Sampling & Quantization

Before we can study the digital communication techniques, we need to study how to convert analog signals to digital signals.

The first step in this process is called SAMPLING.

In sampling, we convert continuous–time analog signals (signals that are defined at all time instants and have amplitudes that may take any real value) to discrete–time analog signals (signals that are defined at specific instants of time but still have amplitudes that may take any real value).

Sampling

A continous–time analog or digital signal is defined at all time instants.

On the other hand, a discrete–time analog or digital signal is defined only at some time instants.

Let the signal $g(t)$ be a continuous–time signal with bandwidth $2\pi B$ rad/s (B Hz)

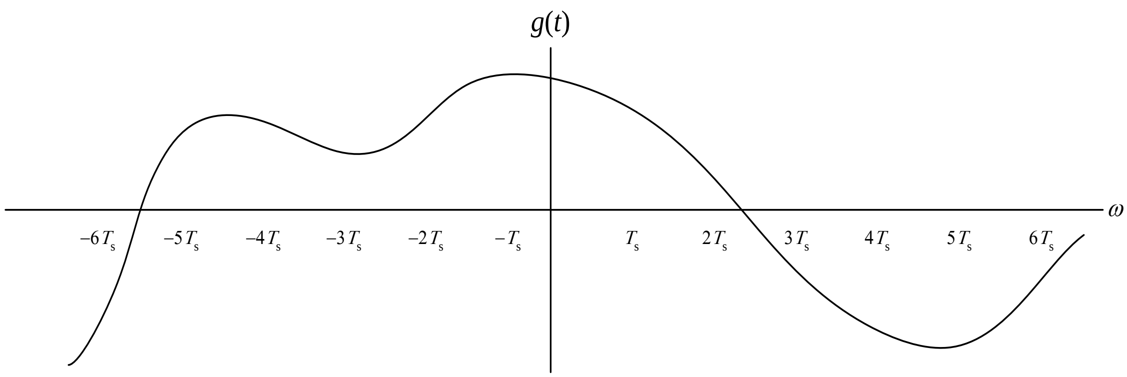
To sample $g(t)$, we multiply it by a train of delta functions that occur every T_s seconds

$$\delta_{T_s}(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT_s).$$

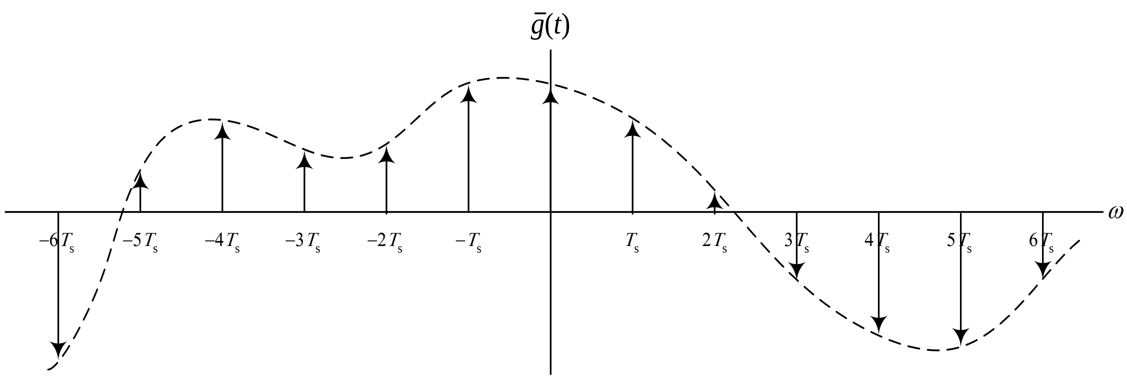
Therefore, the sampled signal $\overline{g}(t)$ is given by

$$\begin{aligned}\overline{g}(t) &= g(t)\sum_{n=-\infty}^{\infty} \delta(t - nT_s) \\ &= \sum_{n=-\infty}^{\infty} g(t)\cdot\delta(t - nT_s) \\ &= \sum_{n=-\infty}^{\infty} g(nT_s)\cdot\delta(t - nT_s).\end{aligned}$$

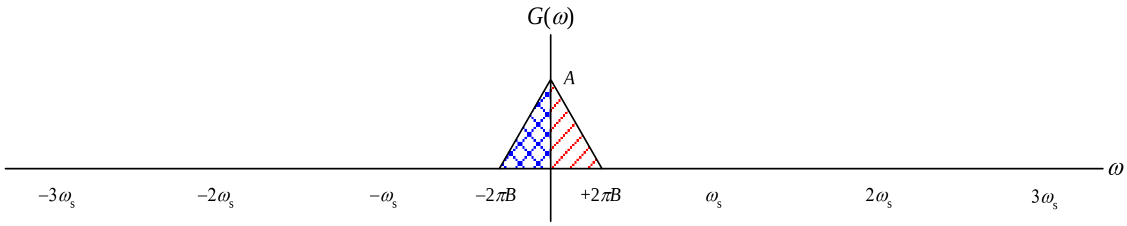
So, the sampled signal is a sum of delta functions that have magnitudes equal to the value of $g(t)$ at the time instants that the delta functions occur. The following figure shows a signal $g(t)$.



and the following figure show the sampled signal $\bar{g}(t)$ where the amplitude of the deltas follows the original signal $g(t)$.



