Signal Processing Handbook

Savanah Ma

First Edition, 2012

ISBN 978-81-323-3660-0

Published by:
University Publications
4735/22 Prakashdeep Bldg,
Ansari Road, Darya Ganj,
Delhi - 110002
Email: info@wtbooks.com

Table of Contents

Chapter- 1

Signal Processing

Signal processing is an area of electrical engineering and applied mathematics that deals with operations on or analysis of signals, in either discrete or continuous time, to perform useful operations on those signals. Signals of interest can include sound, images, time-varying measurement values and sensor data, for example biological data such as electrocardiograms, control system signals, telecommunication transmission signals such as radio signals, and many others. Signals are analog or digital electrical representations of time-varying or spatial-varying physical quantities. In the context of signal processing, arbitrary binary data streams and on-off signalling are not considered as signals, but only analog and digital signals that are representations of analog physical quantities.

Typical operations and applications

Processing of signals includes the following operations and algorithms with application examples:

- Filtering (for example in tone controls and equalizers)
- Smoothing, deblurring (for example in image enhancement)
- Adaptive filtering (for example for echo-cancellation in a conference telephone, or denoising for aircraft identification by radar)
- Spectrum analysis (for example in magnetic resonance imaging, tomographic reconstruction and OFDM modulation)
- Digitization, reconstruction and compression (for example, image compression, sound coding and other source coding)
- Storage (in digital delay lines and reverb)
- Modulation (in modems)
- Wavetable synthesis (in modems and music synthesizers)
- Feature extraction (for example speech-to-text conversion and optical character recognition)
- Pattern recognition and correlation analysis (in spread spectrum receivers and computer vision)
- Prediction

- A variety of other operations

In communication systems, signal processing may occur at OSI layer 1, the Physical Layer (modulation, equalization, multiplexing, etc) in the seven layer OSI model, as well as at OSI layer 6, the Presentation Layer (source coding, including analog-to-digital conversion and data compression).

History

According to Alan V. Oppenheim and Ronald W. Schafer, the principles of signal processing can be found in the classical numerical analysis techniques of the 17th century. They further state that the "digitalization" or digital refinement of these techniques can be found in the digital control systems of the 1940s and 1950s.

Mathematical topics embraced by signal processing

- Linear signals and systems, and transform theory
- System identification and classification
- Calculus
- Differential Equations
- Vector spaces and Linear algebra
- Functional analysis
- Probability and stochastic processes
- Detection theory
- Estimation theory
- Optimization
- Programming
- Numerical methods
- Iterative methods

Categories of signal processing

Analog signal processing

Analog signal processing is for signals that have not been digitized, as in classical radio, telephone, radar, and television systems. This involves linear electronic circuits such as passive filters, active filters, additive mixers, integrators and delay lines. It also involves non-linear circuits such as compandors, multiplicators (frequency mixers and voltage-controlled amplifiers), voltage-controlled filters, voltage-controlled oscillators and phase-locked loops.

Discrete time signal processing

Discrete time signal processing is for sampled signals that are considered as defined only at discrete points in time, and as such are quantized in time, but not in magnitude.

Analog discrete-time signal processing is a technology based on electronic devices such as sample and hold circuits, analog time-division multiplexers, analog delay lines and analog feedback shift registers. This technology was a predecessor of digital signal processing (see below), and is still used in advanced processing of gigahertz signals.

The concept of discrete-time signal processing also refers to a theoretical discipline that establishes a mathematical basis for digital signal processing, without taking quantization error into consideration.

Digital signal processing

Digital signal processing is for signals that have been digitized. Processing is done by general-purpose computers or by digital circuits such as ASICs, field-programmable gate arrays or specialized digital signal processors (DSP chips). Typical arithmetical operations include fixed-point and floating-point, real-valued and complex-valued, multiplication and addition. Other typical operations supported by the hardware are circular buffers and look-up tables. Examples of algorithms are the Fast Fourier transform (FFT), finite impulse response (FIR) filter, Infinite impulse response (IIR) filter, and adaptive filters such as the Wiener and Kalman filters.

Fields of signal processing

- Statistical signal processing — analyzing and extracting information from signals and noise based on their stochastic properties
- Audio signal processing — for electrical signals representing sound, such as speech or music
- Speech signal processing — for processing and interpreting spoken words
- Image processing — in digital cameras, computers, and various imaging systems
- Video processing — for interpreting moving pictures
- Array processing — for processing signals from arrays of sensors
- Time-frequency signal processing — for processing non-stationary signals
- Filtering — used in many fields to process signals
- Seismic signal processing
- Data mining.

Chapter- 2

Analog Signal Processing

Analog signal processing is any signal processing conducted on analog signals by analog means. "Analog" indicates something that is mathematically represented as a set of continuous values. This differs from "digital" which uses a series of discrete quantities to represent signal. Analog values are typically represented as a voltage, electric current, or electric charge around components in the electronic devices. An error or noise affecting such physical quantities will result in a corresponding error in the signals represented by such physical quantities.

Examples of analog signal processing include crossover filters in loudspeakers, "bass", "treble" and "volume" controls on stereos, and "tint" controls on TVs. Common analog processing elements include capacitors, resistors, inductors and transistors.

Tools used in analog signal processing

A system's behavior can be mathematically modeled and is represented in the time domain as h(t) and in the frequency domain as H(s), where s is a complex number in the form of s=a+ib, or s=a+jb in electrical engineering terms (electrical engineers use j because current is represented by the variable i). Input signals are usually called x(t) or X(s) and output signals are usually called y(t) or Y(s).

Convolution

Convolution is the basic concept in signal processing that states an input signal can be combined with the system's function to find the output signal. It is the integral of the product of two waveforms after one has reversed and shifted; the symbol for convolution is *.

$$y(t) = (x * h)(t) = \int_a^b x(\tau)h(t-\tau)\,d\tau$$

That is the convolution integral and is used to find the convolution of a signal and a system; typically a = -∞ and b = +∞.

Consider two waveforms f and g. By calculating the convolution, we determine how much a reversed function g must be shifted along the x-axis to become identical to function f. The convolution function essentially reverses and slides function g along the axis, and calculates the integral of their (f and the reversed and shifted g) product for each possible amount of sliding. When the functions match, the value of (f*g) is maximized. This occurs because when positive areas (peaks) or negative areas (troughs) are multiplied, they contribute to the integral.

Fourier transform

The Fourier transform is a function that transforms a signal or system in the time domain into the frequency domain, but it only works for certain ones. The constraint on which systems or signals can be transformed by the Fourier Transform is that:

$$\int_{-\infty}^{\infty} |x(t)|\,dt < \infty$$

This is the Fourier transform integral:

$$X(j\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t}\,dt$$

Most of the time the Fourier transform integral isn't used to determine the transform. Usually a table of transform pairs is used to find the Fourier transform of a signal or system. The inverse Fourier transform is used to go from frequency domain to time domain:

$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(j\omega)e^{j\omega t}\,d\omega$$

Each signal or system that can be transformed has a unique Fourier transform; there is only one time signal and one frequency signal that goes together.

Laplace transform

The Laplace transform is a generalized Fourier transform. It allows a transform of any system or signal because it is a transform into the complex plane instead of just the jω line like the Fourier transform. The major difference is that the Laplace transform has a region of convergence for which the transform is valid. This implies that a signal in frequency may have more than one signal in time; the correct time signal for the transform is determined by the region of convergence. If the region of convergence includes the jω axis, jω can be substituted into the Laplace transform for s and it's the same as the Fourier transform. The Laplace transform is:

$$X(s) = \int_{0^-}^{\infty} x(t) e^{-st}\, dt$$

and the inverse Laplace transform, if all the singularities of X(s) are in the left half of the complex plane, is:

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(s) e^{st}\, ds$$

Bode plots

Bode plots are plots of magnitude vs. frequency and phase vs. frequency for a system. The magnitude axis is in Decibel (dB). The phase axis is in either degrees or radians. The frequency axes are in a logarithmic scale. These are useful because for sinusoidal inputs, the output is the input multiplied by the value of the magnitude plot at the frequency and shifted by the value of the phase plot at the frequency.

Domains

Time domain

This is the domain that most people are familiar with. A plot in the time domain shows the amplitude of the signal with respect to time.

Frequency domain

A plot in the frequency domain shows either the phase shift or magnitude of a signal at each frequency that it exists at. These can be found by taking the Fourier transform of a time signal and are plotted similarly to a bode plot.

Signals

While any signal can be used in analog signal processing, there are many types of signals that are used very frequently.

Sinusoids

Sinusoids are the building block of analog signal processing. All real world signals can be represented as an infinite sum of sinusoidal functions via a Fourier series. A sinusoidal function can be represented in terms of an exponential by the application of Euler's Formula.

Impulse

An impulse (Dirac delta function) is defined as a signal that has an infinite magnitude and an infinitesimally narrow width with an area under it of one, centered at zero. An impulse can be represented as an infinite sum of sinusoids that includes all possible frequencies. It is not, in reality, possible to generate such a signal, but it can be sufficiently approximated with a large amplitude, narrow pulse, to produce the theoretical impulse response in a network to a high degree of accuracy. The symbol for an impulse is δ(t). If an impulse is used as an input to a system, the output is known as the impulse response. The impulse response defines the system because all possible frequencies are represented in the input.

Step

A unit step function, also called the Heaviside step function, is a signal that has a magnitude of zero before zero and a magnitude of one after zero. The symbol for a unit step is u(t). If a step is used as the input to a system, the output is called the step response. The step response shows how a system responds to a sudden input, similar to turning on a switch. The period before the output stabilizes is called the transient part of a signal. The step response can be multiplied with other signals to show how the system responds when an input is suddenly turned on.

The unit step function is related to the Dirac delta function by;

$$\mathrm{u}(t) = \int_{-\infty}^{t} \delta(s) ds$$

Systems

Linear time-invariant (LTI)

Linearity means that if you have two inputs and two corresponding outputs, if you take a linear combination of those two inputs you will get a linear combination of the outputs. An example of a linear system is a first order low-pass or high-pass filter. Linear systems are made out of analog devices that demonstrate linear properties. These devices don't have to be entirely linear, but must have a region of operation that is linear. An operational amplifier is a non-linear device, but has a region of operation that is linear, so it can be modeled as linear within that region of operation. Time-invariance means it doesn't matter when you start a system, the same output will result. For example, if you have a system and put an input into it today, you would get the same output if you started the system tomorrow instead. There aren't any real systems that are LTI, but many systems can be modeled as LTI for simplicity in determining what their output will be. All systems have some dependence on things like temperature, signal level or other factors that cause them to be non-linear or non-time-invariant, but most are stable enough to model as LTI. Linearity and time-invariance are important because they are the only

types of systems that can be easily solved using conventional analog signal processing methods. Once a system becomes non-linear or non-time-invariant, it becomes a non-linear differential equations problem, and there are very few of those that can actually be solved. (Haykin & Van Veen 2003)

Common systems

Some common systems used in everyday life are filters, AM/FM radio, electric guitars and musical instrument amplifiers. Filters are used in almost everything that has electronic circuitry. Radio and television are good examples of everyday uses of filters. When a channel is changed on an analog television set or radio, an analog filter is used to pick out the carrier frequency on the input signal. Once it's isolated, the television or radio information being broadcast is used to form the picture and/or sound. Another common analog system is an electric guitar and its amplifier. The guitar uses a magnet with a coil wrapped around it (inductor) to turn the vibration of the strings into a small electric current. The current is then filtered, amplified and sent to a speaker in the amplifier. Most amplifiers are analog because they are easier and cheaper to make than making a digital amplifier. There are also many analog guitar effects pedals, although a large number of pedals are now digital (they turn the input current into a digitized value, perform an operation on it, then convert it back into an analog signal).

Chapter- 3

Digital Signal Processing

Digital signal processing (**DSP**) is concerned with the representation of signals by a sequence of numbers or symbols and the processing of these signals. Digital signal processing and analog signal processing are subfields of signal processing. DSP includes subfields like: audio and speech signal processing, sonar and radar signal processing, sensor array processing, spectral estimation, statistical signal processing, digital image processing, signal processing for communications, control of systems, biomedical signal processing, seismic data processing, etc.

The goal of DSP is usually to measure, filter and/or compress continuous real-world analog signals. The first step is usually to convert the signal from an analog to a digital form, by *sampling* it using an analog-to-digital converter (ADC), which turns the analog signal into a stream of numbers. However, often, the required output signal is another analog output signal, which requires a digital-to-analog converter (DAC). Even if this process is more complex than analog processing and has a discrete value range, the application of computational power to digital signal processing allows for many advantages over analog processing in many applications, such as error detection and correction in transmission as well as data compression.

DSP algorithms have long been run on standard computers, on specialized processors called digital signal processors (DSPs), or on purpose-built hardware such as application-specific integrated circuit (ASICs). Today there are additional technologies used for digital signal processing including more powerful general purpose microprocessors, field-programmable gate arrays (FPGAs), digital signal controllers (mostly for industrial apps such as motor control), and stream processors, among others.

Signal sampling

With the increasing use of computers the usage of and need for digital signal processing has increased. In order to use an analog signal on a computer it must be digitized with an analog-to-digital converter. Sampling is usually carried out in two stages, discretization and quantization. In the discretization stage, the space of signals is partitioned into equivalence classes and quantization is carried out by replacing the signal with

representative signal of the corresponding equivalence class. In the quantization stage the representative signal values are approximated by values from a finite set.

The Nyquist–Shannon sampling theorem states that a signal can be exactly reconstructed from its samples if the sampling frequency is greater than twice the highest frequency of the signal; but requires an infinite number of samples. In practice, the sampling frequency is often significantly more than twice that required by the signal's limited bandwidth.

A digital-to-analog converter is used to convert the digital signal back to analog. The use of a digital computer is a key ingredient in digital control systems.

DSP domains

In DSP, engineers usually study digital signals in one of the following domains: time domain (one-dimensional signals), spatial domain (multidimensional signals), frequency domain, autocorrelation domain, and wavelet domains. They choose the domain in which to process a signal by making an informed guess (or by trying different possibilities) as to which domain best represents the essential characteristics of the signal. A sequence of samples from a measuring device produces a time or spatial domain representation, whereas a discrete Fourier transform produces the frequency domain information, that is the frequency spectrum. Autocorrelation is defined as the cross-correlation of the signal with itself over varying intervals of time or space.

Time and space domains

The most common processing approach in the time or space domain is enhancement of the input signal through a method called filtering. Digital filtering generally consists of some linear transformation of a number of surrounding samples around the current sample of the input or output signal. There are various ways to characterize filters; for example:

- A "linear" filter is a linear transformation of input samples; other filters are "non-linear". Linear filters satisfy the superposition condition, i.e. if an input is a weighted linear combination of different signals, the output is an equally weighted linear combination of the corresponding output signals.

- A "causal" filter uses only previous samples of the input or output signals; while a "non-causal" filter uses future input samples. A non-causal filter can usually be changed into a causal filter by adding a delay to it.

- A "time-invariant" filter has constant properties over time; other filters such as adaptive filters change in time.

- Some filters are "stable", others are "unstable". A stable filter produces an output that converges to a constant value with time, or remains bounded within a finite

interval. An unstable filter can produce an output that grows without bounds, with bounded or even zero input.

- A "finite impulse response" (FIR) filter uses only the input signals, while an "infinite impulse response" filter (IIR) uses both the input signal and previous samples of the output signal. FIR filters are always stable, while IIR filters may be unstable.

Filters can be represented by block diagrams which can then be used to derive a sample processing algorithm to implement the filter using hardware instructions. A filter may also be described as a difference equation, a collection of zeroes and poles or, if it is an FIR filter, an impulse response or step response.

The output of a digital filter to any given input may be calculated by convolving the input signal with the impulse response.

Frequency domain

Signals are converted from time or space domain to the frequency domain usually through the Fourier transform. The Fourier transform converts the signal information to a magnitude and phase component of each frequency. Often the Fourier transform is converted to the power spectrum, which is the magnitude of each frequency component squared.

The most common purpose for analysis of signals in the frequency domain is analysis of signal properties. The engineer can study the spectrum to determine which frequencies are present in the input signal and which are missing.

In addition to frequency information, phase information is often needed. This can be obtained from the Fourier transform. With some applications, how the phase varies with frequency can be a significant consideration.

Filtering, particularly in non-realtime work can also be achieved by converting to the frequency domain, applying the filter and then converting back to the time domain. This is a fast, O(n log n) operation, and can give essentially any filter shape including excellent approximations to brickwall filters.

