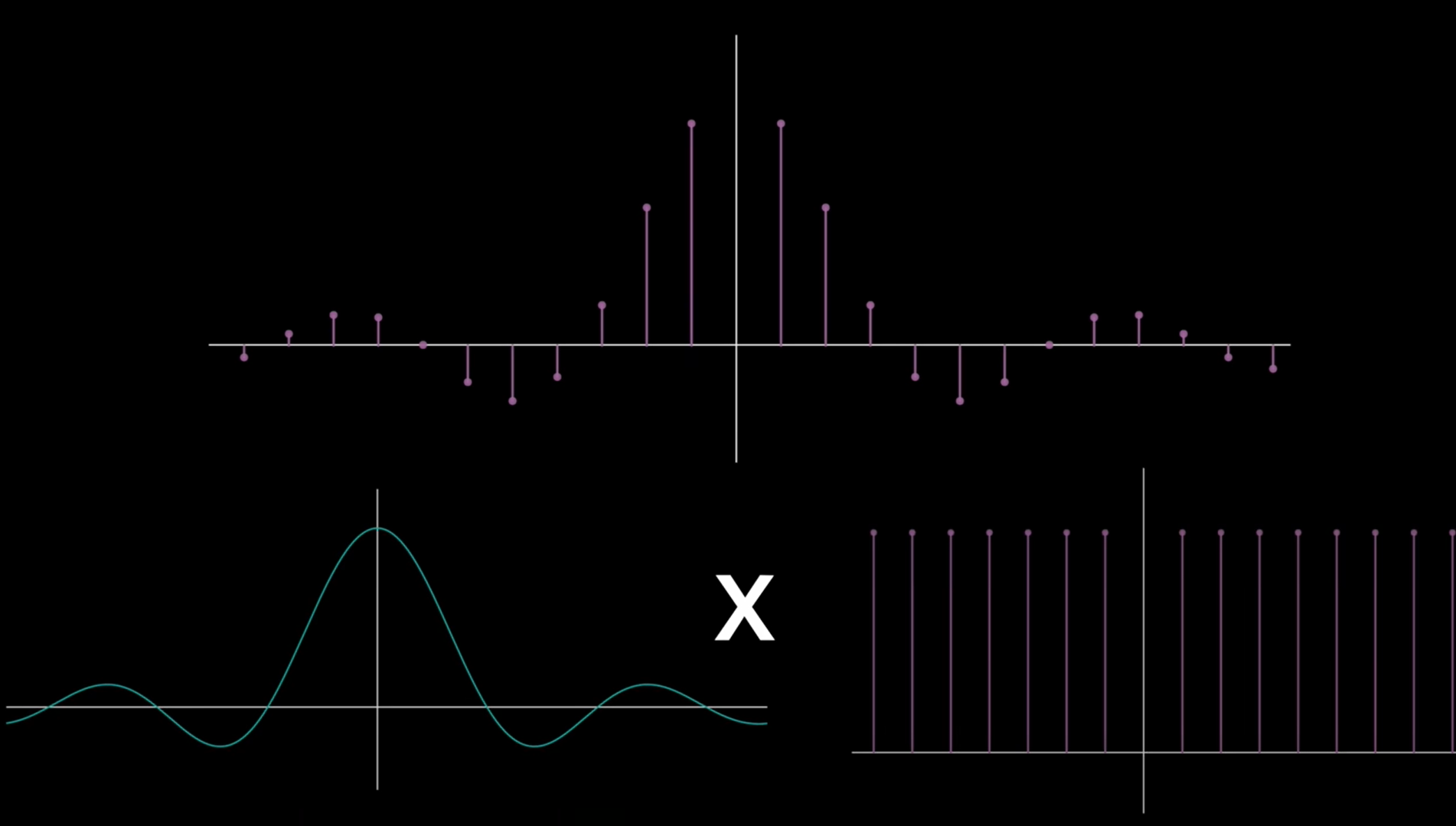
It means how many samples are taken in one second.