Assume that the spectrum of $g(t)$ is given by $G(\omega)$ shown below.



We can get the spectrum of $\bar{g}(t)$ by find the spectrum of the train of delta functions and convolving it with $G(\omega)$, or by decomposing the train function into sine and cosine functions and then taking the Fourier transform of each element independently. Since the train of delta functions $\delta_{T_s}(t)$ is periodic, we can decompose it using the Fourier series as

$$\begin{aligned}\delta_{T_s}(t) &= \sum_{n=-\infty}^{\infty} \delta(t - nT_s) \\ &= \frac{1}{T_s} + \frac{2}{T_s}\cos(\omega_s t) + \frac{2}{T_s}\cos(2\omega_s t) + \frac{2}{T_s}\cos(3\omega_s t) + \ldots\end{aligned}$$

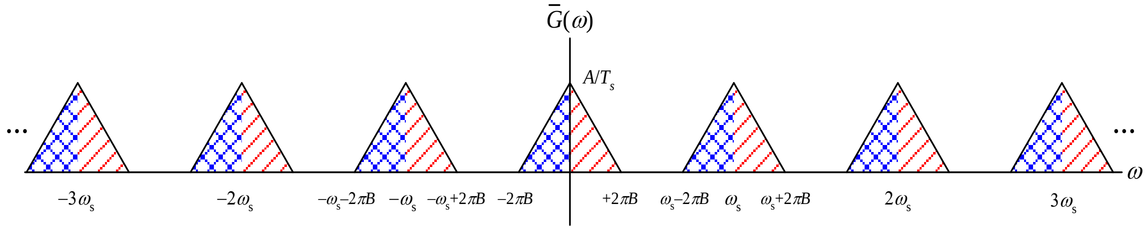
and

$$\begin{aligned}\bar{g}(t) &= g(t)\sum_{n=-\infty}^{\infty} \delta(t - nT_s) \\ &= \frac{1}{T_s}g(t) + \frac{2}{T_s}g(t)\cos(\omega_s t) + \frac{2}{T_s}g(t)\cos(2\omega_s t) + \frac{2}{T_s}g(t)\cos(3\omega_s t) + \ldots.\end{aligned}$$

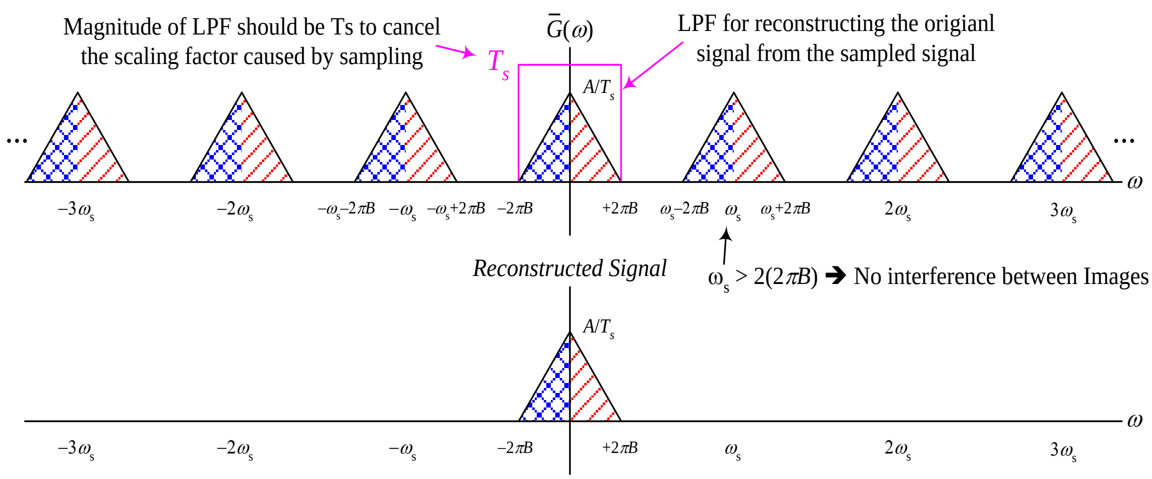
So, by taking the Fourier transform of each term of the above independently, we see that the spectrum $\bar{G}(\omega)$ is given by

$$\begin{aligned}\bar{G}(\omega) &= \frac{1}{T_s}G(\omega) + \frac{1}{T_s}\left[G(\omega - \omega_s) + G(\omega + \omega_s)\right] + \frac{1}{T_s}\left[G(\omega - 2\omega_s) + G(\omega + 2\omega_s)\right] \\ &\quad + \frac{1}{T_s}\left[G(\omega - 3\omega_s) + G(\omega + 3\omega_s)\right] + \ldots. \\ &= \frac{1}{T_s}\sum_{n=-\infty}^{\infty} G(\omega - n\omega_s)\end{aligned}$$