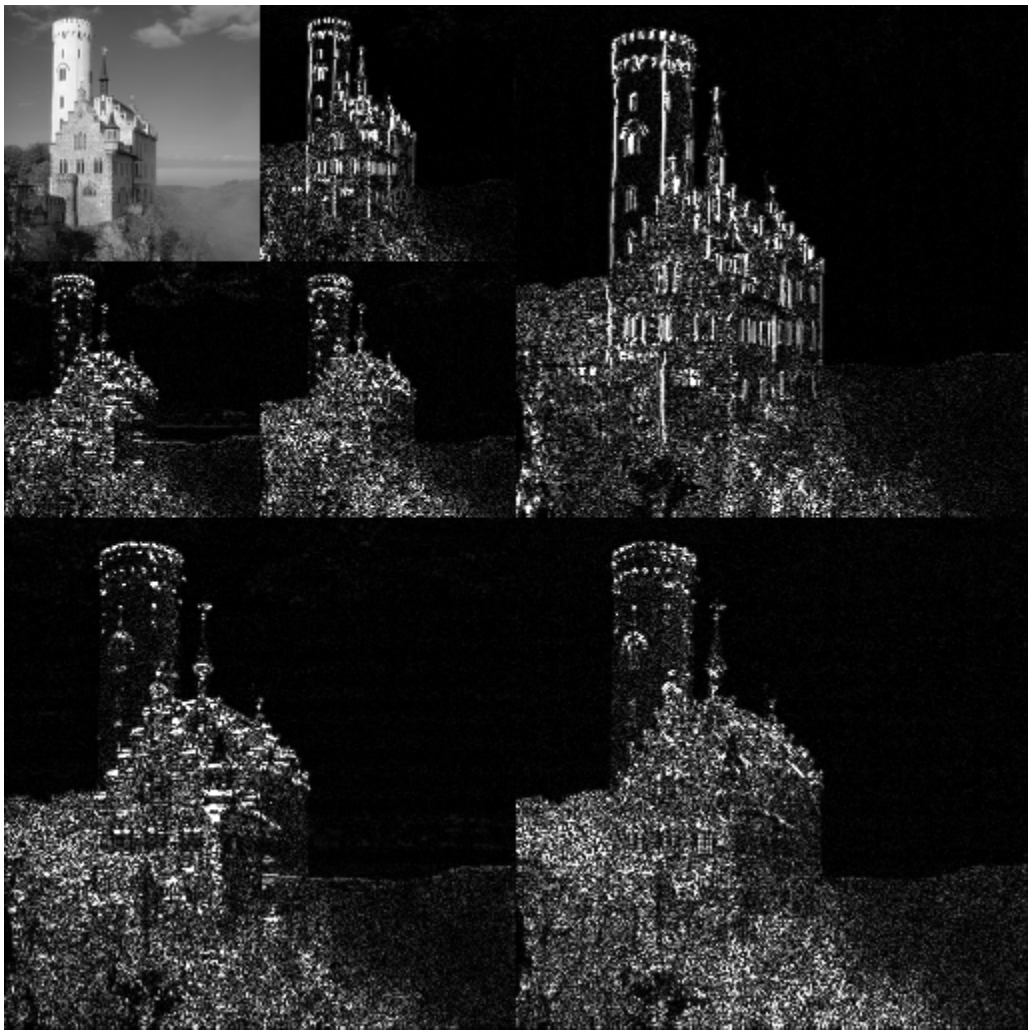
There are some commonly used frequency domain transformations. For example, the cepstrum converts a signal to the frequency domain through Fourier transform, takes the logarithm, then applies another Fourier transform. This emphasizes the frequency components with smaller magnitude while retaining the order of magnitudes of frequency components.

Frequency domain analysis is also called *spectrum-* or *spectral analysis*.

Z-plane analysis

Whereas analog filters are usually analysed in terms of transfer functions in the s plane using Laplace transforms, digital filters are analysed in the z plane in terms of Z-transforms. A digital filter may be described in the z plane by its characteristic collection of zeroes and poles.

Wavelet



An example of the 2D discrete wavelet transform that is used in JPEG2000. The original image is high-pass filtered, yielding the three large images, each describing local changes in brightness (details) in the original image. It is then low-pass filtered and downscaled, yielding an approximation image; this image is high-pass filtered to produce the three smaller detail images, and low-pass filtered to produce the final approximation image in the upper-left.

In numerical analysis and functional analysis, a **discrete wavelet transform** (DWT) is any wavelet transform for which the wavelets are discretely sampled. As with other

wavelet transforms, a key advantage it has over Fourier transforms is temporal resolution: it captures both frequency *and* location information (location in time).

Applications

The main applications of DSP are audio signal processing, audio compression, digital image processing, video compression, speech processing, speech recognition, digital communications, RADAR, SONAR, seismology and biomedicine. Specific examples are speech compression and transmission in digital mobile phones, room correction of sound in hi-fi and sound reinforcement applications, weather forecasting, economic forecasting, seismic data processing, analysis and control of industrial processes, medical imaging such as CAT scans and MRI, MP3 compression, computer graphics, image manipulation, hi-fi loudspeaker crossovers and equalization, and audio effects for use with electric guitar amplifiers.

Implementation

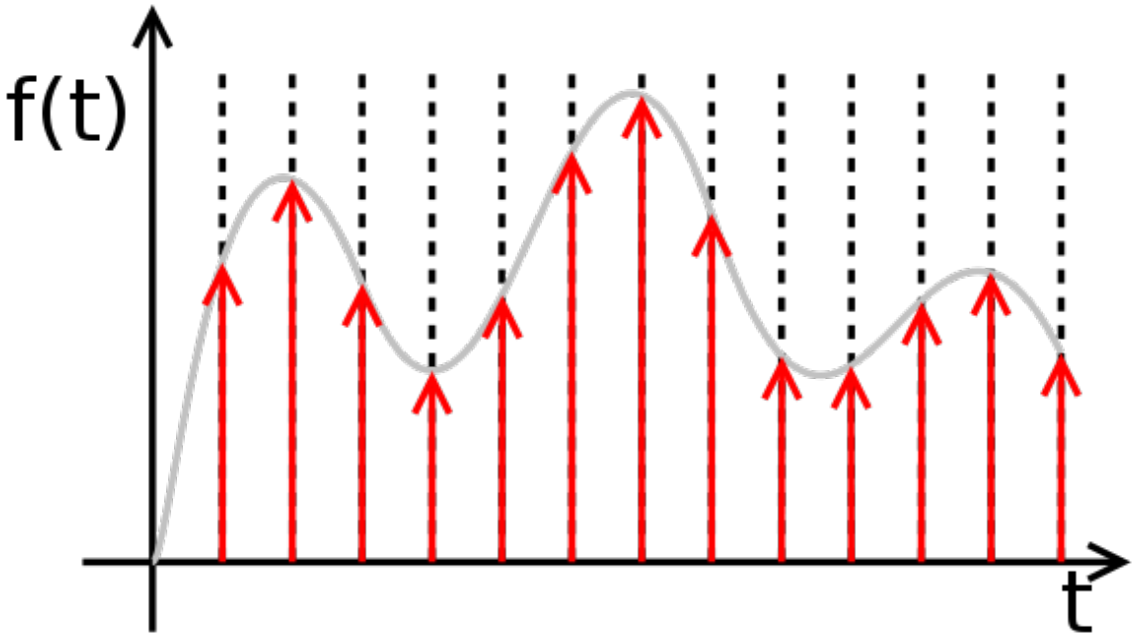
Digital signal processing is often implemented using specialised microprocessors such as the DSP56000, the TMS320, or the SHARC. These often process data using fixed-point arithmetic, although some versions are available which use floating point arithmetic and are more powerful. For faster applications FPGAs might be used. Beginning in 2007, multicore implementations of DSPs have started to emerge from companies including Freescale and Stream Processors, Inc. For faster applications with vast usage, ASICs might be designed specifically. For slow applications, a traditional slower processor such as a microcontroller may be adequate. Also a growing number of DSP applications are now being implemented on Embedded Systems using powerful PCs with a Multi-core processor.

Techniques

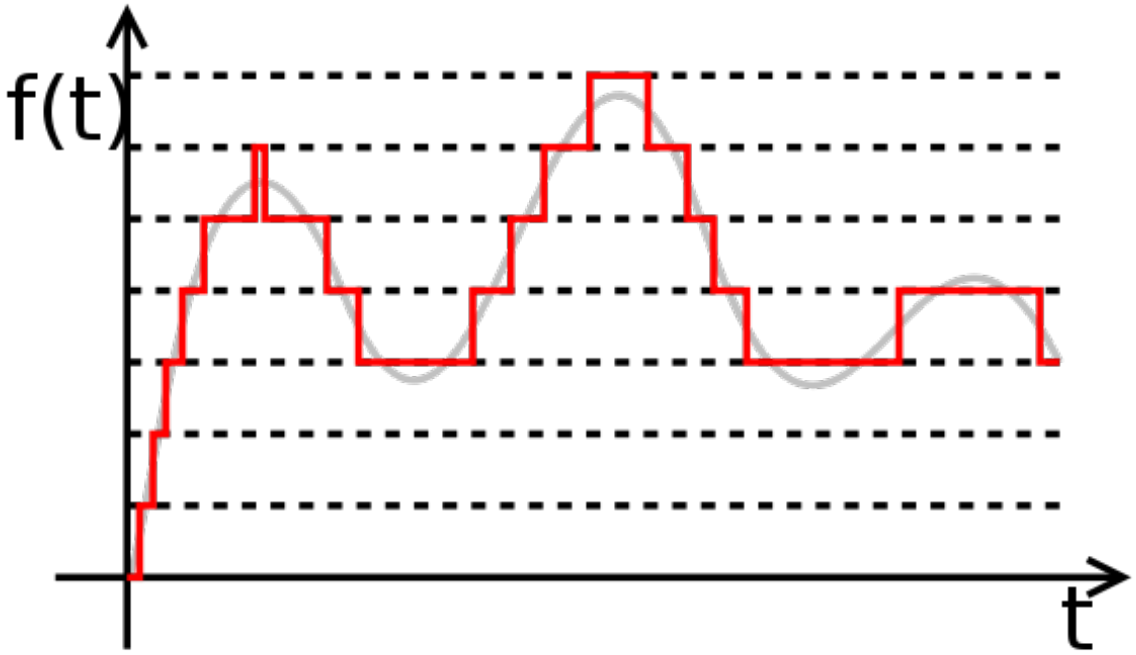
- Bilinear transform
- Discrete Fourier transform
- Discrete-time Fourier transform
- Filter design
- LTI system theory
- Minimum phase
- Transfer function
- Z-transform
- Goertzel algorithm
- s-plane

Chapter- 4

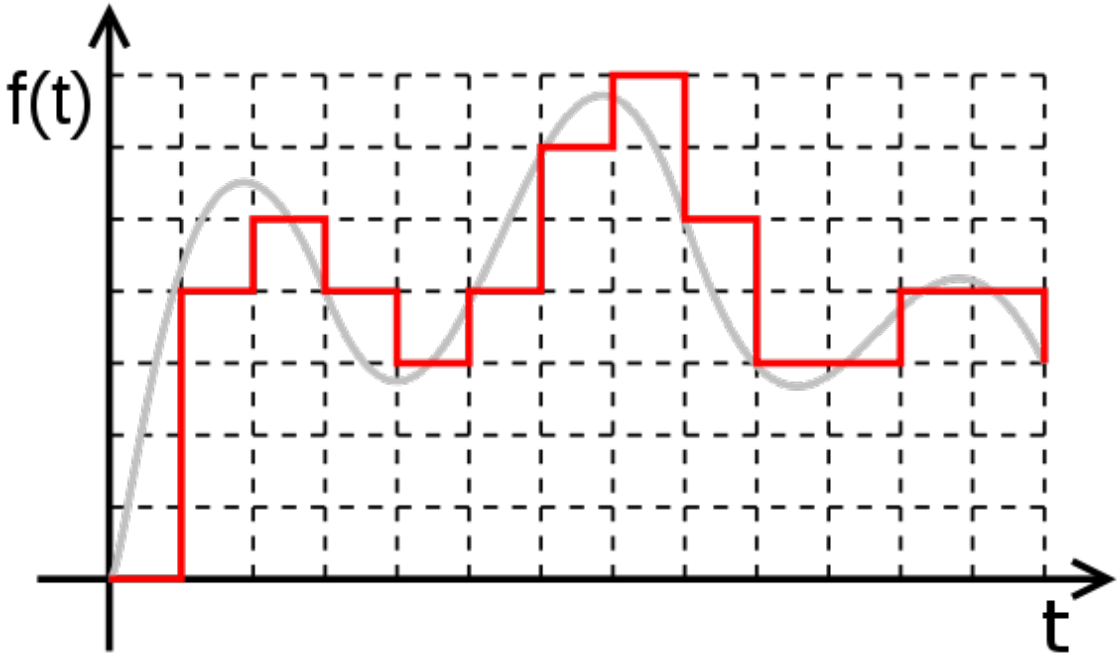
Quantization (Signal Processing)



Sampled signal (discrete signal): discrete time, continuous values.



Quantized signal: continuous time, discrete values.



Digital signal (sampled, quantized): discrete time, discrete values.

In digital signal processing, **quantization** is the process of approximating ("mapping") a continuous range of values (or a very large set of possible discrete values) by a relatively small ("finite") set of ("values which can still take on continuous range") discrete

symbols or integer values. For example, rounding a real number in the interval [0,100] to an integer $0, 1, 2 \ldots, 100.$

In other words, quantization can be described as a mapping that represents a finite continuous interval $I = [a,b]$ of the range of a continuous valued signal, with a single number $c,$ which is also on that interval. For example, rounding to the nearest integer (rounding ½ up) replaces the interval $[c - .5,c + .5)$ with the number $c,$ for integer $c.$ After that quantization we produce a finite set of values which can be encoded by binary techniques for example.

In signal processing, quantization refers to approximating the *output* by one of a discrete and finite set of values, while replacing the *input* by a discrete set is called *discretization,* and is done by sampling: the resulting sampled signal is called a *discrete signal* (discrete *time*), and need not be quantized (it can have continuous *values*). To produce a digital signal (discrete time and discrete values), one both samples (discrete time) and quantizes the resulting sample values (discrete values).

Quantization Noise

When a continuous signal is quantized the difference between the continuous signal and the quantized signal is an error. Strictly speaking this error is distortion since the same signal quantized repeatedly results in the same error. If a periodic signal like a sine wave is synchronously sampled and quantized then the quantized signal will exhibit harmonic distortion. However, even though it is actually distortion, it can be analyzed as noise. If the quantization is uniform and the width of the quantization interval is α, then the noise power, n, is $n = \frac{\alpha^2}{12}.$

Applications

In electronics, **adaptive quantization** is a quantization process that varies the step size based on the changes of the input signal, as a means of efficient compression. Two approaches commonly used are *forward adaptive quantization* and *backward adaptive quantization*.

In digital signal processing the quantization process is the necessary and natural follower of the sampling operation. It is necessary because in practice the digital computer with its general purpose CPU is used to implement DSP algorithms. And since computers can only process finite word length (finite resolution/precision) quantities, any infinite precision continuous valued signal should be quantized to fit a finite resolution, so that it can be represented (stored) in CPU registers and memory.

We shall *be aware* of the fact that, it is not the continuous values of the analog function that inhibits its binary encoding, rather it is the existence of infinitely many such values due to the definition of continuity,(which therefore requires infinitely many bits to

represent). For example we can design a quantizer such that it represents a signal with a single bit (just two levels) such that, one level is "pi=3,14..." (say encoded with a 1) and the other level is "e=2.7183..." (say encoded with a 0), as we can see, the quantized values of the signal take on infinite precision, irrational numbers. But there are only two levels. And we can represent the output of the quantizer with a binary symbol. Concluding from this we can see that it is not the discreteness of the quantized values that enable them to be encoded but the finiteness enabling the encoding with finite number of bits.

In theory there is no relation between quantization values and binary code words used to encode them (rather than a table that shows the corresponding mapping, just as examplified above). However due to practical reasons we may tend to use code words such that their binary mathematical values has a relation with the quantization levels that is encoded. And this last option merges the first two paragraphs in such a way that, if we wish to process the output of a quantizer within a DSP/CPU system (which is always the case) then we can not allow the representation levels of the quantizers to take on arbitrary values, but only a restricted range such that they can fit in computer registers.

A quantizer is identified with its number of levels M, the decision boundaries {di} and the corresponding representation values {ri}.

The output of a quantizer has two important properties: 1) a Distortion resulting from the approximation and 2) a Bit-Rate resulting from binary encoding of its levels. Therefore the Quantizer design problem is a Rate-Distortion optimization type.

If we are only allowed to use fixed length code for the output level encoding (the practical case) then the problem reduces into a distortion minimization one.

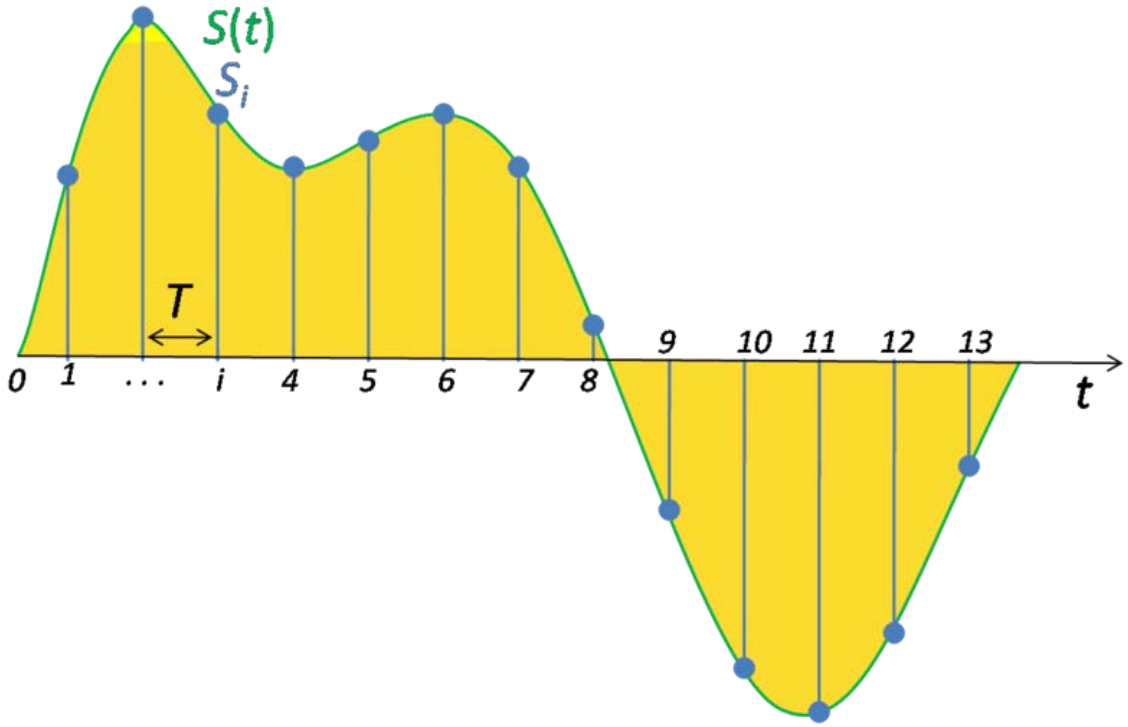
The design of a quantizer usually means the process to find the sets {di} and {ri} such that a measure of optimality is satisfied (such as MMSEQ (Minimum Mean Squared Quantization Error))

Given the number of levels M, the optimal quantizer which minimizes the MSQE with regards to the given signal statistics is called the Max-Lloyd quantizer, which is a non-uniform type in general.