# Nyquist-Shannon sampling theorem:

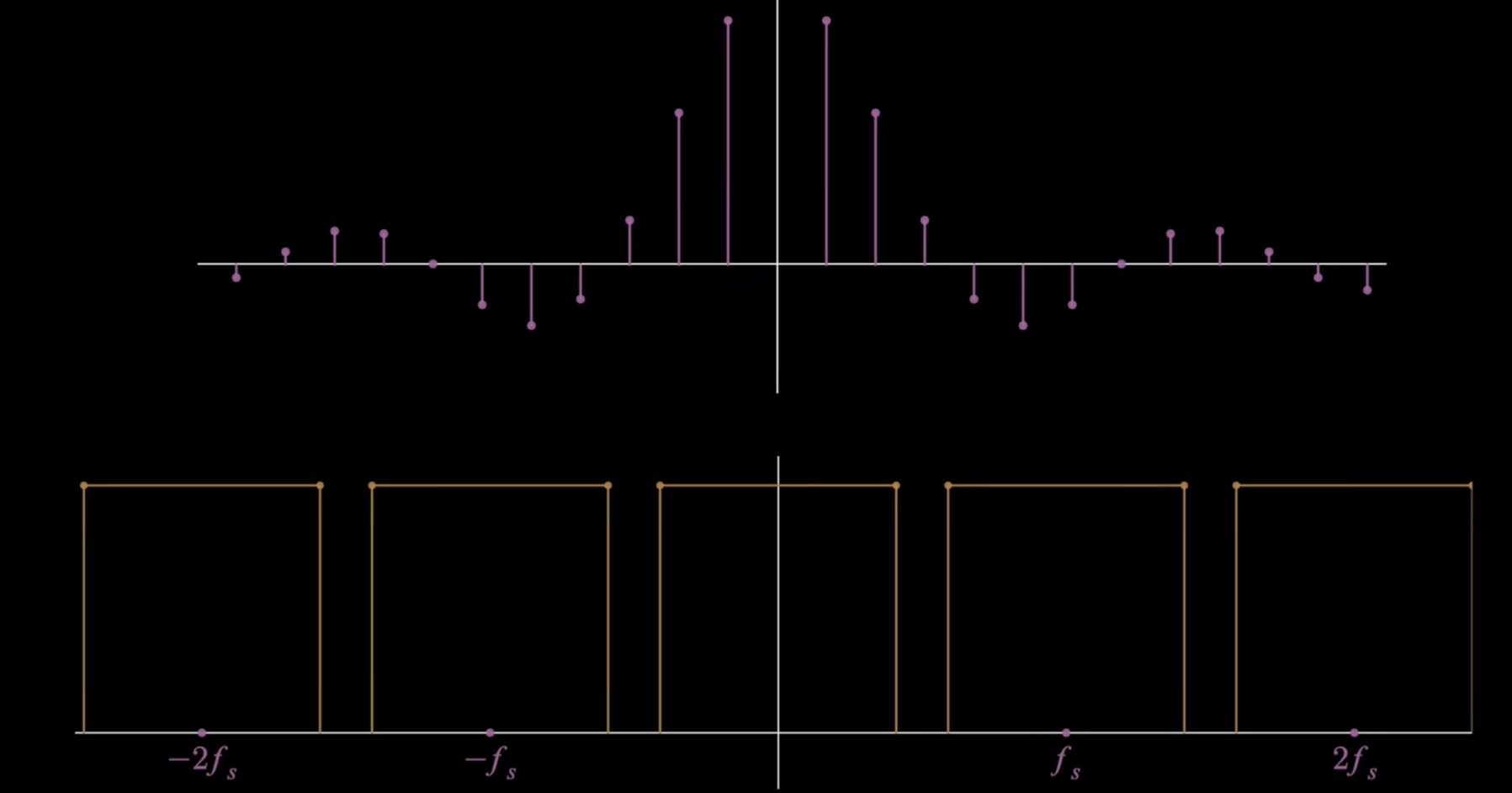
1. If you sample with any frequency, then an infinite number of frequencies can have the same value at the points of sampling.
2. consider it to be the original signal that can be correct. The problem is that doesn't happen unless the sampling frequency is two times bigger or more than the original
3. Aliasing means that the information is lost when trying to get the original signal from samples and ed up with something else.

To sum up, fS>2fmax sampling frequency must be bigger than 2 times the maximum frequency in a signal.