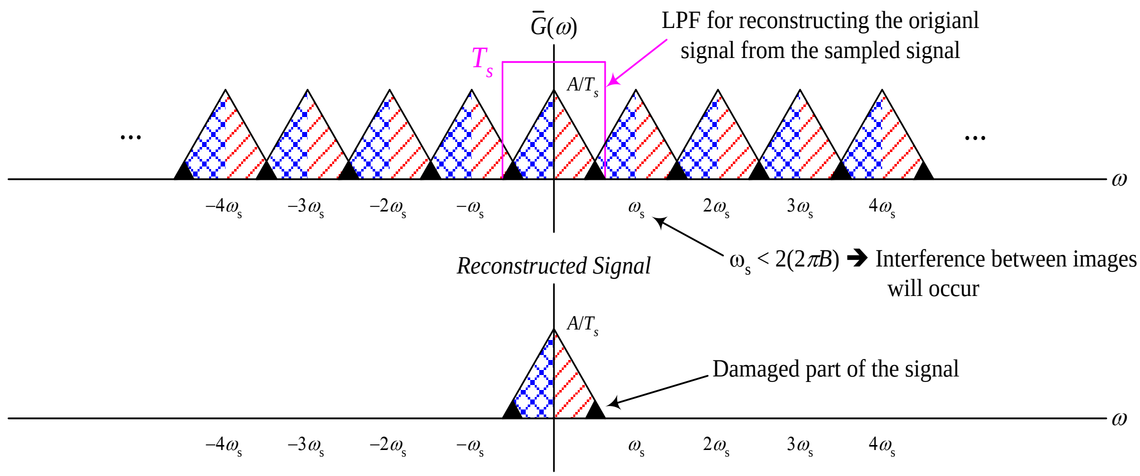
Therefore, the spectrum of the sampled signal would be



To extract the original signal from the sampled signal, it is clear that using a LPF with bandwidth equal to the bandwidth of the original signal $g(t)$ (which is $2\pi B$ rad/s in this case) will do the job. However, this is true only if the signal was sampled at a sampling rate that is greater than twice the bandwidth of the signal.

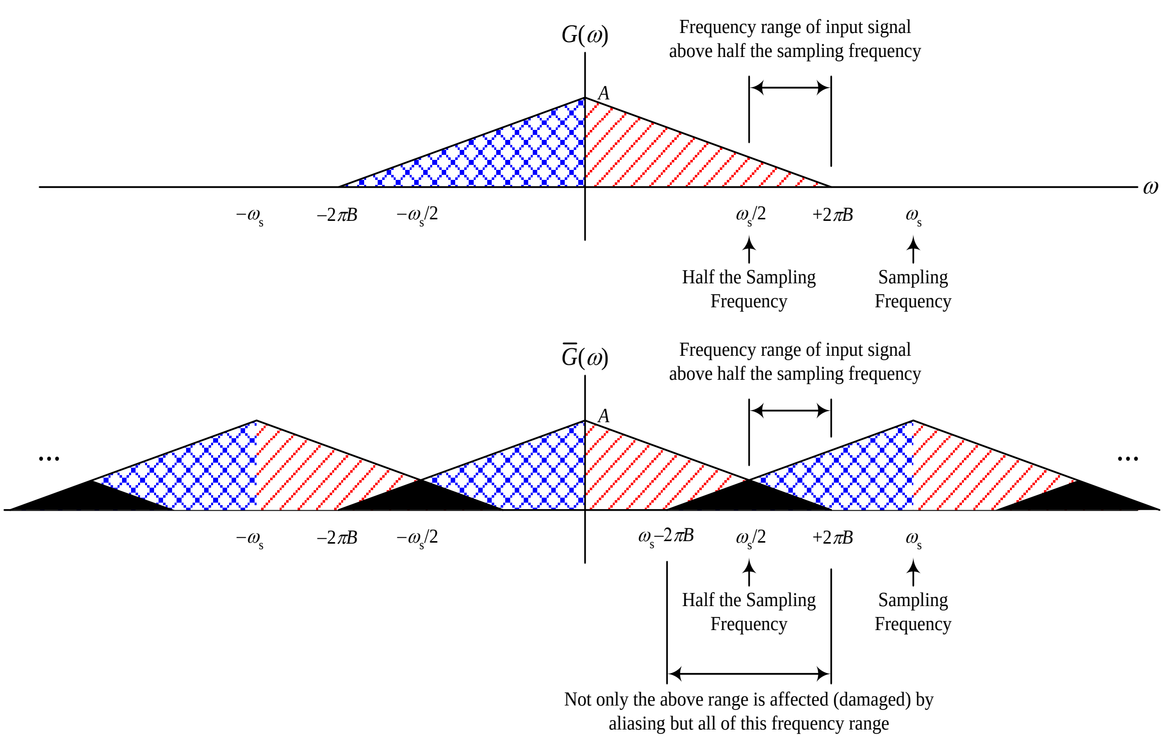


If the signal was sampled at a sampling rate lower that 2 times the bandwidth of the signal (called the **NYQUIST SAMPLING RATE**), the different spectral components of the sampled signal (called IMAGES) will interfere with each other and reconstructing the original signal will be impossible. This is illustrated in the following figure. The dark parts in the figure represent parts of the sampled signal and reconstructed signal that have been damaged.



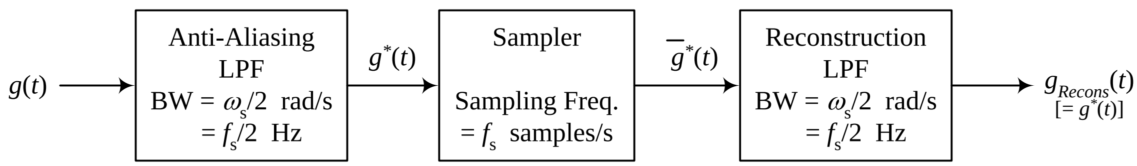
Anti–Aliasing Filters

So now we know that when the whenever the bandwidth of the input signal to a sampler is greater than half the sampling frequency (in other words, the sampling frequency is less than twice the bandwidth of the input signal), aliasing will occur. Unfortunately, aliasing does not only destroy the part of the input signal that has frequency greater than half the sampling frequency, but also an equal part of input signal that is below half the sampling frequency. This is illustrated in the figure below.