The most common quantizer type is the uniform one. It is simple to design and implement and for most cases it suffices to get satisfactory results. Indeed by the very inherent nature of the design process, a given quantizer will only produce optimal results for the assumed signal statistics. Since it is very difficult to correctly predict that in advance, any static design will never produce actual optimal performance whenever the input statistics deviates from that of the design assumption. The only solution is to use an adaptive quantizer.

Chapter- 5

Sampling (Signal Processing)



Signal sampling representation. The continuous signal is represented with a green color whereas the discrete samples are in blue.

In signal processing, **sampling** is the reduction of a continuous signal to a discrete signal. A common example is the conversion of a sound wave (a continuous-time signal) to a sequence of samples (a discrete-time signal).

A **sample** refers to a value or set of values at a point in time and/or space.

A **sampler** is a subsystem or operation that extracts samples from a continuous signal. A theoretical ideal sampler produces samples equivalent to the instantaneous value of the continuous signal at the desired points.

Theory

For convenience, we will discuss signals which vary with time. However, the same results can be applied to signals varying in space or in any other dimension and similar results are obtained in two or more dimensions.

Let $x(t)$ be a continuous signal which is to be sampled, and that sampling is performed by measuring the value of the continuous signal every T seconds, which is called the sampling interval. Thus, the sampled signal $x[n]$ given by**:**

$$x[n] = x(nT), \text{ with } n = 0, 1, 2, 3, ...$$

The sampling frequency or sampling rate f_s is defined as the number of samples obtained in one second, or $f_s = 1/T$. The sampling rate is measured in hertz or in samples per second.

We can now ask: under what circumstances is it possible to reconstruct the original signal completely and exactly (perfect reconstruction)?

A partial answer is provided by the Nyquist–Shannon sampling theorem, which provides a sufficient (but not always necessary) condition under which perfect reconstruction is possible. The sampling theorem guarantees that bandlimited signals (i.e., signals which have a maximum frequency) can be reconstructed perfectly from their sampled version, if the sampling rate is more than twice the maximum frequency. Reconstruction in this case can be achieved using the Whittaker–Shannon interpolation formula.

The frequency equal to one-half of the sampling rate is therefore a bound on the highest frequency that can be unambiguously represented by the sampled signal. This frequency (half the sampling rate) is called the Nyquist frequency of the sampling system. Frequencies above the Nyquist frequency f_N can be observed in the sampled signal, but their frequency is ambiguous. That is, a frequency component with frequency f cannot be distinguished from other components with frequencies $Nf_N + f$ and $Nf_N - f$ for nonzero integers N. This ambiguity is called aliasing. To handle this problem as gracefully as possible, most analog signals are filtered with an anti-aliasing filter (usually a low-pass filter with cutoff near the Nyquist frequency) before conversion to the sampled discrete representation.

Observation period

The observation period is the span of time during which a series of data samples are collected at regular intervals. More broadly, it can refer to any specific period during which a set of data points is gathered, regardless of whether or not the data is periodic in nature. Thus a researcher might study the incidence of earthquakes and tsunamis over a particular time period, such as a year or a century.

The observation period is simply the span of time during which the data is studied, regardless of whether data so gathered represents a set of discrete events having arbitrary timing within the interval, or whether the samples are explicitly bound to specified sub-intervals.

Practical implications

In practice, the continuous signal is sampled using an analog-to-digital converter (ADC), a non-ideal device with various physical limitations. This results in deviations from the theoretically perfect reconstruction capabilities, collectively referred to as distortion.

Various types of distortion can occur, including:

- Aliasing. A precondition of the sampling theorem is that the signal be bandlimited. However, in practice, no time-limited signal can be bandlimited. Since signals of interest are almost always time-limited (e.g., at most spanning the lifetime of the sampling device in question), it follows that they are not bandlimited. However, by designing a sampler with an appropriate guard band, it is possible to obtain output that is as accurate as necessary.
- Integration effect or aperture effect. This results from the fact that the sample is obtained as a time average within a sampling region, rather than just being equal to the signal value at the sampling instant. The integration effect is readily noticeable in photography when the exposure is too long and creates a blur in the image. An ideal camera would have an exposure time of zero. In a capacitor-based sample and hold circuit, the integration effect is introduced because the capacitor cannot instantly change voltage thus requiring the sample to have non-zero width.
- Jitter or deviation from the precise sample timing intervals.
- Noise, including thermal sensor noise, analog circuit noise, etc.
- Slew rate limit error, caused by an inability for an ADC output value to change sufficiently rapidly.
- Quantization as a consequence of the finite precision of words that represent the converted values.
- Error due to other non-linear effects of the mapping of input voltage to converted output value (in addition to the effects of quantization).

The conventional, practical digital-to-analog converter (DAC) does not output a sequence of dirac impulses (such that, if ideally low-pass filtered, result in the original signal before sampling) but instead output a sequence of piecewise constant values or rectangular pulses. This means that there is an inherent effect of the zero-order hold on the effective frequency response of the DAC resulting in a mild roll-off of gain at the higher frequencies (a 3.9224 dB loss at the Nyquist frequency). This zero-order hold effect is a consequence of the *hold* action of the DAC and is **not** due to the sample and hold that might precede a conventional ADC as is often misunderstood. The DAC can also suffer errors from jitter, noise, slewing, and non-linear mapping of input value to output voltage.

Jitter, noise, and quantization are often analyzed by modeling them as random errors added to the sample values. Integration and zero-order hold effects can be analyzed as a form of low-pass filtering. The non-linearities of either ADC or DAC are analyzed by replacing the ideal linear function mapping with a proposed nonlinear function.

Applications

Audio sampling

Digital audio uses pulse-code modulation and digital signals for sound reproduction. This includes analog-to-digital conversion (ADC), digital-to-analog conversion (DAC), storage, and transmission. In effect, the system commonly referred to as digital is in fact a discrete-time, discrete-level analog of a previous electrical analog. While modern systems can be quite subtle in their methods, the primary usefulness of a digital system is the ability to store, retrieve and transmit signals without any loss of quality.

Sampling rate

When it is necessary to capture audio covering the entire 20–20,000 Hz range of human hearing, such as when recording music or many types of acoustic events, audio waveforms are typically sampled at 44.1 kHz (CD), 48 kHz (professional audio), or 96 kHz. The approximately double-rate requirement is a consequence of the Nyquist theorem.

There has been an industry trend towards sampling rates well beyond the basic requirements; 96 kHz and even 192 kHz are available. This is in contrast with laboratory experiments, which have failed to show that ultrasonic frequencies are audible to human observers; however in some cases ultrasonic sounds do interact with and modulate the audible part of the frequency spectrum (intermodulation distortion). It is noteworthy that intermodulation distortion is not present in the live audio and so it represents an artificial coloration to the live sound.

One advantage of higher sampling rates is that they can relax the low-pass filter design requirements for ADCs and DACs, but with modern oversampling sigma-delta converters this advantage is less important.

Bit depth (quantization)

Audio is typically recorded at 8-, 16-, and 20-bit depth, which yield a theoretical maximum signal to quantization noise ratio (SQNR) for a pure sine wave of, approximately, 49.93 dB, 98.09 dB and 122.17 dB. Eight-bit audio is generally not used due to prominent and inherent quantization noise (low maximum SQNR), although the A-law and u-law 8-bit encodings pack more resolution into 8 bits while increase total harmonic distortion. CD quality audio is recorded at 16-bit. In practice, not many consumer stereos can produce more than about 90 dB of dynamic range, although some

can exceed 100 dB. Thermal noise limits the true number of bits that can be used in quantization. Few analog systems have signal to noise ratios (SNR) exceeding 120 dB; consequently, few situations will require more than 20-bit quantization.

For playback and not recording purposes, a proper analysis of typical programme levels throughout an audio system reveals that the capabilities of well-engineered 16-bit material far exceed those of the very best hi-fi systems, with the microphone noise and loudspeaker headroom being the real limiting factors.

Speech sampling

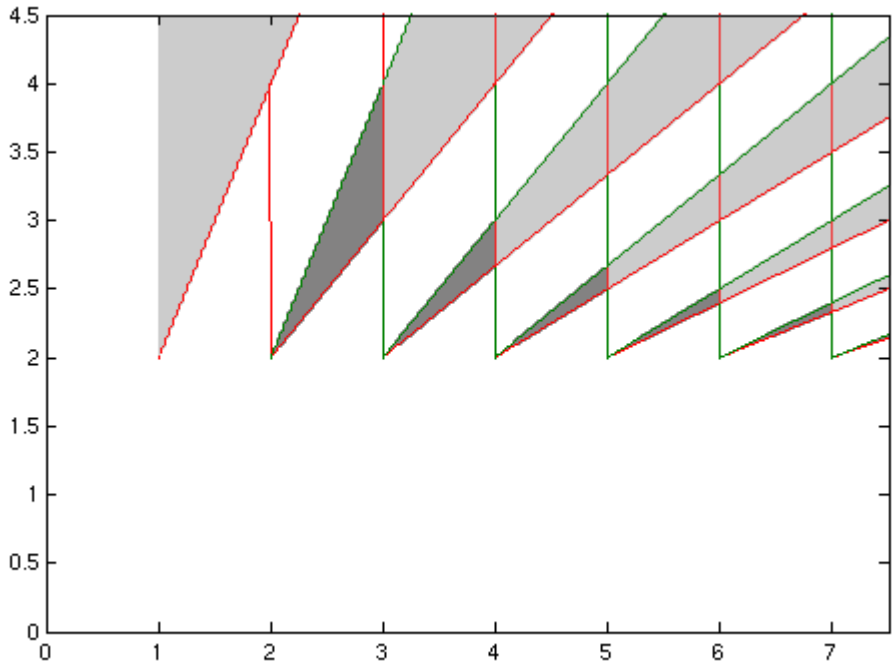
Speech signals, i.e., signals intended to carry only human speech, can usually be sampled at a much lower rate. For most phonemes, almost all of the energy is contained in the 5Hz-4 kHz range, allowing a sampling rate of 8 kHz. This is the sampling rate used by nearly all telephony systems, which use the G.711 sampling and quantization specifications.

Video sampling

Standard-definition television (SDTV) uses either 720 by 480 pixels (US NTSC 525-line) or 704 by 576 pixels (UK PAL 625-line) for the visible picture area.

High-definition television (HDTV) is currently moving towards three standards referred to as 720p (progressive), 1080i (interlaced) and 1080p (progressive, also known as Full-HD) which all 'HD-Ready' sets will be able to display.

Undersampling



Plot of sample rates (y axis) versus the upper edge frequency (x axis) for a band of width 1; grays areas are combinations that are "allowed" in the sense that no two frequencies in the band alias to same frequency. The darker gray areas correspond to undersampling with the lowest allowable sample rate.

When one samples a bandpass signal at a rate lower than the Nyquist rate, the samples are equal to samples of a low-frequency alias of the high-frequency signal; the original signal will still be uniquely represented and recoverable if the spectrum of its alias does not cross over half the sampling rate. Such undersampling is also known as *bandpass sampling*, *harmonic sampling*, *IF sampling*, and *direct IF to digital conversion.*

Oversampling

Oversampling is used in most modern analog-to-digital converters to reduce the distortion introduced by practical digital-to-analog converters, such as a zero-order hold instead of idealizations like the Whittaker–Shannon interpolation formula.

Complex sampling

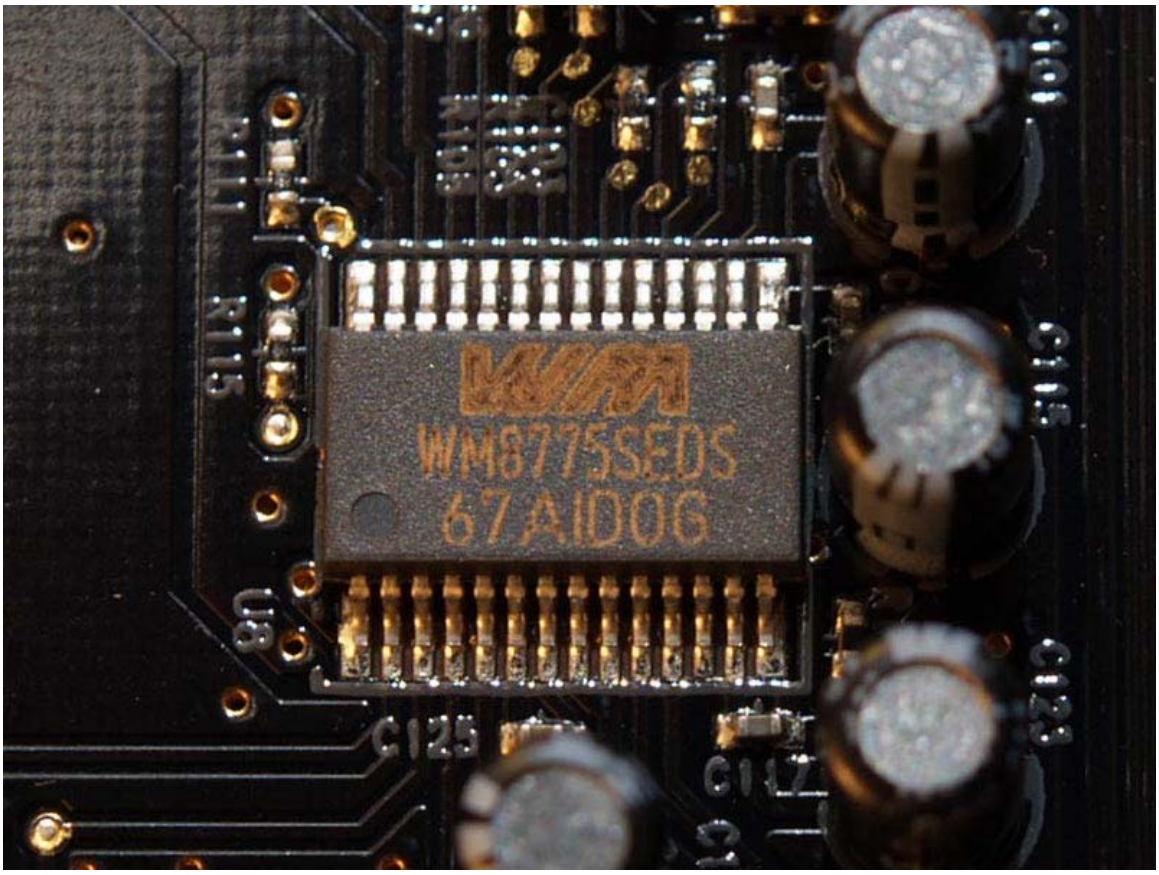
Complex sampling refers to the simultaneous sampling of two different, but related, waveforms, resulting in pairs of samples that are subsequently treated as complex numbers. Usually one waveform, $\hat{s}(t)$, is the Hilbert transform of the other

waveform, $s(t)$, and the complex-valued function, $s_a(t) \stackrel{\text{def}}{=} s(t) + j \cdot \hat{s}(t)$, is called an analytic signal, whose Fourier transform is zero for all negative values of frequency. In that case, the Nyquist rate for a waveform with no frequencies ≥ **B** can be reduced to just *B* (complex samples/sec), instead of *2B* (real samples/sec). More apparently, the equivalent baseband waveform, $s_a(t) \cdot e^{-j2\pi\frac{B}{2}t}$, also has a Nyquist rate of *B*, because all of its non-zero frequency content is shifted into the interval [-B/2, B/2).

Although complex-valued samples can be obtained as described above, they are much more commonly created by manipulating samples of a real-valued waveform. For instance, the equivalent baseband waveform can be created without explicitly computing $\hat{s}(t)$, by processing the product sequence, $\left[s(nT) \cdot e^{-j2\pi\frac{B}{2}Tn}\right]$, through a digital lowpass filter whose cutoff frequency is B/2. Computing only every other sample of the output sequence reduces the sample-rate commensurate with the reduced Nyquist rate. The result is half as many complex-valued samples as the original number of real samples. No information is lost, and the original s(t) waveform can be recovered, if necessary.

Chapter- 6

Analog-to-Digital Converter



4-channel stereo multiplexed analog-to-digital converter WM8775SEDS made by Wolfson Microelectronics placed on a X-Fi Fatal1ty Pro sound card.

An **analog-to-digital converter** (abbreviated **ADC**, **A/D** or **A to D**) is a device which converts a continuous quantity to a discrete digital number. The reverse operation is performed by a digital-to-analog converter (**DAC**).

Typically, an ADC is an electronic device that converts an input analog voltage (or current) to a digital number proportional to the magnitude of the voltage or current.

However, some non-electronic or only partially electronic devices, such as rotary encoders, can also be considered ADCs.

The digital output may use different coding schemes. Typically the digital output will be a two's complement binary number that is proportional to the input, but there are other possibilities. An encoder, for example, might output a Gray code.

An ADC may provide an isolated measurement. ADCs are also used in quantization of time-varying signals by turning them into a sequence of digital samples. The result is quantized in both time and value.