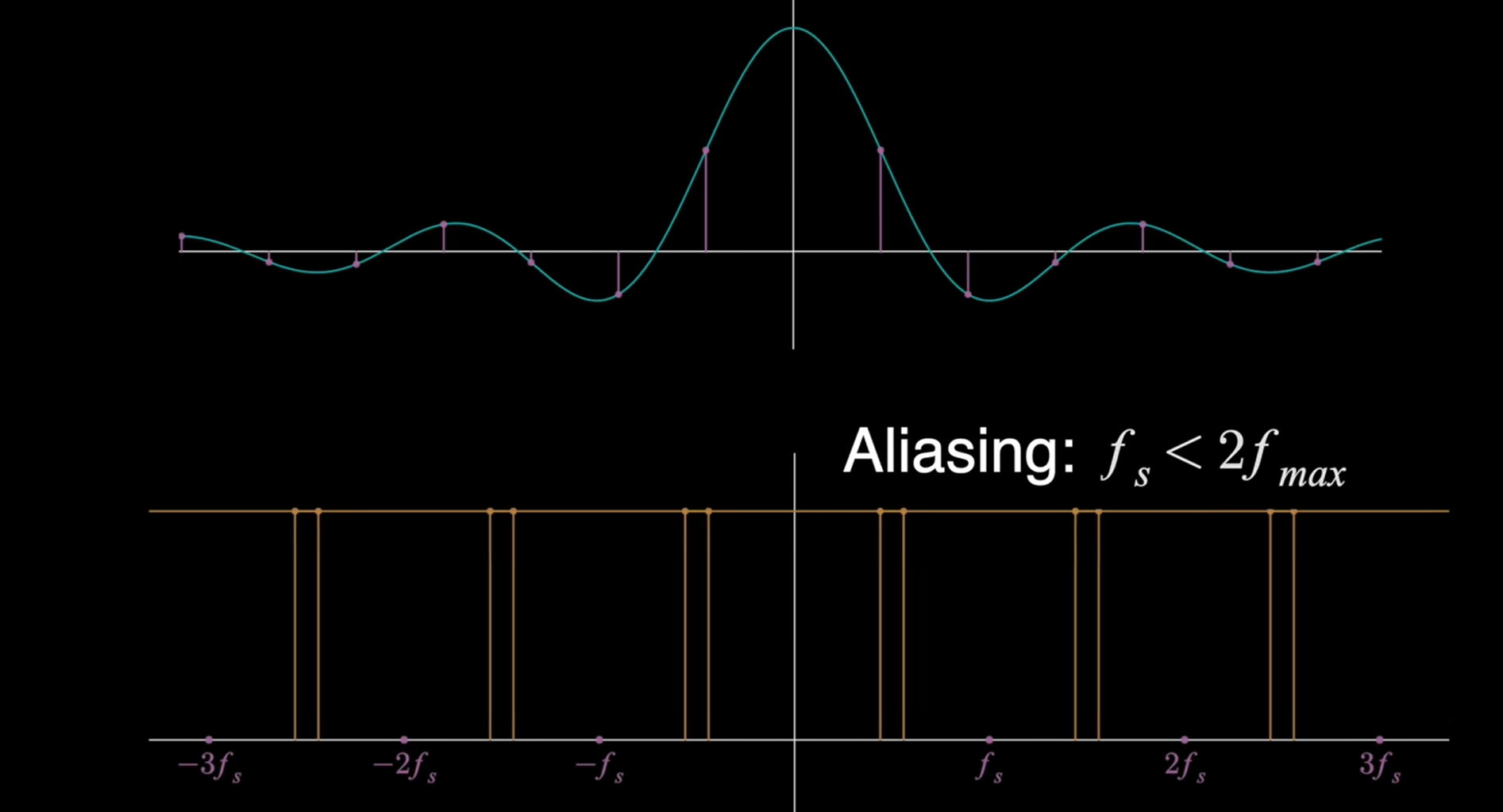
Nyquist theorem is the reason why the standard sampling rate in recording audio is 44.1khz which is bigger than double the maximum hearing frequency(20khz).

The sampled signal looks like multiple delta functions, so to sample a function you can simply multiply the signal with the delta function (which is convolution in the frequency domain). 

Imagine you have a sinc function()whose Fourier transform is a rectangle function, then the Fourier transform is a rec function, so by sampling then we multiply sinc times multiple delta functions so(sinc(t).), then in Fourier transform it is (rec(t) convolution with 1), so we will end up with multiple square function for each delta function.



As you can see the bandwidth of the or smaller than sampling frequency (fs), In fact the width of a full rectangle is equal to times of the original bandwidth so, if the sampling frequency were smaller than that it would look like this.



An overlap happened and will cause information loss.

# Recording notes:

1. In audio, "16-bit" refers to the bit depth, which is how many bits used to define the amplitude, so more bit means more accuracy. That is why in matlab the audio samples are in datatype double, and that the two bytes(in double) represent values from -1 to +1.
   1. Clipping occurs when an audio signal exceeds the maximum amplitude that can be represented or output, causing the waveform to be cut off (flattened) at the top or bottom. This results in distortion and a harsh, buzzy sound.
2. A mono channel (or monophonic audio) means the audio has only one channel — it’s recorded and played back as a single audio stream.
3. In matlab the way the audio is stored by using an object name "audiorecorder" that object contains the samples and other properties of the audio, and you can't access the samples due to how matlab works (you don't have to know why?).

# Filter order:

Higher filter order means sharper filtering and steeper transition band, and the downsides are more complexity and stability issues and greater phase distortion.  
  
  
General notes:

* When you play any sound and the sound is sharp, like the sound of some metal classing slowly, then use a lowpass filter.  
    
  sometimes the solution you are looking for is just increasing the sampling frequency. So just use a massive sampling frequency at the start like 4 times the maximum frequency of the message.
* In the fft() function and fftshift(), the fftshift() does not generate Fourier transform and then shift it no, what is it does is it takes a Fourier transform signal as a parameter then shift it. This is even seen in the document page about modulate function.  
  <https://www.mathworks.com/help/matlab/ref/fftshift.html>
* Some of the conditions we are working in are considered to be ideal, like the oscillator at the receiver is assumed to have the exact same frequency as the carrier frequency and having no phase shift between them.
* In SSB the message should not have a big power density around zero frequency meaning , in other words the message should not have a big DC component, but the filters in MATLAB are ideal that is why SSB can modulation and demodulation can be simulated on any message.