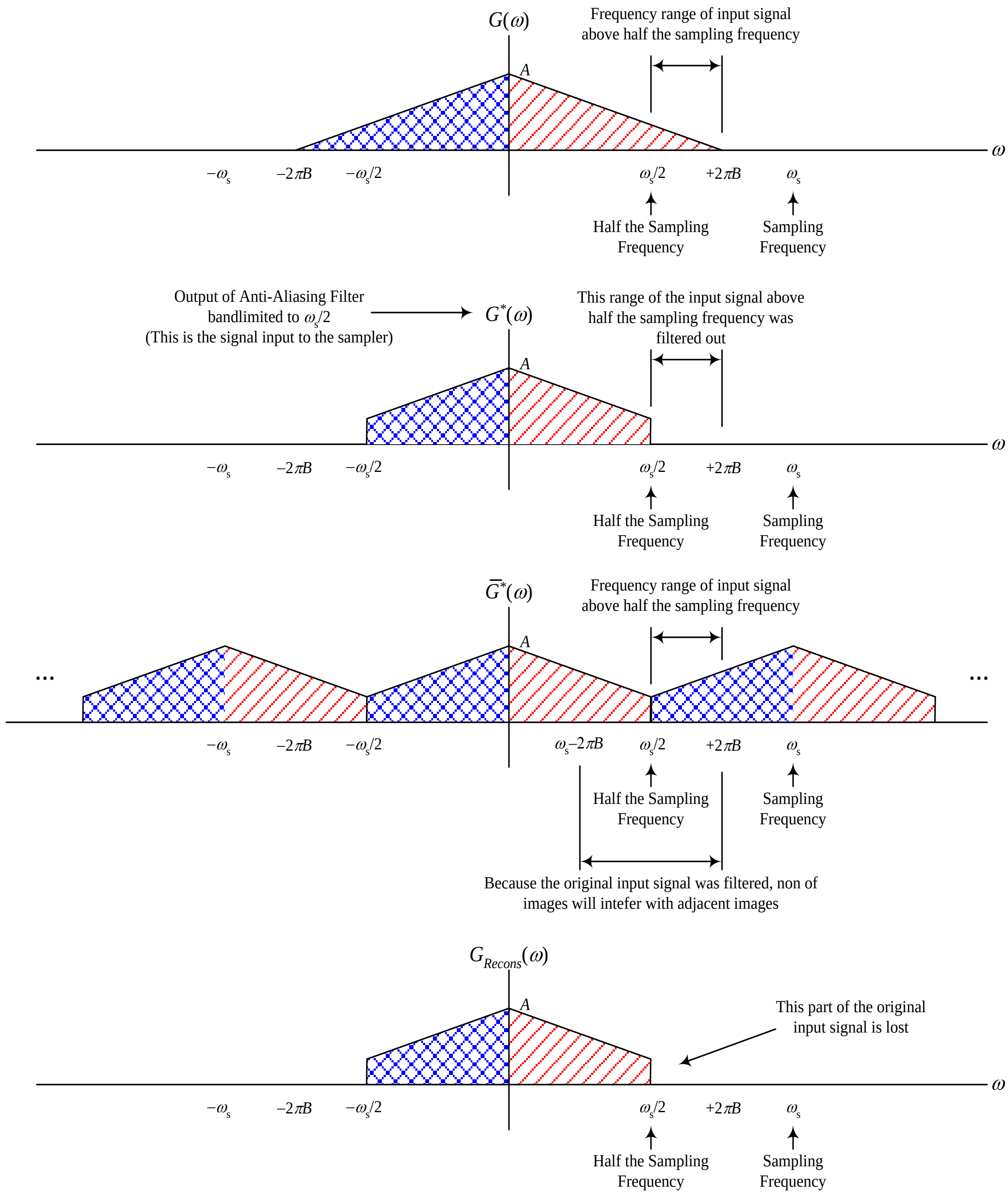


So, it is clear that not only the range of the input signal [$\omega_s/2$, $2\pi B$] gets affected by aliasing, but all the range from [$\omega_s - 2\pi B$, $2\pi B$] is affected by aliasing.

To SAVE HALF of the signal in the frequency range [$\omega_s - 2\pi B$, $2\pi B$], we can pass the input signal before sampling into a LPF that will cut all the part that is above $\omega_s/2$ so that the input signal to the sampling device has a bandwidth of exactly $\omega_s/2$. This means that a LPF with bandwidth $\omega_s/2$ called ANTI–ALIASING filter must be used. If the input signal to the sampler (which was produced by the anti–aliasing filter) has exactly half the sampling frequency, there will be no aliasing at all (but we will require an ideal LPF with bandwidth $\omega_s/2$ to reconstruct the continuous–time signal from the samples). Notice that the original input signal cannot be reconstructed back exactly because we removed part of it to avoid aliasing. Therefore, the block diagram of a practical sampling system is shown below.



The signals in the above block diagram will be as follows.