Concepts

Resolution

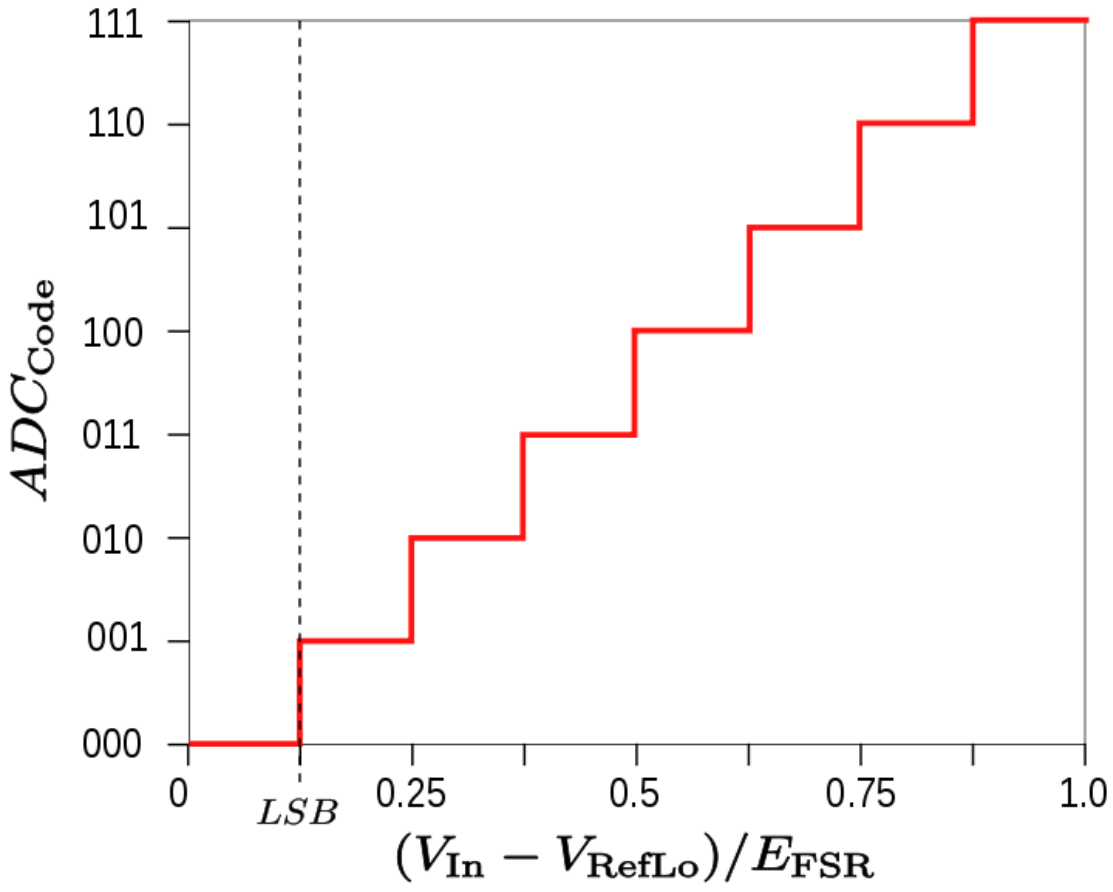


Fig. 1. An 8-level ADC coding scheme.

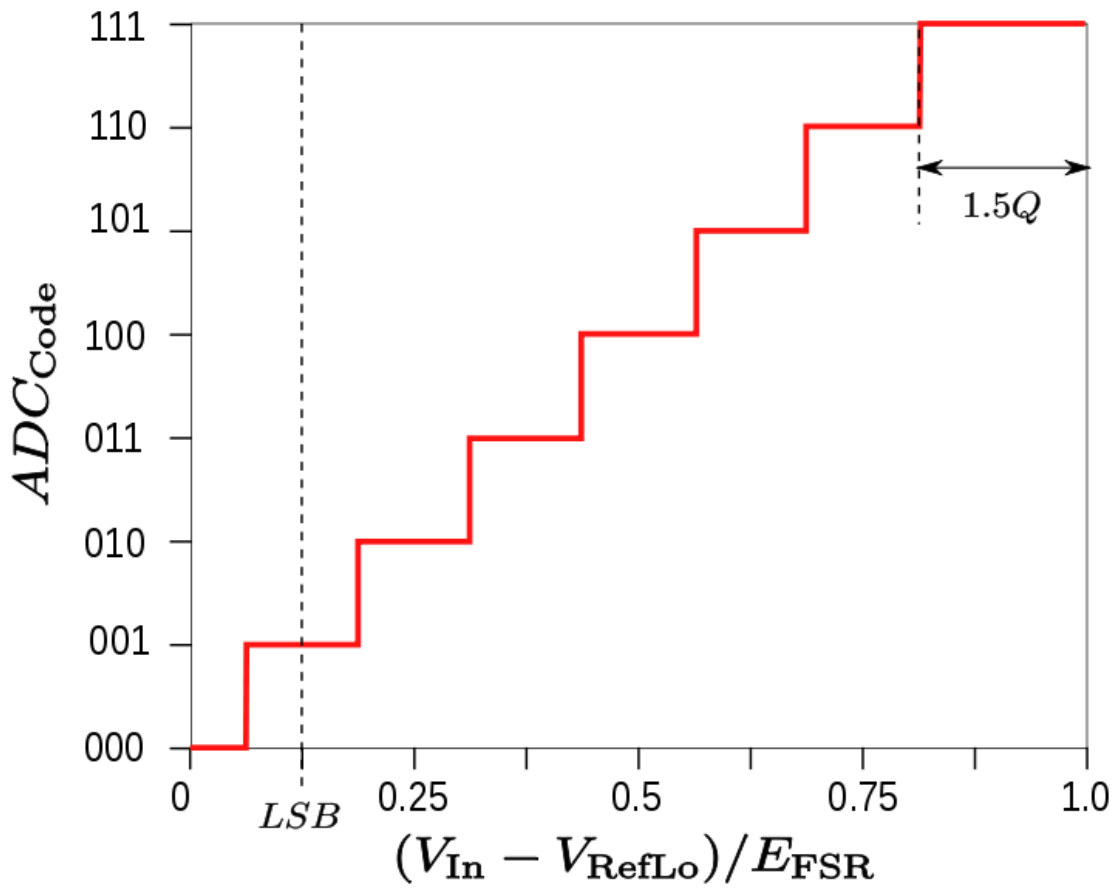


Fig. 2. An 8-level ADC coding scheme. As in figure 1 but with mid-tread coding.

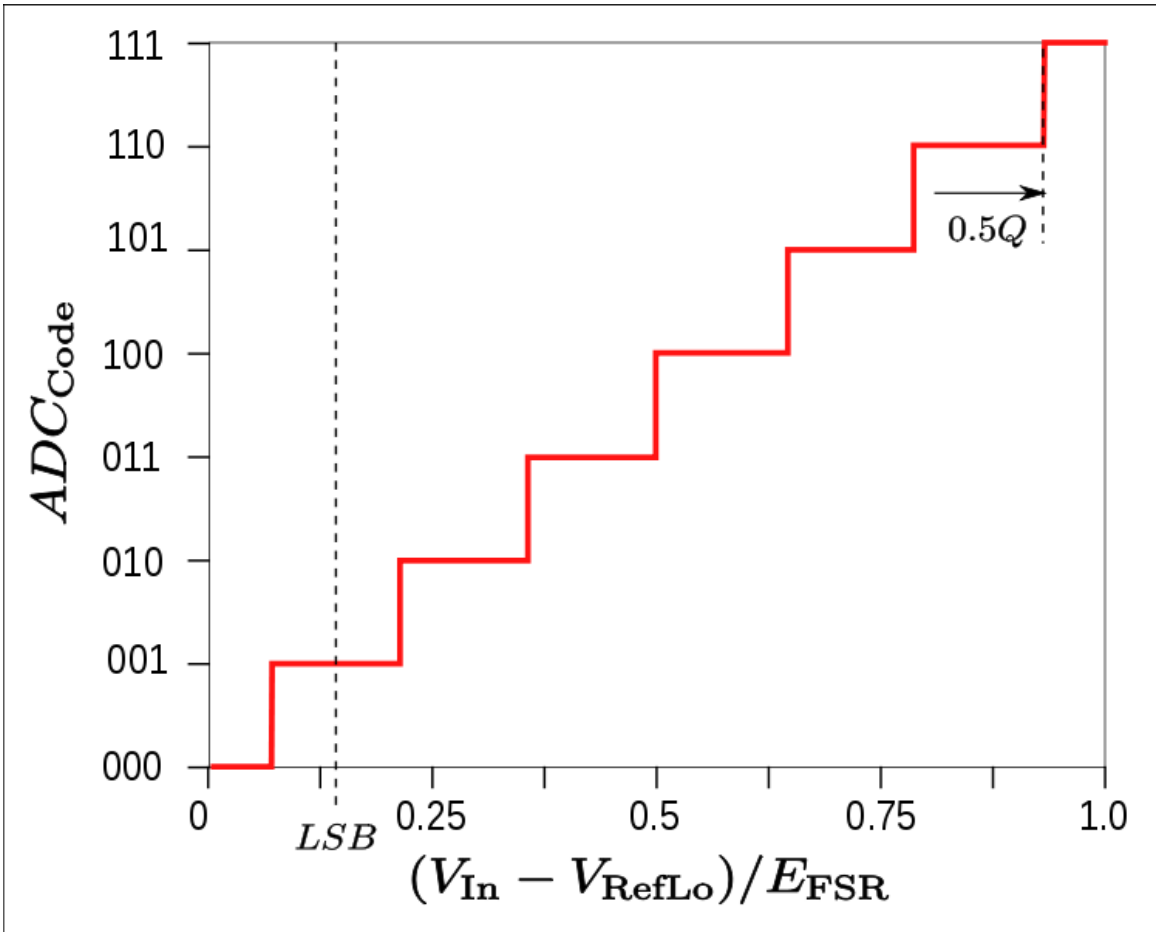


Fig. 3. An 8-level ADC mid-tread coding scheme. As in figure 2 but with equal half-*LSB* intervals at the highest and lowest codes. Note that *LSB* is now slightly larger than in figures 1 and 2.

The resolution of the converter indicates the number of discrete values it can produce over the range of analog values. The values are usually stored electronically in binary form, so the resolution is usually expressed in bits. In consequence, the number of discrete values available, or "levels", is usually a power of two. For example, an ADC with a resolution of 8 bits can encode an analog input to one in 256 different levels, since $2^8 = 256$. The values can represent the ranges from 0 to 255 (i.e. unsigned integer) or from −128 to 127 (i.e. signed integer), depending on the application.

Resolution can also be defined electrically, and expressed in volts. The minimum change in voltage required to guarantee a change in the output code level is called the *LSB* (least significant bit, since this is the voltage represented by a change in the LSB). The resolution Q of the ADC is equal to the *LSB* voltage. The voltage resolution of an ADC is equal to its overall voltage measurement range divided by the number of discrete voltage intervals:

$$Q = \frac{E_{\text{FSR}}}{N},$$

where N is the number of voltage intervals and E_{FSR} is the full scale voltage range. E_{FSR} is given by

$$E_{\mathrm{FSR}} = V_{\mathrm{RefHi}} - V_{\mathrm{RefLow}},$$

where V_{RefHi} and V_{RefLow} are the upper and lower extremes, respectively, of the voltages that can be coded.

Normally, the number of voltage intervals is given by

$$N = 2^M,$$

where M is the ADC's resolution in bits.

That is, one voltage interval is assigned per code level. However, figure 3 shows a situation where

$$N = 2^M - 1$$

Some examples:

- Example 1
  - Coding scheme as in figure 1
  - Full scale measurement range = 0 to 10 volts
  - ADC resolution is 12 bits: 2^{12} = 4096 quantization levels (codes)
  - ADC voltage resolution, Q = (10 V − 0 V) / 4096 = 10 V / 4096 ≈ 0.00244 V ≈ 2.44 mV.

- Example 2
  - Coding scheme as in figure 2
  - Full scale measurement range = -10 to +10 volts
  - ADC resolution is 14 bits: 2^{14} = 16384 quantization levels (codes)
  - ADC voltage resolution is, Q = (10 V − (−10 V)) / 16384 = 20 V / 16384 ≈ 0.00122 V ≈ 1.22 mV.

- Example 3
  - Coding scheme as in figure 3
  - Full scale measurement range = 0 to 7 volts
  - ADC resolution is 3 bits: 2^3 = 8 quantization levels (codes)
  - ADC voltage resolution is, Q = (7 V − 0 V)/7 = 7 V/7 = 1 V = 1000 mV

In most ADCs, the smallest output code ("0" in an unsigned system) represents a voltage range which is 0.5Q, that is, half the ADC voltage resolution (Q). The largest code represents a range of 1.5Q as in figure 2 (if this were 0.5Q also, the result would be as figure 3). The other $N − 2$ codes are all equal in width and represent the ADC voltage resolution (Q) calculated above. Doing this centers the code on an input voltage that

represents the M th division of the input voltage range. This practice is called "mid-tread" operation. This type of ADC can be modeled mathematically as:

$$ADC_{\mathrm{Code}} = \mathrm{round}\left(\left(\frac{2^M}{V_{\mathrm{RefHi}} - V_{\mathrm{RefLow}}}\right) \cdot (V_{\mathrm{In}} - V_{\mathrm{RefLow}})\right)$$

The exception to this convention seems to be the Microchip PIC processor, where all M steps are equal width, as shown in figure 1. This practice is called "Mid-Rise with Offset" operation.

$$ADC_{\mathrm{Code}} = \mathrm{floor}\left(\left(\frac{2^M}{V_{\mathrm{RefHi}} - V_{\mathrm{RefLow}}}\right) \cdot (V_{\mathrm{In}} - V_{\mathrm{RefLow}})\right)$$

In practice, the useful resolution of a converter is limited by the best signal-to-noise ratio (SNR) that can be achieved for a digitized signal. An ADC can resolve a signal to only a certain number of bits of resolution, called the effective number of bits (ENOB). One effective bit of resolution changes the signal-to-noise ratio of the digitized signal by 6 dB, if the resolution is limited by the ADC. If a preamplifier has been used prior to A/D conversion, the noise introduced by the amplifier can be an important contributing factor towards the overall SNR.

Response type

Linear ADCs

Most ADCs are of a type known as linear The term *linear* implies here the range of the input values that map to each output value has a linear relationship with the output value, i.e., that the output value k is used for the range of input values from

$$m(k + b)$$

to

$$m(k + 1 + b),$$

where m and b are constants. Here b is typically 0 or −0.5. When $b = 0$, the ADC is referred to as *mid-rise*, and when $b = -0.5$ it is referred to as *mid-tread*.

Non-linear ADCs

If the probability density function of a signal being digitized is uniform, then the signal-to-noise ratio relative to the quantization noise is the best possible. Because this is often not the case, it is usual to pass the signal through its cumulative distribution function (CDF) before the quantization. This is good because the regions that are more important

get quantized with a better resolution. In the dequantization process, the inverse CDF is needed.

This is the same principle behind the companders used in some tape-recorders and other communication systems, and is related to entropy maximization.

For example, a voice signal has a Laplacian distribution. This means that the region around the lowest levels, near 0, carries more information than the regions with higher amplitudes. Because of this, logarithmic ADCs are very common in voice communication systems to increase the dynamic range of the representable values while retaining fine-granular fidelity in the low-amplitude region.

An eight-bit A-law or the μ-law logarithmic ADC covers the wide dynamic range and has a high resolution in the critical low-amplitude region, that would otherwise require a 12-bit linear ADC.

Accuracy

An ADC has several sources of errors. Quantization error and (assuming the ADC is intended to be linear) non-linearity are intrinsic to any analog-to-digital conversion. There is also a so-called *aperture error* which is due to a clock jitter and is revealed when digitizing a time-variant signal (not a constant value).

These errors are measured in a unit called the *LSB*, which is an abbreviation for least significant bit. In the above example of an eight-bit ADC, an error of one LSB is 1/256 of the full signal range, or about 0.4%.

Quantization error

Quantization error (or quantization noise) is the difference between the original signal and the digitized signal. Hence, The magnitude of the quantization error at the sampling instant is between zero and half of one LSB. Quantization error is due to the finite resolution of the digital representation of the signal, and is an unavoidable imperfection in all types of ADCs.

Non-linearity

All ADCs suffer from non-linearity errors caused by their physical imperfections, causing their output to deviate from a linear function (or some other function, in the case of a deliberately non-linear ADC) of their input. These errors can sometimes be mitigated by calibration, or prevented by testing.

Important parameters for linearity are integral non-linearity (INL) and differential non-linearity (DNL). These non-linearities reduce the dynamic range of the signals that can be digitized by the ADC, also reducing the effective resolution of the ADC.

Aperture error

Imagine that we are digitizing a sine wave $x(t) = A\sin(2\pi f_0 t)$. Provided that the actual sampling time *uncertainty* due to the *clock jitter* is Δt, the error caused by this phenomenon can be estimated as $E_{ap} \leq |x'(t)\Delta t| \leq 2A\pi f_0 \Delta t$.

The error is zero for DC, small at low frequencies, but significant when high frequencies have high amplitudes. This effect can be ignored if it is drowned out by the *quantizing error*. Jitter requirements can be calculated using the following formula: $\Delta t < \frac{1}{2^q \pi f_0}$, where q is a number of ADC bits.