G(ω)
Frequency range of input signal
above half the sampling frequency
A
ω
−ωs
−2πB
−ωs/2
ωs/2
+2πB
ωs
Half the Sampling
Frequency
Sampling
Frequency
Output of Anti-Aliasing Filter
bandlimited to ωs/2
(This is the signal input to the sampler)
G*(ω)
This range of the input signal above
half the sampling frequency was
filtered out
A
ω
−ωs
−2πB
−ωs/2
ωs/2
+2πB
ωs
Half the Sampling
Frequency
Sampling
Frequency
Ḡ*(ω)
Frequency range of input signal
above half the sampling frequency
A
...
...
−ωs
−2πB
−ωs/2
ωs−2πB
ωs/2
+2πB
ωs
Half the Sampling
Frequency
Sampling
Frequency
Because the original input signal was filtered, non of
images will intefer with adjacent images
GRecons(ω)
A
This part of the original
input signal is lost
ω
−ωs
−2πB
−ωs/2
ωs/2
+2πB
ωs
Half the Sampling
Frequency
Sampling
Frequency

Examples of Aliasing in Real Life

There are many real–life phenomena that result from aliasing, but many do not know that these are actually caused by aliasing. Here we will give two examples.

1. When video taping a TV or a PC monitor, sometimes wide black lines appear moving at some constant speed from top to bottom or vice versa across the screen. This results because of the difference in sampling (number of pictures per second) of the video camera and the number of frames the TV or PC monitor display per second.
2. When looking at something that rotates at high speed (such as a fan or a car's wheel), you sometimes see that it is rotating in the opposite direction. This also happens because the human eye works like a video camera where it also takes pictures at a rate close to 24 pictures per second. If the rotating object rotates at a high speed that by the time the eye takes the next picture that object has revolved slightly less than one rotations, this object will appear as if it is rotating in the opposite direction.

Pulse Code Modulation (PCM)

The modulation methods PAM, PWM, and PPM discussed in the previous lecture still represent analog communication signals since the height, width, and position of the PAM, PWM, and PPM, respectively, can take any value in a range of values. Digital communication systems require the transmission of a digital for of the samples of the information signal. Therefore, a device that converts the analog samples of the message signal to digital form would be required. Analog to Digital Converters (ADC) are such devices. ADCs sample the input signal and then apply a process called quantization. The quantized forms of the samples are then converted to binary digits and are outputted in the form of 1's and 0's. The sequence of 1's and 0's outputted by the ADC is called a PCM signal (Pulses have been coded to 1's and 0's).

Example: A color scanner is scanning a picture of height 11 inches and width 8.5 inches (Letter size paper). The resolution of the scanner is 600 dots per inch (dpi) in each dimension and the picture will be quantized using 256 levels per each color. Find the time it would require to transmit this picture using a modem of speed 56 k bits per second (kbps).

We need to find the total number of bits that will represent the picture. We know that 256 quantization levels require 8 bits to represent each quantization level.

Number of bits = 11 inches (height) * 8.5 inches (width) * 600 dots / inch (height)
* 600 dots / inch (width) * 3 colors (red, green, blue)
* 8 bits / color = 807,840,000 bits

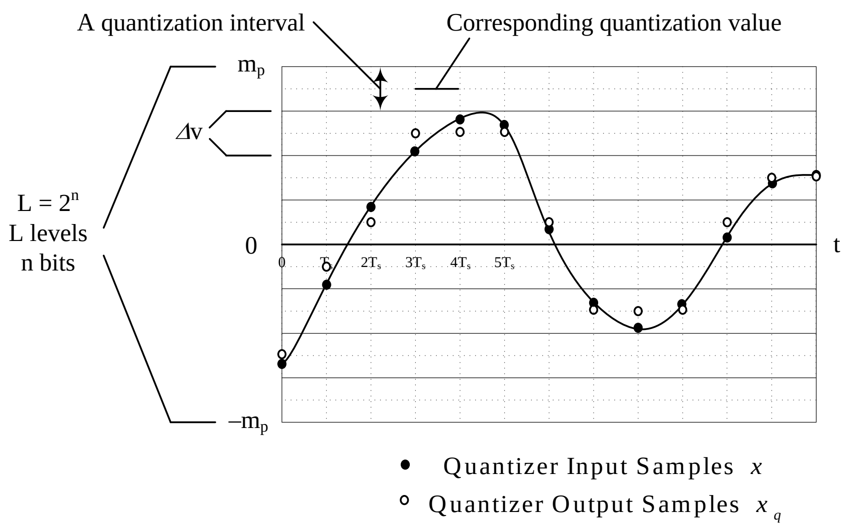
Using a 56 kbps modem would require 807,840,000 / 56,000 = 14426 seconds of transmission time = 4 hours.

For this reason, compression techniques are generally used to store and transmit pictures over slow transmission channels.

Quantization

The process of quantizing a signal is the first part of converting an sequence of analog samples to a PCM code. In quantization, an analog sample with an amplitude that may take value in a specific range is converted to a digital sample with an amplitude that takes one of a specific pre–defined set of quantization values. This is performed by dividing the range of possible values of the analog samples into L different levels, and assigning the center value of each level to any sample that falls in that quantization interval. The problem with this process is that it approximates the value of an analog sample with the nearest of the quantization values. So, for almost all samples, the quantized samples will differ from the original samples by a small amount. This amount is called the quantization error. To get some idea on the effect of this quantization error, quantizing audio signals results in a hissing noise similar to what you would hear when play a random signal.