ADC resolution in bit	input frequency						
	1 Hz	**44.1 kHz**	**192 kHz**	**1 MHz**	**10 MHz**	**100 MHz**	**1 GHz**
8	1243 µs	28.2 ns	6.48 ns	1.24 ns	124 ps	12.4 ps	1.24 ps
10	311 µs	7.05 ns	1.62 ns	311 ps	31.1 ps	3.11 ps	0.31 ps
12	77.7 µs	1.76 ns	405 ps	77.7 ps	7.77 ps	0.78 ps	0.08 ps
14	19.4 µs	441 ps	101 ps	19.4 ps	1.94 ps	0.19 ps	0.02 ps
16	4.86 µs	110 ps	25.3 ps	4.86 ps	0.49 ps	0.05 ps	–
18	1.21 µs	27.5 ps	6.32 ps	1.21 ps	0.12 ps	–	–
20	304 ns	6.88 ps	1.58 ps	0.16 ps	–	–	–
24	19.0 ns	0.43 ps	0.10 ps	–	–	–	–
32	74.1 ps	–	–	–	–	–	–

This table shows, for example, that it is not worth using a precise 24-bit ADC for sound recording if there is not an *ultra low jitter* clock. One should consider taking this phenomenon into account before choosing an ADC.

Clock jitter is caused by phase noise. The resolution of ADCs with a digitization bandwidth between 1 MHz and 1 GHz is limited by jitter.

When sampling audio signals at 44.1 kHz, the anti-aliasing filter should have eliminated all frequencies above 22 kHz. The input frequency (in this case, 22 kHz), not the ADC clock frequency, is the determining factor with respect to jitter performance.

Sampling rate

The analog signal is continuous in time and it is necessary to convert this to a flow of digital values. It is therefore required to define the rate at which new digital values are sampled from the analog signal. The rate of new values is called the *sampling rate* or *sampling frequency* of the converter.

A continuously varying bandlimited signal can be sampled (that is, the signal values at intervals of time T, the sampling time, are measured and stored) and then the original signal can be *exactly* reproduced from the discrete-time values by an interpolation formula. The accuracy is limited by quantization error. However, this faithful reproduction is only possible if the sampling rate is higher than twice the highest frequency of the signal. This is essentially what is embodied in the Shannon-Nyquist sampling theorem.

Since a practical ADC cannot make an instantaneous conversion, the input value must necessarily be held constant during the time that the converter performs a conversion (called the *conversion time*). An input circuit called a sample and hold performs this task—in most cases by using a capacitor to store the analog voltage at the input, and using an electronic switch or gate to disconnect the capacitor from the input. Many ADC integrated circuits include the sample and hold subsystem internally.

Aliasing

All ADCs work by sampling their input at discrete intervals of time. Their output is therefore an incomplete picture of the behaviour of the input. There is no way of knowing, by looking at the output, what the input was doing between one sampling instant and the next. If the input is known to be changing slowly compared to the sampling rate, then it can be assumed that the value of the signal between two sample instants was somewhere between the two sampled values. If, however, the input signal is changing rapidly compared to the sample rate, then this assumption is not valid.

If the digital values produced by the ADC are, at some later stage in the system, converted back to analog values by a digital to analog converter or DAC, it is desirable that the output of the DAC be a faithful representation of the original signal. If the input signal is changing much faster than the sample rate, then this will not be the case, and spurious signals called *aliases* will be produced at the output of the DAC. The frequency of the aliased signal is the difference between the signal frequency and the sampling rate. For example, a 2 kHz sine wave being sampled at 1.5 kHz would be reconstructed as a 500 Hz sine wave. This problem is called *aliasing*.

To avoid aliasing, the input to an ADC must be low-pass filtered to remove frequencies above half the sampling rate. This filter is called an *anti-aliasing* filter, and is essential for a practical ADC system that is applied to analog signals with higher frequency content.

Although aliasing in most systems is unwanted, it should also be noted that it can be exploited to provide simultaneous down-mixing of a band-limited high frequency signal.

Dither

In A-to-D converters, performance can usually be improved using dither. This is a very small amount of random noise (white noise) which is added to the input before

conversion. Its amplitude is set to be twice the value of the least significant bit. Its effect is to cause the state of the LSB to randomly oscillate between 0 and 1 in the presence of very low levels of input, rather than sticking at a fixed value. Rather than the signal simply getting cut off altogether at this low level (which is only being quantized to a resolution of 1 bit), it extends the effective range of signals that the A-to-D converter can convert, at the expense of a slight increase in noise - effectively the quantization error is diffused across a series of noise values which is far less objectionable than a hard cutoff. The result is an accurate representation of the signal over time. A suitable filter at the output of the system can thus recover this small signal variation.

An audio signal of very low level (with respect to the bit depth of the ADC) sampled without dither sounds extremely distorted and unpleasant. Without dither the low level may cause the least significant bit to "stick" at 0 or 1. With dithering, the true level of the audio may be calculated by averaging the actual quantized sample with a series of other samples [the dither] that are recorded over time.

A virtually identical process, also called dither or dithering, is often used when quantizing photographic images to a fewer number of bits per pixel—the image becomes noisier but to the eye looks far more realistic than the quantized image, which otherwise becomes banded. This analogous process may help to visualize the effect of dither on an analogue audio signal that is converted to digital.

Dithering is also used in integrating systems such as electricity meters. Since the values are added together, the dithering produces results that are more exact than the LSB of the analog-to-digital converter.

Note that dither can only increase the resolution of a sampler, it cannot improve the linearity, and thus accuracy does not necessarily improve.

Oversampling

Usually, signals are sampled at the minimum rate required, for economy, with the result that the quantization noise introduced is white noise spread over the whole pass band of the converter. If a signal is sampled at a rate much higher than the Nyquist frequency and then digitally filtered to limit it to the signal bandwidth there are the following advantages:

- digital filters can have better properties (sharper rolloff, phase) than analogue filters, so a sharper anti-aliasing filter can be realised and then the signal can be downsampled giving a better result
- a 20-bit ADC can be made to act as a 24-bit ADC with 256× oversampling
- the signal-to-noise ratio due to quantization noise will be higher than if the whole available band had been used. With this technique, it is possible to obtain an effective resolution larger than that provided by the converter alone
- The improvement in SNR is 3 dB (equivalent to 0.5 bits) per octave of oversampling which is not sufficient for many applications. Therefore,

oversampling is usually coupled with noise shaping. With noise shaping, the improvement is 6L+3 dB per octave where L is the order of loop filter used for noise shaping. e.g. - a 2nd order loop filter will provide an improvement of 15 dB/octave.

Relative speed and precision

The speed of an ADC varies by type. The Wilkinson ADC is limited by the clock rate which is processable by current digital circuits. Currently, frequencies up to 300 MHz are possible. The conversion time is directly proportional to the number of channels. For a successive approximation ADC, the conversion time scales with the logarithm of the number of channels. Thus for a large number of channels, it is possible that the successive approximation ADC is faster than the Wilkinson. However, the time consuming steps in the Wilkinson are digital, while those in the successive approximation are analog. Since analog is inherently slower than digital, as the number of channels increases, the time required also increases. Thus there are competing processes at work. Flash ADCs are certainly the fastest type of the three. The conversion is basically performed in a single parallel step. For an 8-bit unit, conversion takes place in a few tens of nanoseconds.

There is, as expected, somewhat of a trade off between speed and precision. Flash ADCs have drifts and uncertainties associated with the comparator levels, which lead to poor uniformity in channel width. Flash ADCs have a resulting poor linearity. For successive approximation ADCs, poor linearity is also apparent, but less so than for flash ADCs. Here, non-linearity arises from accumulating errors from the subtraction processes. Wilkinson ADCs are the best of the three. These have the best differential non-linearity. The other types require channel smoothing in order to achieve the level of the Wilkinson.

The sliding scale principle

The sliding scale or randomizing method can be employed to greatly improve the channel width uniformity and differential linearity of any type of ADC, but especially flash and successive approximation ADCs. Under normal conditions, a pulse of a particular amplitude is always converted to a certain channel number. The problem lies in that channels are not always of uniform width, and the differential linearity decreases proportionally with the divergence from the average width. The sliding scale principle uses an averaging effect to overcome this phenomenon. A random, but known analog voltage is added to the input pulse. It is then converted to digital form, and the equivalent digital version is subtracted, thus restoring it to its original value. The advantage is that the conversion has taken place at a random point. The statistical distribution of the final channel numbers is decided by a weighted average over a region of the range of the ADC. This in turn desensitizes it to the width of any given channel.

ADC structures

These are the most common ways of implementing an electronic ADC:

- A **direct conversion ADC** or **flash ADC** has a bank of comparators sampling the input signal in parallel, each firing for their decoded voltage range. The comparator bank feeds a logic circuit that generates a code for each voltage range. Direct conversion is very fast, capable of gigahertz sampling rates, but usually has only 8 bits of resolution or fewer, since the number of comparators needed, $2^N - 1$, doubles with each additional bit, requiring a large expensive circuit. ADCs of this type have a large die size, a high input capacitance, high power dissipation, and are prone to produce glitches on the output (by outputting an out-of-sequence code). Scaling to newer submicrometre technologies does not help as the device mismatch is the dominant design limitation. They are often used for video, wideband communications or other fast signals in optical storage.

- A **successive-approximation ADC** uses a comparator to reject ranges of voltages, eventually settling on a final voltage range. Successive approximation works by constantly comparing the input voltage to the output of an internal digital to analog converter (DAC, fed by the current value of the approximation) until the best approximation is achieved. At each step in this process, a binary value of the approximation is stored in a successive approximation register (SAR). The SAR uses a reference voltage (which is the largest signal the ADC is to convert) for comparisons. For example if the input voltage is 60 V and the reference voltage is 100 V, in the 1st clock cycle, 60 V is compared to 50 V (the reference, divided by two. This is the voltage at the output of the internal DAC when the input is a '1' followed by zeros), and the voltage from the comparator is positive (or '1') (because 60 V is greater than 50 V). At this point the first binary digit (MSB) is set to a '1'. In the 2nd clock cycle the input voltage is compared to 75 V (being halfway between 100 and 50 V: This is the output of the internal DAC when its input is '11' followed by zeros) because 60 V is less than 75 V, the comparator output is now negative (or '0'). The second binary digit is therefore set to a '0'. In the 3rd clock cycle, the input voltage is compared with 62.5 V (halfway between 50 V and 75 V: This is the output of the internal DAC when its input is '101' followed by zeros). The output of the comparator is negative or '0' (because 60 V is less than 62.5 V) so the third binary digit is set to a 0. The fourth clock cycle similarly results in the fourth digit being a '1' (60 V is greater than 56.25 V, the DAC output for '1001' followed by zeros). The result of this would be in the binary form 1001. This is also called *bit-weighting conversion*, and is similar to a binary search. The analogue value is rounded to the nearest binary value below, meaning this converter type is mid-rise (see above). Because the approximations are successive (not simultaneous), the conversion takes one clock-cycle for each bit of resolution desired. The clock frequency must be equal to the sampling frequency multiplied by the number of bits of resolution desired. For example, to sample audio at 44.1 kHz with 32 bit resolution, a clock frequency of over 1.4 MHz would be required. ADCs of this type have good resolutions and quite wide ranges. They are more complex than some other designs.

- A **ramp-compare ADC** produces a saw-tooth signal that ramps up or down then quickly returns to zero. When the ramp starts, a timer starts counting. When the

ramp voltage matches the input, a comparator fires, and the timer's value is recorded. Timed ramp converters require the least number of transistors. The ramp time is sensitive to temperature because the circuit generating the ramp is often just some simple oscillator. There are two solutions: use a clocked counter driving a DAC and then use the comparator to preserve the counter's value, or calibrate the timed ramp. A special advantage of the ramp-compare system is that comparing a second signal just requires another comparator, and another register to store the voltage value. A very simple (non-linear) ramp-converter can be implemented with a microcontroller and one resistor and capacitor. Vice versa, a filled capacitor can be taken from an integrator, time-to-amplitude converter, phase detector, sample and hold circuit, or peak and hold circuit and discharged. This has the advantage that a slow comparator cannot be disturbed by fast input changes.

- The **Wilkinson ADC** was designed by D. H. Wilkinson in 1950. The Wilkinson ADC is based on the comparison of an input voltage with that produced by a charging capacitor. The capacitor is allowed to charge until its voltage is equal to the amplitude of the input pulse. (A comparator determines when this condition has been reached.) Then, the capacitor is allowed to discharge linearly, which produces a ramp voltage. At the point when the capacitor begins to discharge, a gate pulse is initiated. The gate pulse remains on until the capacitor is completely discharged. Thus the duration of the gate pulse is directly proportional to the amplitude of the input pulse. This gate pulse operates a linear gate which receives pulses from a high-frequency oscillator clock. While the gate is open, a discrete number of clock pulses pass through the linear gate and are counted by the address register. The time the linear gate is open is proportional to the amplitude of the input pulse, thus the number of clock pulses recorded in the address register is proportional also. Alternatively, the charging of the capacitor could be monitored, rather than the discharge.

- An **integrating ADC** (also **dual-slope** or **multi-slope** ADC) applies the unknown input voltage to the input of an integrator and allows the voltage to ramp for a fixed time period (the run-up period). Then a known reference voltage of opposite polarity is applied to the integrator and is allowed to ramp until the integrator output returns to zero (the run-down period). The input voltage is computed as a function of the reference voltage, the constant run-up time period, and the measured run-down time period. The run-down time measurement is usually made in units of the converter's clock, so longer integration times allow for higher resolutions. Likewise, the speed of the converter can be improved by sacrificing resolution. Converters of this type (or variations on the concept) are used in most digital voltmeters for their linearity and flexibility.

- A **delta-encoded ADC** or Counter-ramp has an up-down counter that feeds a digital to analog converter (DAC). The input signal and the DAC both go to a comparator. The comparator controls the counter. The circuit uses negative feedback from the comparator to adjust the counter until the DAC's output is close

enough to the input signal. The number is read from the counter. Delta converters have very wide ranges, and high resolution, but the conversion time is dependent on the input signal level, though it will always have a guaranteed worst-case. Delta converters are often very good choices to read real-world signals. Most signals from physical systems do not change abruptly. Some converters combine the delta and successive approximation approaches; this works especially well when high frequencies are known to be small in magnitude.

- A **pipeline ADC** (also called **subranging quantizer**) uses two or more steps of subranging. First, a coarse conversion is done. In a second step, the difference to the input signal is determined with a digital to analog converter (DAC). This difference is then converted finer, and the results are combined in a last step. This can be considered a refinement of the successive approximation ADC wherein the feedback reference signal consists of the interim conversion of a whole range of bits (for example, four bits) rather than just the next-most-significant bit. By combining the merits of the successive approximation and flash ADCs this type is fast, has a high resolution, and only requires a small die size.

- A **Sigma-Delta ADC** (also known as a Delta-Sigma ADC) oversamples the desired signal by a large factor and filters the desired signal band. Generally, a smaller number of bits than required are converted using a Flash ADC after the filter. The resulting signal, along with the error generated by the discrete levels of the Flash, is fed back and subtracted from the input to the filter. This negative feedback has the effect of noise shaping the error due to the Flash so that it does not appear in the desired signal frequencies. A digital filter (decimation filter) follows the ADC which reduces the sampling rate, filters off unwanted noise signal and increases the resolution of the output (sigma-delta modulation, also called delta-sigma modulation).

- A **Time-interleaved ADC** uses M parallel ADCs where each ADC sample data every M:th cycle of the effective sample clock. The result is that the sample rate is increased M times compared to what each individual ADC can manage. In practice, the individual differences between the M ADCs degrade the overall performance reducing the SFDR. However, technologies exist to correct for these time-interleaving mismatch errors.

- An **ADC with intermediate FM stage** first uses a voltage-to-frequency converter to converts the desired signal into an oscillating signal with a frequency proportional to the voltage of the desired signal, and then uses a frequency counter to convert that frequency into a digital count proportional to the desired signal voltage. Longer integration times allow for higher resolutions. Likewise, the speed of the converter can be improved by sacrificing resolution. The two parts of the ADC may be widely separated, with the frequency signal passed through a opto-isolator or transmitted wirelessly. Some such ADCs use sine wave or square wave frequency modulation; others use pulse-frequency modulation.

Such ADCs were once the most popular way to show a digital display of the status of a remote analog sensor.

There can be other ADCs that use a combination of electronics and other technologies:

- A **Time-stretch analog-to-digital converter (TS-ADC)** digitizes a very wide bandwidth analog signal, that cannot be digitized by a conventional electronic ADC, by time-stretching the signal prior to digitization. It commonly uses a photonic preprocessor frontend to time-stretch the signal, which effectively slows the signal down in time and compresses its bandwidth. As a result, an electronic backend ADC, that would have been too slow to capture the original signal, can now capture this slowed down signal. For continuous capture of the signal, the frontend also divides the signal into multiple segments in addition to time-stretching. Each segment is individually digitized by a separate electronic ADC. Finally, a digital signal processor rearranges the samples and removes any distortions added by the frontend to yield the binary data that is the digital representation of the original analog signal.

Commercial analog-to-digital converters

These are usually integrated circuits.

Most converters sample with 6 to 24 bits of resolution, and produce fewer than 1 megasample per second. Thermal noise generated by passive components such as resistors masks the measurement when higher resolution is desired. For audio applications and in room temperatures, such noise is usually a little less than 1 μV (microvolt) of white noise. If the Most Significant Bit corresponds to a standard 2 volts of output signal, this translates to a noise-limited performance that is less than 20~21 bits, and obviates the need for any dithering. Mega- and gigasample per second converters are available, though (Feb 2002). Megasample converters are required in digital video cameras, video capture cards, and TV tuner cards to convert full-speed analog video to digital video files. Commercial converters usually have ±0.5 to ±1.5 LSB error in their output.

In many cases the most expensive part of an integrated circuit is the pins, because they make the package larger, and each pin has to be connected to the integrated circuit's silicon. To save pins, it is common for slow ADCs to send their data one bit at a time over a serial interface to the computer, with the next bit coming out when a clock signal changes state, say from zero to 5 V. This saves quite a few pins on the ADC package, and in many cases, does not make the overall design any more complex (even microprocessors which use memory-mapped I/O only need a few bits of a port to implement a serial bus to an ADC).

Commercial ADCs often have several inputs that feed the same converter, usually through an analog multiplexer. Different models of ADC may include sample and hold

circuits, instrumentation amplifiers or differential inputs, where the quantity measured is the difference between two voltages.

Applications

Application to music recording

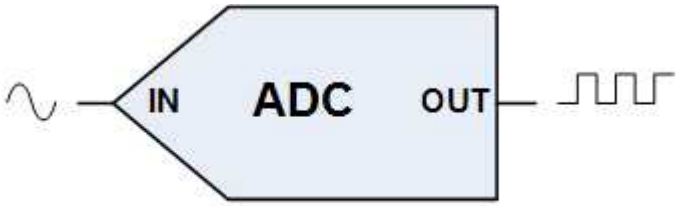
ADCs are integral to current music reproduction technology. Since much music production is done on computers, when an analog recording is used, an ADC is needed to create the PCM data stream that goes onto a compact disc or digital music file.

The current crop of AD converters utilized in music can sample at rates up to 192 kilohertz. High bandwidth headroom allows the use of cheaper or faster anti-aliasing filters of less severe filtering slopes. The proponents of oversampling assert that such shallower anti-aliasing filters produce less deleterious effects on sound quality, exactly because of their gentler slopes. Others prefer entirely filterless AD conversion, arguing that aliasing is less detrimental to sound perception than pre-conversion brickwall filtering. Considerable literature exists on these matters, but commercial considerations often play a significant role. Most high-profile recording studios record in 24-bit/192-176.4 kHz PCM or in DSD formats, and then downsample or decimate the signal for Red-Book CD production (44.1 kHz or at 48 kHz for commonly used for radio/TV broadcast applications).

Digital Signal Processing

AD converters are used virtually everywhere where an analog signal has to be processed, stored, or transported in digital form. Fast video ADCs are used, for example, in TV tuner cards. Slow on-chip 8, 10, 12, or 16 bit ADCs are common in microcontrollers. Very fast ADCs are needed in digital oscilloscopes, and are crucial for new applications like software defined radio.

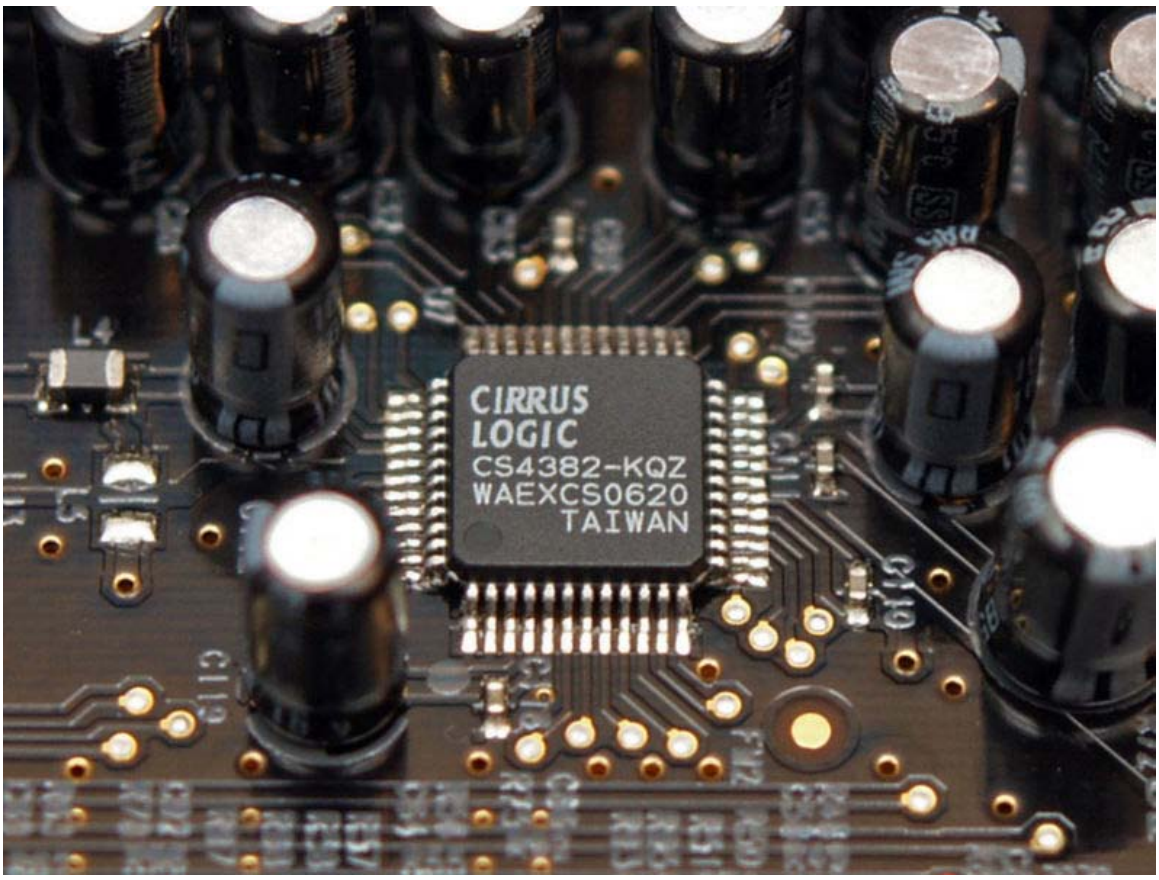
Electrical Symbol



ELECTRICAL SYMBOL FOR ANALOG TO DIGITAL CONVERTER (ADC)

Chapter- 7

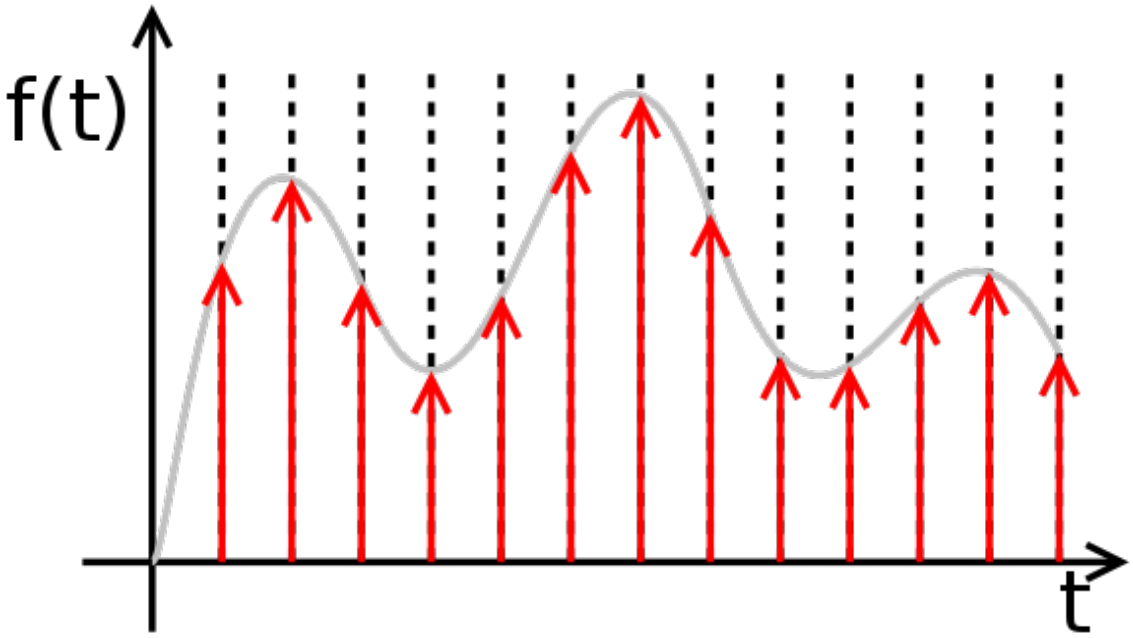
Digital-to-Analog Converter



8-channel digital-to-analog converter Cirrus Logic CS4382 as used in a soundcard.

In electronics, a **digital-to-analog converter** (**DAC** or **D-to-A**) is a device that converts a digital (usually binary) code to an analog signal (current, voltage, or electric charge). An analog-to-digital converter (ADC) performs the reverse operation.

Basic ideal operation



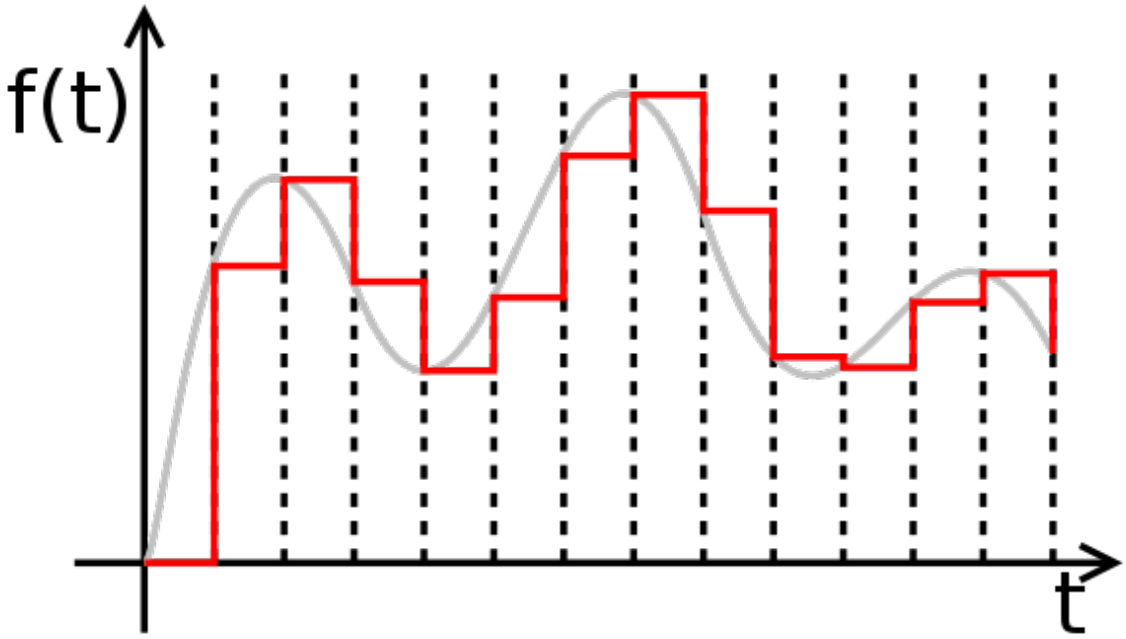
Ideally sampled signal.

A DAC converts an abstract finite-precision number (usually a fixed-point binary number) into a concrete physical quantity (e.g., a voltage or a pressure). In particular, DACs are often used to convert finite-precision time series data to a continually varying physical signal.

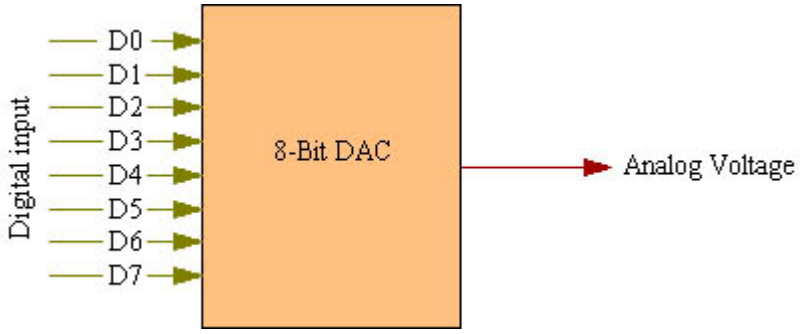
A typical DAC converts the abstract numbers into a concrete sequence of impulses that are then processed by a reconstruction filter using some form of interpolation to fill in data between the impulses. Other DAC methods (e.g., methods based on Delta-sigma modulation) produce a pulse-density modulated signal that can then be filtered in a similar way to produce a smoothly varying signal.

By the Nyquist–Shannon sampling theorem, sampled data can be reconstructed perfectly provided that its bandwidth meets certain requirements (e.g., a baseband signal with bandwidth less than the Nyquist frequency; BUT requires an infinite number of samples. The finite number used in real life cause other problems especially with the D/A reconstruction of the original signal. However, even with an ideal reconstruction filter, digital sampling introduces quantization error that makes perfect reconstruction practically impossible. Increasing the digital resolution (i.e., increasing the number of bits used in each sample) or introducing sampling dither can reduce this error.

Practical operation



Piecewise constant output of a conventional practical DAC.



A simplified functional diagram of an 8-bit DAC

Instead of impulses, usually the sequence of numbers update the analogue voltage at uniform sampling intervals.

These numbers are written to the DAC, typically with a clock signal that causes each number to be latched in sequence, at which time the DAC output voltage changes rapidly from the previous value to the value represented by the currently latched number. The effect of this is that the output voltage is *held* in time at the current value until the next input number is latched resulting in a piecewise constant or 'staircase' shaped output. This is equivalent to a zero-order hold operation and has an effect on the frequency response of the reconstructed signal.

The fact that DACs output a sequence of piecewise constant values (known as zero-order hold in sample data textbooks) or rectangular pulses causes multiple harmonics above the

Nyquist frequency. Usually, these are removed with a low pass filter acting as a reconstruction filter in applications that require it.

Applications

Audio

Top-loading CD player and external digital-to-analog converter.

Most modern audio signals are stored in digital form (for example MP3s and CDs) and in order to be heard through speakers they must be converted into an analog signal. DACs are therefore found in CD players, digital music players, and PC sound cards.

Specialist standalone DACs can also be found in high-end hi-fi systems. These normally take the digital output of a compatible CD player or dedicated transport and convert the signal into an analog line-level output that can then be fed into an amplifier to drive speakers.

Similar digital-to-analog converters can be found in digital speakers such as USB speakers, and in sound cards.

VOIP (Voice over IP) Phone, Data transmission over the Internet is done digitally so in order for voice to be transmitted it must be converted to digital using an Analog-to-Digital Converter and be converted into analog again using a DAC so the voice it can be heard on the other end.

Video

Video signals from a digital source, such as a computer, must be converted to analog form if they are to be displayed on an analog monitor. As of 2007, analog inputs are more commonly used than digital, but this may change as flat panel displays with DVI and/or HDMI connections become more widespread. A video DAC is, however, incorporated in any digital video player with analog outputs. The DAC is usually integrated with some memory (RAM), which contains conversion tables for gamma correction, contrast and brightness, to make a device called a RAMDAC.

A device that is distantly related to the DAC is the digitally controlled potentiometer, used to control an analog signal digitally.

Mechanical

An unusual application of digital-to-analog conversion was the whiffletree electromechanical digital-to-analog convertor linkage in the IBM Selectric typewriter.

DAC types

The most common types of electronic DACs are:

- The pulse-width modulator, the simplest DAC type. A stable current or voltage is switched into a low-pass analog filter with a duration determined by the digital input code. This technique is often used for electric motor speed control, but has many other applications as well.
- Oversampling DACs or interpolating DACs such as the delta-sigma DAC, use a pulse density conversion technique. The oversampling technique allows for the use of a lower resolution DAC internally. A simple 1-bit DAC is often chosen because the oversampled result is inherently linear. The DAC is driven with a pulse-density modulated signal, created with the use of a low-pass filter, step nonlinearity (the actual 1-bit DAC), and negative feedback loop, in a technique called delta-sigma modulation. This results in an effective high-pass filter acting on the quantization (signal processing) noise, thus steering this noise out of the low frequencies of interest into the megahertz frequencies of little interest, which is called noise shaping. The quantization noise at these high frequencies is removed or greatly attenuated by use of an analog low-pass filter at the output (sometimes a simple RC low-pass circuit is sufficient). Most very high resolution DACs (greater than 16 bits) are of this type due to its high linearity and low cost. Higher oversampling rates can relax the specifications of the output low-pass filter and enable further suppression of quantization noise. Speeds of greater than

100 thousand samples per second (for example, 192 kHz) and resolutions of 24 bits are attainable with delta-sigma DACs. A short comparison with pulse-width modulation shows that a 1-bit DAC with a simple first-order integrator would have to run at 3 THz (which is physically unrealizable) to achieve 24 meaningful bits of resolution, requiring a higher-order low-pass filter in the noise-shaping loop. A single integrator is a low-pass filter with a frequency response inversely proportional to frequency and using one such integrator in the noise-shaping loop is a first order delta-sigma modulator. Multiple higher order topologies (such as MASH) are used to achieve higher degrees of noise-shaping with a stable topology.

- The binary-weighted DAC, which contains one resistor or current source for each bit of the DAC connected to a summing point. These precise voltages or currents sum to the correct output value. This is one of the fastest conversion methods but suffers from poor accuracy because of the high precision required for each individual voltage or current. Such high-precision resistors and current sources are expensive, so this type of converter is usually limited to 8-bit resolution or less.
- The R-2R ladder DAC which is a binary-weighted DAC that uses a repeating cascaded structure of resistor values R and 2R. This improves the precision due to the relative ease of producing equal valued-matched resistors (or current sources). However, wide converters perform slowly due to increasingly large RC-constants for each added R-2R link.
- The thermometer-coded DAC, which contains an equal resistor or current-source segment for each possible value of DAC output. An 8-bit thermometer DAC would have 255 segments, and a 16-bit thermometer DAC would have 65,535 segments. This is perhaps the fastest and highest precision DAC architecture but at the expense of high cost. Conversion speeds of >1 billion samples per second have been reached with this type of DAC.
- Hybrid DACs, which use a combination of the above techniques in a single converter. Most DAC integrated circuits are of this type due to the difficulty of getting low cost, high speed and high precision in one device.
    - The segmented DAC, which combines the thermometer-coded principle for the most significant bits and the binary-weighted principle for the least significant bits. In this way, a compromise is obtained between precision (by the use of the thermometer-coded principle) and number of resistors or current sources (by the use of the binary-weighted principle). The full binary-weighted design means 0% segmentation, the full thermometer-coded design means 100% segmentation.