Assume that a signal with power P_s is to be quantized using a quantizer with $L = 2^n$ levels ranging in voltage from $-m_p$ to m_p as shown in the figure below.

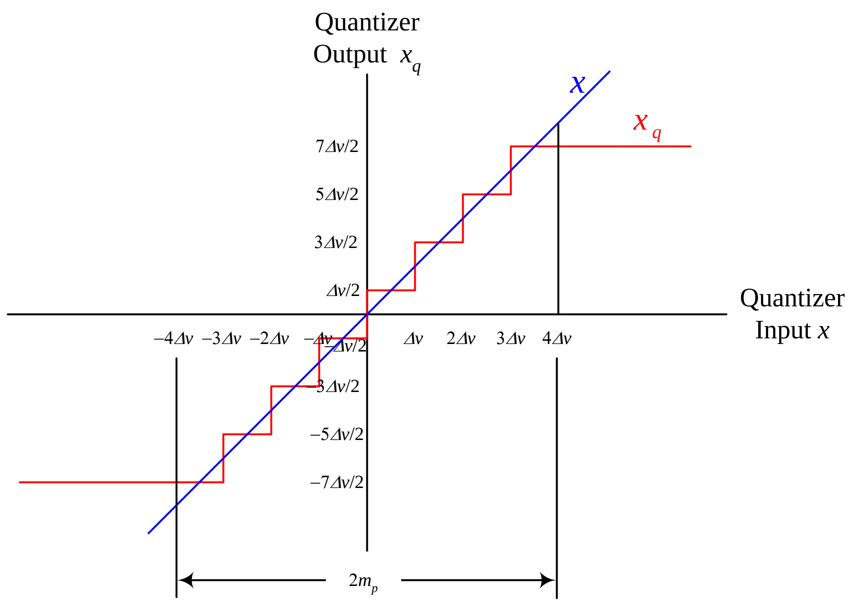


We can define the variable Δv to be the height of the each of the L levels of the quantizer as shown above. This gives a value of Δv equal to

$$\Delta v = \frac{2m_p}{L}.$$

Therefore, for a set of quantizers with the same m_p, the larger the number of levels of a quantizer, the smaller the size of each quantization interval, and for a set of quantizers with the same number of quantization intervals, the larger m_p is the larger the quantization interval length to accommodate all the quantization range.

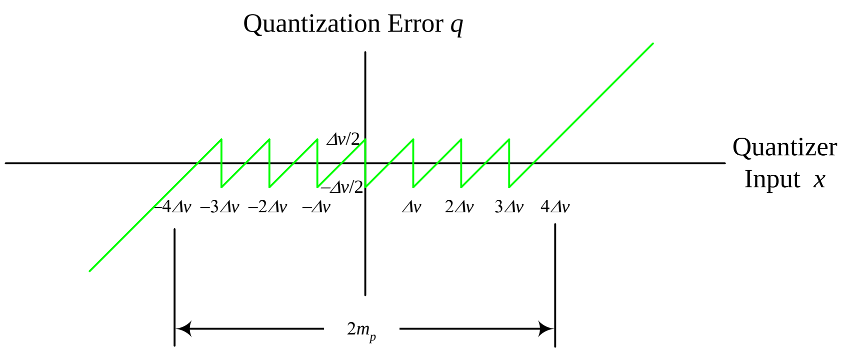
Now if we look at the input output characteristics of the quantizer, it will be similar to the red line in the following figure. Note that as long as the input is within the quantization range of the quantizer, the output of the quantizer represented by the red line follows the input of the quantizer. When the input of the quantizer exceeds the range of $-m_p$ to m_p, the output of the quantizer starts to deviate from the input and the quantization error (difference between an input and the corresponding output sample) increases significantly.



Now let us define the quantization error represented by the difference between the input sample and the corresponding output sample to be q, or

$$q = x - x_q \, .$$

Plotting this quantization error versus the input signal of a quantizer is seen next. Notice that the plot of the quantization error is obtained by taking the difference between the blow and red lines in the above figure.



It is seen from this figure that the quantization error of any sample is restricted between $-\Delta v/2$ and $\Delta v/2$ except when the input signal exceeds the range of quantization of $-m_p$ to m_p.

Quantization (Continued)

To understand the following, you will need to know something about probability theory. Assuming that the input signal is restricted between $-m_p$ to m_p, the resulting quantization error q (or we can call it quantization noise) will be a random process that is uniformly distributed between $-\Delta v/2$ and $\Delta v/2$ with a constant height of $1/\Delta v$. That is, all values of quantization error in the range $-\Delta v/2$ and $\Delta v/2$ are equally probable to happen. The power of such a random process can be easily found by finding the average of the square of all noise values multiplied by probability of each of these values of the noise occurring. So,

$$P_q = \int_{-\Delta v/2}^{\Delta v/2} q^2 \frac{1}{\Delta v} dq = \frac{1}{\Delta v}\left[\frac{q^3}{3}\right]_{q=-\Delta v/2}^{\Delta v/2}$$

$$= \frac{1}{\Delta v}\left[\frac{(\Delta v/2)^3}{3} - \frac{(-\Delta v/2)^3}{3}\right] = \frac{1}{\Delta v}\left[\frac{(\Delta v)^3}{24} + \frac{(\Delta v)^3}{24}\right]$$

$$= \frac{(\Delta v)^2}{12}$$

Now substituting for $\Delta v = \frac{2m_p}{L}$ in the above equation gives

$$P_q = \frac{(2m_p/L)^2}{12} = \frac{m_p^2}{3L^2},$$

As predicted, the power of the noise decreases as the number of levels L increases, and increases as the edge of the quantization range m_p increases.