DAC performance

DACs are very important to system performance. The most important characteristics of these devices are:

- **Resolution**: This is the number of possible output levels the DAC is designed to reproduce. This is usually stated as the number of bits it uses, which is the base

two logarithm of the number of levels. For instance a 1 bit DAC is designed to reproduce 2 (2^1) levels while an 8 bit DAC is designed for 256 (2^8) levels. Resolution is related to the **effective number of bits** (ENOB) which is a measurement of the actual resolution attained by the DAC.

- **Maximum sampling frequency**: This is a measurement of the maximum speed at which the DACs circuitry can operate and still produce the correct output. As stated in the Nyquist–Shannon sampling theorem, a signal must be sampled at over twice the frequency of the desired signal. For instance, to reproduce signals in all the audible spectrum, which includes frequencies of up to 20 kHz, it is necessary to use DACs that operate at over 40 kHz. The CD standard samples audio at 44.1 kHz, thus DACs of this frequency are often used. A common frequency in cheap computer sound cards is 48 kHz — many work at only this frequency, offering the use of other sample rates only through (often poor) internal resampling.
- **Monotonicity**: This refers to the ability of a DAC's analog output to move only in the direction that the digital input moves (i.e., if the input increases, the output doesn't dip before asserting the correct output.) This characteristic is very important for DACs used as a low frequency signal source or as a digitally programmable trim element.
- **THD+N**: This is a measurement of the distortion and noise introduced to the signal by the DAC. It is expressed as a percentage of the total power of unwanted harmonic distortion and noise that accompany the desired signal. This is a very important DAC characteristic for dynamic and small signal DAC applications.
- **Dynamic range**: This is a measurement of the difference between the largest and smallest signals the DAC can reproduce expressed in decibels. This is usually related to DAC resolution and noise floor.

Other measurements, such as phase distortion and jitter, can also be very important for some applications.

Bits	Color limit	Frequency	Examples
10		54 MHz	
12		54 MHz	Sony NS-575p
12	4,096 colors	108 MHz	
12		150 MHz	NeoDigits Helios X5000
12		216 MHz	Philips BDP9000 (Blu-ray)
12		297 MHz	Toshiba HD-XE1
12		216 MHz	Samsung BD-P1200 (Blu-ray)
14	16,384 colors	108 MHz	Pioneer Elite, Black Finish, DV79AVI
14		216 MHz	Marantz DV9600, Sony DVPNS9100ES
16		149 MHz	NeuNeo HVD108

DAC figures of merit

- Static performance:
  - Differential nonlinearity (DNL) shows how much two adjacent code analog values deviate from the ideal 1LSB step
  - Integral nonlinearity (INL) shows how much the DAC transfer characteristic deviates from an ideal one. That is, the ideal characteristic is usually a straight line; INL shows how much the actual voltage at a given code value differs from that line, in LSBs (1LSB steps).
  - Gain
  - Offset
  - Noise is ultimately limited by the thermal noise generated by passive components such as resistors. For audio applications and in room temperatures, such noise is usually a little less than 1 μV (microvolt) of white noise. This limits performance to less than 20~21 bits even in 24-bit DACs.
- Frequency domain performance
  - Spurious-free dynamic range (SFDR) indicates in dB the ratio between the powers of the converted main signal and the greatest undesired spur
  - Signal to noise and distortion ratio (SNDR) indicates in dB the ratio between the powers of the converted main signal and the sum of the noise and the generated harmonic spurs
  - i-th harmonic distortion (HDi) indicates the power of the i-th harmonic of the converted main signal
  - Total harmonic distortion (THD) is the sum of the powers of all HDi
  - If the maximum DNL error is less than 1 LSB, then D/A converter is guaranteed to be monotonic.

However, many monotonic converters may have a maximum DNL greater than 1 LSB.

- Time domain performance:
  - Glitch energy
  - Response un

Chapter- 8

Oversampling & Downsampling

Oversampling

In signal processing, **oversampling** is the process of sampling a signal with a sampling frequency significantly higher than twice the bandwidth or highest frequency of the signal being sampled. Oversampling helps avoid aliasing, improves resolution and reduces noise.

Oversampling factor

An oversampled signal is said to be oversampled by a factor of β, defined as

$$\beta \stackrel{\text{def}}{=} \frac{f_s}{2B}$$

or

$$f_s = 2\beta B.$$

where

- f_s is the sampling frequency
- B is the bandwidth or highest frequency of the signal; the Nyquist rate is $2B$.

Motivation

There are three main reasons for performing oversampling:

Anti-aliasing

It aids in anti-aliasing because realisable analog anti-aliasing filters are very difficult to implement with the sharp cutoff necessary to maximize use of the available bandwidth without exceeding the Nyquist limit. By increasing the bandwidth of the sampled signal, the anti-aliasing filter has less complexity and can be made less expensively by relaxing the requirements of the filter at the cost of a faster sampler. Once sampled, the signal can be digitally filtered and downsampled to the desired sampling frequency. In modern integrated circuit technology, digital filters are much easier to implement than comparable analog filters of high order.

Resolution

In practice, oversampling is implemented in order to achieve cheaper higher-resolution A/D and D/A conversion. For instance, to implement a 24-bit converter, it is sufficient to use a 20-bit converter that can run at 256 times the target sampling rate. Averaging a group of 256 consecutive 20-bit samples adds 4 bits to the resolution of the average, producing a single sample with 24-bit resolution. Number of samples required to get $\mathbf{n}$bits of additional data:

samples = 2^{2n}

The result in software from $\mathbf{n}$samples is then divided by 2^n:

$$\text{result} = \frac{\text{Oversample Data}}{2^n}$$

Note that this averaging is possible only if the signal contains perfect equally distributed noise (i.e. if the A/D is perfect and the signal's deviation from an A/D result step lies below the threshold, the conversion result will be as inaccurate as if it had been measured by the low-resolution core A/D and the oversampling benefits will not take effect).

Noise

If multiple samples are taken of the same quantity with a different (and uncorrelated) random noise added to each sample, then averaging *N* samples reduces the noise variance (or noise power) by a factor of 1/*N*. This means that the signal-to-noise-ratio improves by a factor of 4 (6 dB or one additional meaningful bit) if we oversample by a factor of 4 relative to the Nyquist rate (i.e. a β of 4) and low-pass filter.

Certain kinds of A/D converters known as delta-sigma converters produce disproportionately more quantization noise in the upper portion of their output spectrum. By running these converters at some multiple of the target sampling rate, and low-pass filtering the oversampled down to half the target sampling rate, it is possible to obtain a result with *less* noise than the average over the entire band of the converter. Delta-sigma

converters use a technique called noise shaping to move the quantization noise to the higher frequencies.

Example

For example, consider a signal with a bandwidth or highest frequency of B = 100 Hz. The sampling theorem states that sampling frequency would have to be greater than 200 Hz. Sampling at 200 Hz would result in $\beta = 1$. Sampling at four times that rate ($\beta = 4$) would result in a sampling rate of 800 Hz. This gives the anti-aliasing filter a transition band of 600 Hz ($(f_s - B) - B$ = (800 Hz−100 Hz) − 100 Hz = 600 Hz) instead of 0 Hz if the sampling frequency was virtually 200 Hz.

An anti-aliasing filter with a transition band of 600 Hz is much more realizable than that of 0 Hz (which would require a perfect filter). If the sampler went to eight times over then the transition band would increase to 1400 Hz, which means the anti-aliasing filter could be less expensive due to relaxation of the transition band requirements.

After being sampled at 800 Hz, the signal (ostensibly with a bandwidth of 400 Hz) could be digitally filtered to have a bandwidth of 100 Hz and then further downsampled to closer to 200 Hz.

Downsampling

In signal processing, **downsampling** (or "subsampling") is the process of reducing the sampling rate of a signal. This is usually done to reduce the data rate or the size of the data.

The downsampling factor (commonly denoted by M) is usually an integer or a rational fraction greater than unity. This factor multiplies the sampling time or, equivalently, divides the sampling rate. For example, if compact disc audio at 44,100Hz is downsampled to 22,050Hz before broadcasting over FM radio, the bit rate is reduced in half, from 1,411,200 bit/s to 705,600 bit/s, assuming that each sample retains its size of 16 bits. The audio was therefore downsampled by a factor of 2.

Maintaining the sampling theorem criterion

Since downsampling reduces the sampling rate, we must be careful to make sure the Shannon-Nyquist sampling theorem criterion is maintained. If the sampling theorem is not satisfied then the resulting digital signal will have aliasing. To ensure that the sampling theorem is satisfied, a low-pass filter is used as an anti-aliasing filter to reduce the bandwidth of the signal *before* the signal is downsampled; the overall process (low-pass filter, then downsample) is called decimation.

Note that the anti-aliasing filter must be a low-pass filter in downsampling. This is different from sampling a continuous signal, where either a low-pass filter or a band-pass filter may be used.

Remark: A bandpass signal, i.e. a band-limited signal whose minimum frequency is different from zero, can be downsampled avoiding superposition of the spectra if certain conditions are satisfied.

Downsampling process

Consider a discrete signal $f(k)$ on a radian frequency digital frequency range.

Downsampling by integer factor

Let M denote the downsampling factor.

1. Filter the signal to ensure that the sampling theorem is satisfied. This filter should, theoretically, be the sinc filter with frequency cutoff at $\frac{\pi}{M}$. Let the filtered signal be denoted $g(k)$.
2. Reduce the data by picking out every M^{th} sample: $h(k) = g(Mk)$. Data rate reduction occurs in this step.

The first step calls for the use of a perfect low-pass filter, which is not implementable for real-time signals. When choosing a realizable low-pass filter this will have to be considered along with the aliasing effects it will have. Realizable low-pass filters have a "skirt", where the response diminishes from near unity to near zero. So in practice the cutoff frequency is placed far enough below the theoretical cutoff that the filter's skirt is contained below the theoretical cutoff.

Downsampling by rational fraction

Let M/L denote the downsampling factor.

1. Upsample by a factor of L
2. Downsample by a factor of M

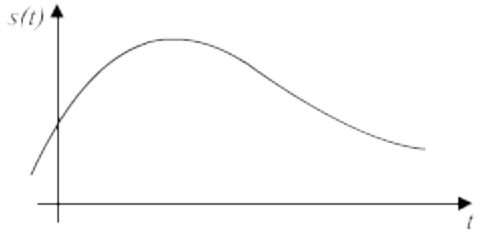
Note that a proper upsampling design requires an interpolation filter after increasing the data rate and that a proper downsampling design requires a filter before eliminating some samples. These two low-pass filters can be combined into a single filter.

Also note that these two steps are generally not reversible. Downsampling results in a loss of data and, if performed first, could result in data loss if there is any data filtered out by the downsampler's low-pass filter. Since both interpolation and anti-aliasing filters are low-pass filters, the filter with the smallest bandwidth is more restrictive and can

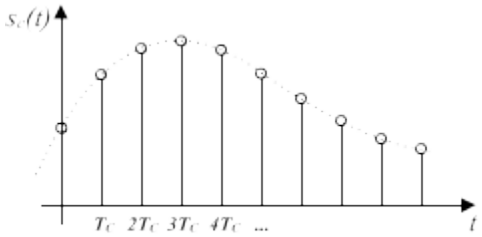
therefore be used in place of both filters. Since the rational fraction M/L is greater than unity then $L < M$ and the single low-pass filter should have cutoff at π / M.

Chapter- 9

Sampling Rate



Analog signal;



and resulting sampled signal.

The **sampling rate**, **sample rate**, or **sampling frequency** defines the number of samples per second (or per other unit) taken from a continuous signal to make a discrete signal. For time-domain signals, the unit for sampling rate is hertz (inverse seconds, 1/s, s^{-1}). The inverse of the sampling frequency is the **sampling period** or **sampling interval**, which is the time between samples.

Sample rate is usually noted in Sa/s (non-SI) and expanded as kSa/s, MSa/s, etc. The common notation for sampling frequency is f_s which stands for frequency (subscript) sampled.

Sampling theorem

The Nyquist–Shannon sampling theorem states that perfect reconstruction of a signal is possible when the sampling frequency is greater than twice the maximum frequency of the signal being sampled, or equivalently, when the Nyquist frequency (half the sample rate) exceeds the highest frequency of the signal being sampled. If lower sampling rates are used, the original signal's information may not be completely recoverable from the sampled signal. For example, if a signal has an upper band limit of 100 Hz, a sampling frequency greater than 200 Hz will avoid aliasing and allow theoretically perfect reconstruction.

The full range of human hearing is between 20 Hz and 20 kHz. The minimum sampling rate that satisfies the sampling theorem for this full bandwidth is 40 kHz. The 44.1 kHz sampling rate used for Compact disc was chosen for this and other technical reasons.

Oversampling

In some cases, it is desirable to have a sampling frequency more than twice the desired system bandwidth so that a digital filter can be used in exchange for a weaker analog anti-aliasing filter. This process is known as oversampling.

Undersampling

Conversely, one may sample below the Nyquist rate. For a baseband signal (one that has components from 0 to the band limit), this introduces aliasing, but for a passband signal (one that does not have low frequency components), there are no low frequency signals for the aliases of high frequency signals to collide with, and thus one can sample a high frequency (but narrow bandwidth) signal at a much lower sample rate than the Nyquist rate.

Audio

In digital audio the most common sampling rates are 44.1 kHz, 48 kHz, and 96 kHz. A more complete list is as follows:

Sampling rate	Use
8,000 Hz	telephone and encrypted walkie-talkie, wireless intercom and wireless microphone transmission; adequate for human speech but without sibilance; *ess* sounds like *eff*
11,025 Hz	one quarter the sampling rate of audio CDs; used for lower-quality PCM, MPEG audio and for audio analysis of subwoofer bandpasses
16,000 Hz	wideband frequency extension over standard telephone narrowband

	8,000 Hz. Used in most modern VoIP and VVoIP communication products.
22,050 Hz	one half the sampling rate of audio CDs; used for lower-quality PCM and MPEG audio and for audio analysis of low frequency energy. Suitable for digitizing early 20th century audio formats such as 78s
32,000 Hz	miniDV digital video camcorder, video tapes with extra channels of audio (e.g. DVCAM with 4 Channels of audio), DAT (LP mode), Germany's Digitales Satellitenradio **(German)**, NICAM digital audio, used alongside analogue television sound in some countries. High-quality digital wireless microphones.
44,056 Hz	Used by digital audio locked to NTSC *color* video signals (245 lines by 3 samples by 59.94 fields per second = 29.97 frames per second).
44,100 Hz	audio CD, also most commonly used with MPEG-1 audio (VCD, SVCD, MP3). Originally chosen by Sony because it could be recorded on modified video equipment running at either 25 frames per second (PAL) or 30fps (using an NTSC *monochrome* video recorder) and cover the 20 kHz bandwidth thought necessary to match professional analog recording equipment of the time. A PCM adaptor would fit digital audio samples into the analog video channel of, for example, PAL video tapes using 588 lines by 3 samples by 25 frames per second. Much pro audio gear uses (or is able to select) 44.1 kHz sampling, including mixers, EQs, compressors, reverb, crossovers, recording devices and CD-quality encrypted wireless microphones.
47,250 Hz	world's first commercial PCM sound recorder by Nippon Columbia (Denon)
48,000 Hz	The standard audio sampling rate used by professional digital video equipment such as tape recorders, video servers, vision mixers and so on. This rate was chosen because it could deliver a 22 kHz frequency response and work with 29.97 frames per second NTSC video - as well as 25 fps, 30fps and 24fps systems. With 29.97fps systems it is necessary to handle 1601.6 audio samples per frame delivering an integer number of audio samples only every fifth video frame. Also used for sound with consumer video formats like DV, digital TV, DVD, and films. The professional Serial Digital Interface (SDI) and High-definition Serial Digital Interface (HD-SDI) used to connect broadcast television equipment together uses this audio sampling frequency. Much professional audio gear uses (or is able to select) 48 kHz sampling, including mixers, EQs, compressors, reverb, crossovers and recording devices such as DAT.
50,000 Hz	first commercial digital audio recorders from the late 70s from 3M and Soundstream
50,400 Hz	sampling rate used by the Mitsubishi X-80 digital audio recorder
88,200 Hz	sampling rate used by some professional recording equipment when the destination is CD (multiples of 44,100 Hz). Some pro audio gear uses (or is able to select) 88.2 kHz sampling, including mixers, EQs, compressors,

	reverb, crossovers and recording devices.
96,000 Hz	DVD-Audio, some LPCM DVD tracks, BD-ROM (Blu-ray Disc) audio tracks, HD DVD (High-Definition DVD) audio tracks. Most pro audio gear uses (or is able to select) 96 kHz sampling, including mixers, EQs, compressors, reverb, crossovers and recording devices. This sampling frequency is twice the 48 kHz standard commonly used with audio on professional video equipment.
176,400 Hz	Sampling rate used by HDCD recorders and other professional applications for CD production.
192,000 Hz	DVD-Audio, some LPCM DVD tracks, BD-ROM (Blu-ray Disc) audio tracks, and HD DVD (High-Definition DVD) audio tracks, High-Definition audio recording devices and audio editing software. This sampling frequency is four times the 48 kHz standard commonly used with audio on professional video equipment.
352,800 Hz	Digital eXtreme Definition, used for recording and editing Super Audio CDs, as 1-bit DSD is not suited for editing. Eight times the frequency of 44.1 kHz.
2,822,400 Hz	SACD, 1-bit sigma-delta modulation process known as Direct Stream Digital, co-developed by Sony and Philips.
5,644,800 Hz	Double-Rate DSD, 1-bit Direct Stream Digital at 2x the rate of the SACD. Used in some professional DSD recorders.