Now let us define the Signal to Noise Ratio (SNR) as the ratio of the power of the input signal of the quantizer to the power of the noise introduced by the quantizer (note that the SNR has many other definitions used in communication systems depending on the applications)

$$SNR = \frac{\text{Signal Power}}{\text{Noise Power}} = \frac{P_s}{P_q}$$

$$= \frac{3L^2}{m_p^2} P_s.$$

In general the values of the SNR are either much greater than 1 or much less than 1. A more useful representation of the SNR can be obtained by using logarithmic scale or dB. We know that L of a quantizer is always a power of two or $L = 2^n$. Therefore,

$$SNR_{Linear} = \frac{3L^2}{m_p^2} P_s,$$

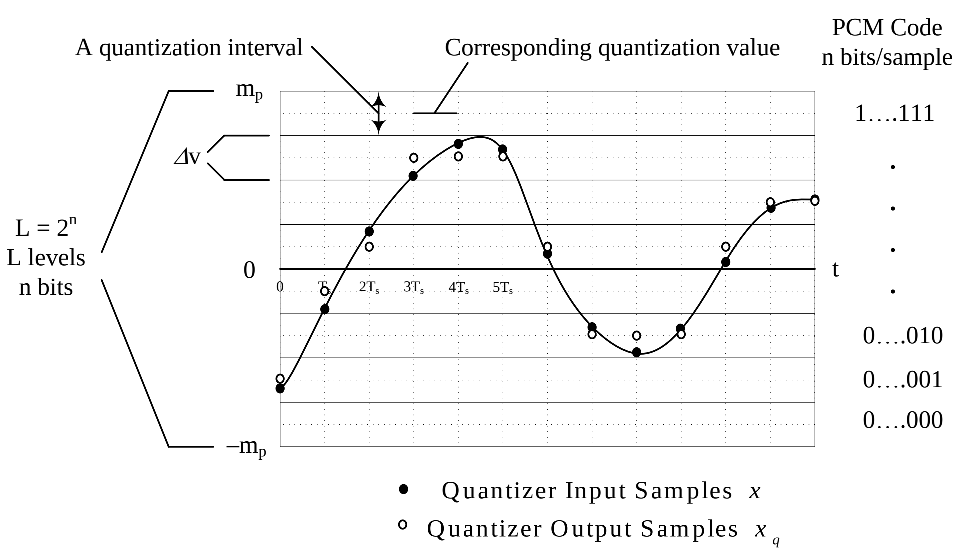
$$
\begin{aligned}
SNR_{dB} &= 10\cdot\log_{10}\left(\frac{3L^2}{m_p^2}P_s\right) = 10\cdot\log_{10}\left(\frac{3}{m_p^2}P_s\right) + 10\cdot\log_{10}\left(2^{2n}\right) \\
&= \underbrace{10\cdot\log_{10}\left(\frac{3}{m_p^2}P_s\right)}_{\alpha} + \underbrace{20n\cdot\log_{10}(2)}_{6n} \\
&= \alpha + 6n \qquad \text{dB.}
\end{aligned}
$$

Note that α shown in the above representation of the SNR is a constant when quantizing a specific signal with different quantizers as long as all of these quantizers have the same value of m_p.

It is clear that the SNR of a quantizer in dB increases linearly by 6 dB as we increase the number of bits that the quantizer uses by 1 bit. The cost for increasing the SNR of a quantizer is that more bits are generated and therefore either a higher bandwidth or a longer time period is required to transmit the PCM signal.

Generation of the PCM Signal

Now, once the signal has been quantized by the quantizer, the quantizer converts it to bits (1's and 0's) and outputs these bits. Looking at the figure in the previous lecture, which shown here for convenience. We see that each of the levels of the quantizer is assigned a code from 000…000 for the lowest quantization interval to 111…111 for the highest quantization interval as shown in the column to the left of the figure. The PCM signal is obtained by outputting the bits of the different samples one bit after the other and one sample after the other.



Differential Pulse Code Modulation (DPCM)

According to the Nyquist sampling criterion, a signal must be sampled at a sampling rate that is at least twice the highest frequency in the signal to be able to reconstruct it without aliasing. The samples of a signal that is sampled at that rate or close to this rate generally have little correlation between each other (knowing a sample does not give much information about the next sample). However, when a signal is highly oversampled (sampled at several times the Nyquist rate, the signal does not change a lot between from one sample to another. Consider, for example, a sine function that is sampled at the Nyquist rate. Consecutive samples of this signal may alternate over the whole range of amplitudes from –1 and 1. However, when this signal is sampled at a rate that is 100 times the Nyquist rate (sampling period is 1/100 of the sampling period in the previous case), consecutive samples will change a little from each other. This fact can be used to improve the performance of quantizers significantly by quantizing a signal that is the difference between consecutive samples instead of quantizing the original signal. This will result in either requiring a quantizer with much less number of bits (less information to transmit) or a quantizer with the same number of bits but much smaller quantization intervals (less quantization noise and much higher SNR).

Consider a signal $x(t)$ that is sampled to obtain the samples $x(kT_s)$, where T_s is the sampling period and k is an integer representing the sample number. For simplicity, the samples can be written in the form $x[k]$, where the sample period T_s is implied. Assume that the signal $x(t)$ is sampled at a very high sampling rate. We can define $d[k]$ to be the difference between the present sample of a signal and the previous sample, or

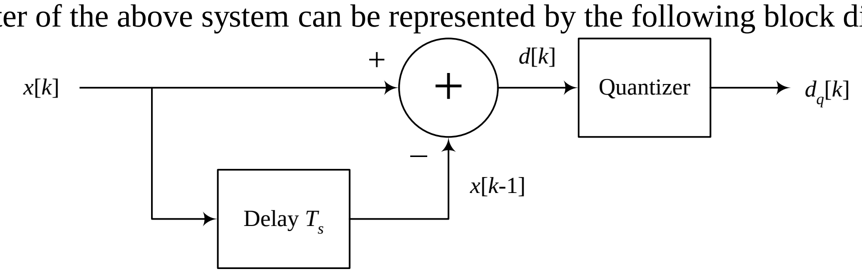
$$d[k] = x[k] - x[k-1].$$

Now this signal $d[k]$ can be quantized instead of $x[k]$ to give the quantized signal $d_q[k]$. As mentioned above, for signals $x(t)$ that are sampled at a rate much higher than the Nyquist rate, the range of values of $d[k]$ will be less than the range of values of $x[k]$.

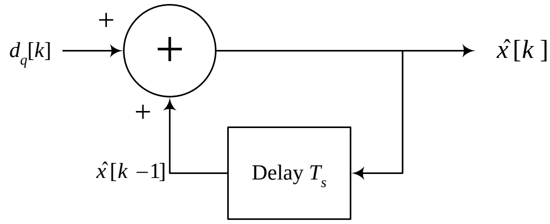
After the transmission of the quantized signal $d_q[k]$, theoretically we can reconstruct the original signal by doing an operation that is the inverse of the above operation. So, we can obtain an approximation of $x[k]$ using

$$\hat{x}[k] = d_q[k] + \hat{x}[k-1].$$

So, if $d_q[k]$ is close to $d[k]$, it appears from the above equation that obtained $\hat{x}[k]$ will be close to $d[k]$. However, this is generally not the case as will be shown later. The transmitter of the above system can be represented by the following block diagram.



The receiver that will attempt to reconstruct the original signal after transmitting it through the channel can be represented by the following block diagram.



Because we are quantizing a difference signal and transmitting that difference over the channel, the reconstructed signal may suffer from one or two possible problems.

1. Accumulation of quantization noise: the above system suffers from the possible accumulation of the quantization noise. Unlike the quantization of a signal where quantization error in each sample of that signal is completely independent from the quantization error in other samples, the quantization error in this system may accumulate to the point that it will result in a reconstructed signal that is very different from the original signal. This is illustrated using the following table. Consider the samples of the input signal $x[k]$ given in the table. The reconstructed signal is given by $\hat{x}[k]$ shown in table. Assume the quantizer used to quantize $d[k]$ is an 8–level quantizer with quantization intervals [–4,–3), [–3,–2), [–2,–1), … , [3,4) and the output quantization levels are the center points in each interval (–3.5, –2.5, –1.5, … , 3.5).

k	–1	0	1	2	3	4	5	6	7	8	9
$x[k]$	0	0.3	1.5	0.7	1.0	2.3	3.7	2.8	3.5	2.8	0
$x[k-1]$	0	0	0.3	1.5	0.7	1.0	2.3	3.7	2.8	3.5	3.1
$d[k]$	0	0.3	1.2	–0.8	0.3	1.3	1.4	–0.9	0.7	–0.7	–2.8
Quantization Up/Down	U (or D)	U	U	U	U	U	U	U	D	U	U
$d_q[k]$	0.5	0.5	1.5	–0.5	0.5	1.5	1.5	–0.5	0.5	–0.5	–2.5
$\hat{x}[k-1]$	0	0.5	1.0	2.5	2.0	2.5	4.0	5.5	5.0	5.5	5.0
$\hat{x}[k]$	0.5	1.0	2.5	2.0	2.5	4.0	5.5	5.0	5.5	5.0	2.5
$\hat{x}[k]-x[k]$	0.5	0.7	1.0	1.3	1.5	1.7	1.8	2.2	2.0	2.2	2.5
Err. Direction Up/Down	U	U	U	U	U	U	U	U	D	U	U

So, it is clear from this table that if the quantization error for a series of samples is going in one direction, the error adds up to produce a output signal that deviates from the original signal. Note that the error between the original and reconstructed samples always increased except when the quantization error switched direction at k = 7 (the shaded box).

2. Effect of transmission errors: in a regular PCM system, the effect of an error that happens in the transmitted signal is only limited to the sample in which the error occurs. In DPCM, an error that occurs in the transmitted signal will cause all the reconstructed samples at the receiver after that error occurs to have errors. Therefore, even if quantization error did not accumulate, an error caused by the channel will cause all successive samples to be wrong. Try this as an exercise by constructing g a table similar to the one above. Intentionally introduce an error in the reconstructed signal at a point and see what happens to the remainder of the reconstructed signal.