Video systems

In digital video, the temporal sampling rate is defined the frame rate – or rather the field rate – rather than the notional pixel clock. The image sampling frequency is the repetition rate of the sensor integration period. Since the integration period may be significantly shorter than the time between repetitions, the sampling frequency can be different from the inverse of the sample time.

- 50 Hz - PAL video
- 60 / 1.001 Hz ~= 59.97 Hz - NTSC video

When analog video is converted to digital video, a different sampling process occurs, this time at the pixel frequency, corresponding to a spatial sampling rate along scan lines. Some common pixel sampling rates are:

- 13.5 MHz - CCIR 601, D1 video

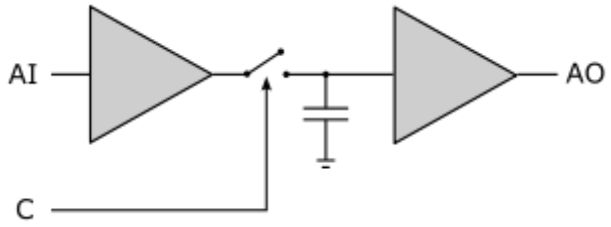
Spatial sampling in the other direction is determined by the spacing of scan lines in the raster. The sampling rates and resolutions in both spatial directions can be measured in units of lines per picture height.

Spatial aliasing of high-frequency luma or chroma video components shows up as a moiré pattern.

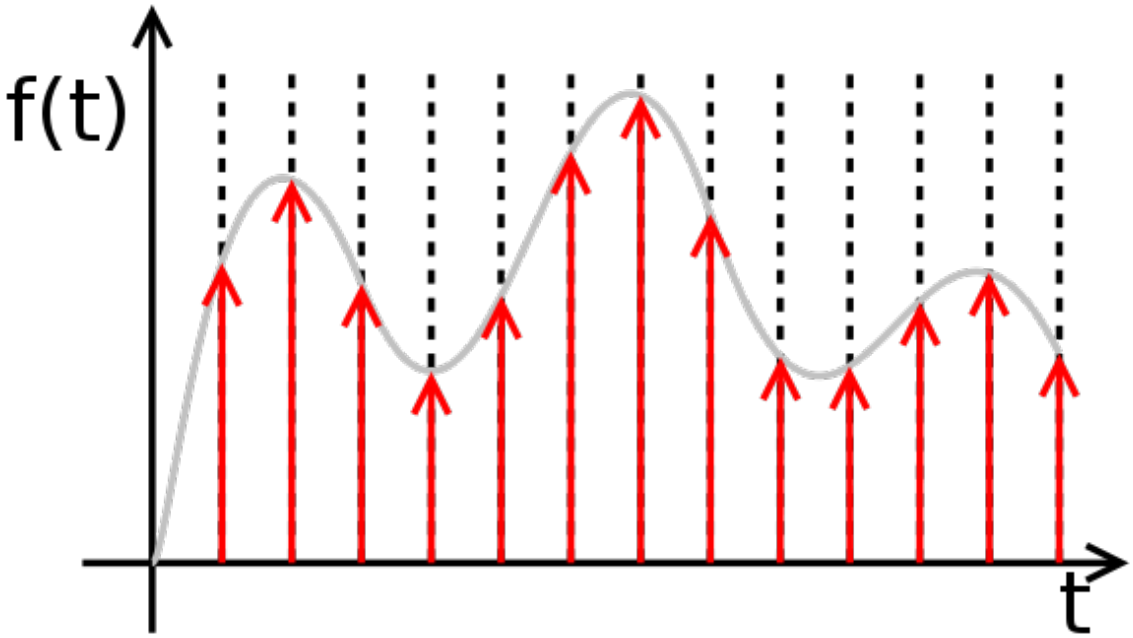
Chapter- 10

Sample and Hold, Undersampling, Upsampling & Nyquist Frequency

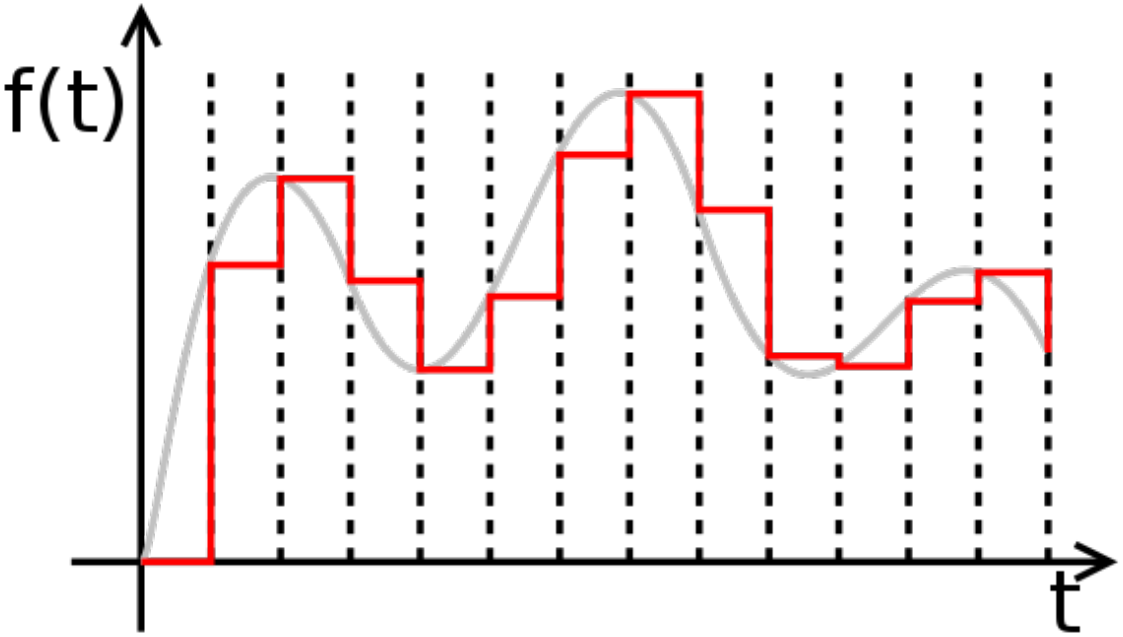
Sample and hold



A simplified sample and hold circuit diagram. AI is an analog input, AO — an analog output, C — a control signal.



Sample times.



Sample and hold.

In electronics, a **sample and hold** (**S/H**, also "follow-and-hold") circuit is an analog device that samples (captures, grabs) the voltage of a continuously varying analog signal and holds (locks, freezes) its value at a constant level for a specified minimal period of time. Sample and hold circuits and related peak detectors are the elementary analog memory devices. They are typically used in analog-to-digital converters to eliminate variations in input signal that can corrupt the conversion process.

A typical sample and hold circuit stores electric charge in a capacitor and contains at least one fast FET switch and at least one operational amplifier. To sample the input signal the switch connects the capacitor to the output of a buffer amplifier. The buffer amplifier charges or discharges the capacitor so that the voltage across the capacitor is practically equal, or proportional to, input voltage. In hold mode the switch disconnects the capacitor from the buffer. The capacitor is invariably discharged by its own leakage currents and useful load currents, which makes the circuit inherently volatile, but the loss of voltage (*voltage drop*) within a specified hold time remains within an acceptable error margin.

Purpose

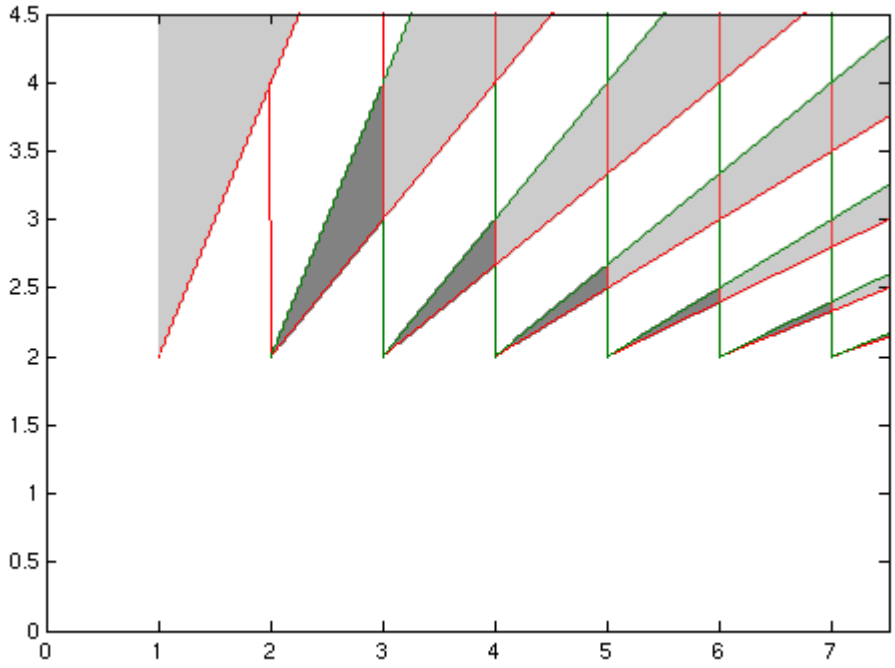
The reasons for using such a circuit are varied. In some kinds of analog-to-digital converters, the input is often compared to a voltage generated internally from a digital-to-analog converter. The circuit tries a series of values and stops converting once the voltages are "the same" within some defined error margin. If the input value was permitted to change during this comparison process, the resulting conversion would be inaccurate and possibly completely unrelated to the true input value. Such successive approximation converters will often incorporate internal sample and hold circuitry. In addition, sample and hold circuits are often used when multiple samples need to be measured at the same time. Each value is sampled and held, using a common sample clock.

Implementation

In order that the input voltage is held constant for all practical purposes, it is essential that the capacitor have very low leakage, and that it not be loaded to any significant degree which calls for a very high input impedance.

A true sample and hold circuit is connected to the buffer for a short period of time; a *track and hold* circuit is designed to track input continuously.

Undersampling



Plot of sample rates (y axis) versus the upper edge frequency (x axis) for a band of width 1; grays areas are combinations that are "allowed" in the sense that no two frequencies in the band alias to same frequency. The darker gray areas correspond to undersampling with the maximum value of *n* in the equations of this section.

In signal processing, **undersampling** or **bandpass sampling** is a technique where one samples a bandpass filtered signal at a sample rate below the usual Nyquist rate (twice the baseband bandwidth, i.e. twice the upper cut-off frequency), but is still able to reconstruct the signal.

When one samples a bandpass signal, the samples are equal to samples of a low-frequency alias of the high-frequency signal. Such undersampling is also known as bandpass sampling, harmonic sampling, IF sampling, and direct IF-to-digital conversion.

Description

Real-valued signals have Fourier spectra with symmetry about zero. That is, they have a negative-frequency spectrum that is a mirror image of the positive-frequency spectrum. Sampling effectively shifts both sides of the spectrum by multiples of the sampling frequency. The criterion to avoid aliasing is that none of these shifted copies of the spectrum overlap.

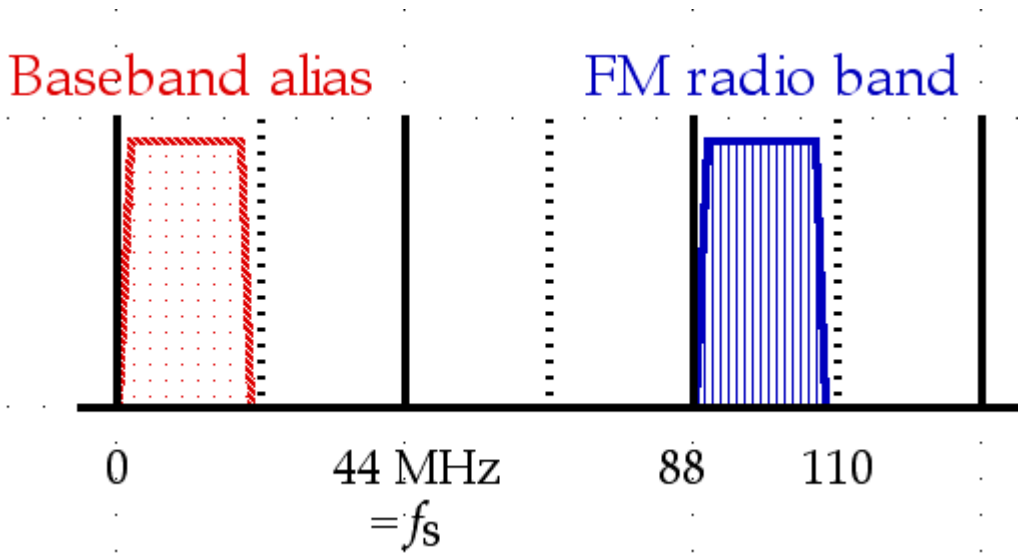
In the case of the bandpass (non-baseband) signals, with low and high band limits f_L and f_H respectively, the condition for an acceptable sample rate is that shifts of the bands from f_L to f_H and from $-f_H$ to $-f_L$ must not overlap when shifted by all integer multiples of sampling rate f_s. This condition reduces to the constraint:

$$\frac{2f_H}{n} \leq f_s \leq \frac{2f_L}{n-1}, \text{ for some } n \text{ satisfying: } 1 \leq n \leq \left\lfloor \frac{f_H}{f_H - f_L} \right\rfloor$$

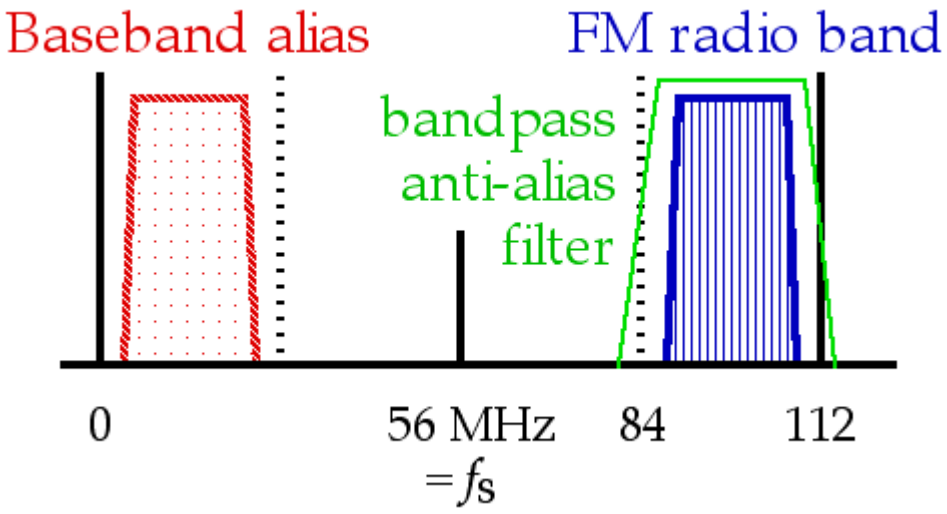
The highest n for which the condition is satisfied leads to the lowest possible sampling rates.

Important signals of this sort include a radio's intermediate-frequency (IF) or radio-frequency (RF) signal.

If $n > 1$, then the conditions result in what is sometimes referred to as *undersampling*, *bandpass sampling*, or using a sampling rate less than the *Nyquist rate* $2f_H$ obtained from the upper bound of the spectrum.



Spectrum of the FM radio band (88–108 MHz) and its baseband alias under 44 MHz (n = 5) sampling. An anti-alias filter quite tight to the FM radio band is required, and there's not room for stations at nearby expansion channels such as 87.9 without aliasing.



Spectrum of the FM radio band (88–108 MHz) and its baseband alias under 56 MHz (n = 4) sampling, showing plenty of room for bandpass anti-aliasing filter transition bands. The baseband image is frequency-reversed in this case (even n).

Example: Consider FM radio to illustrate the idea of undersampling.

In the US, FM radio operates on the frequency band from f_L = 88 MHz to f_H = 108 MHz. The bandwidth is given by

$$W = f_H - f_L = 108\ \text{MHz} - 88\ \text{MHz} = 20\ \text{MHz}$$

The sampling conditions are satisfied for

$$1 \leq n \leq \lfloor 5.4 \rfloor = \left\lfloor \frac{108\ \text{MHz}}{20\ \text{MHz}} \right\rfloor$$

Therefore, n can be 1, 2, 3, 4, or 5.

The value n = 5 gives the lowest sampling frequencies interval

$$43.2\ \text{MHz} < f_s < 44\ \text{MHz}$$

and this is a scenario of undersampling. In this case, the signal spectrum fits between 2 and 2.5 times the sampling rate (higher than 86.4–88 MHz but lower than 108–110 MHz).

A lower value of n will also lead to a useful sampling rate. For example, using n = 4, the FM band spectrum fits easily between 1.5 and 2.0 times the sampling rate, for a sampling rate near 56 MHz (multiples of the Nyquist frequency being 28, 56, 84, 112, etc.).

When undersampling a real-world signal, the sampling circuit must be fast enough to capture the highest signal frequency of interest. Theoretically, each sample should be taken during an infinitesimally short interval, but this is not practically feasible. Instead, the sampling of the signal should be made in a short enough interval that it can represent the instantaneous value of the signal with the highest frequency. This means that in the FM radio example above, the sampling circuit must be able to capture a signal with a frequency of 108 MHz, not 43.2 MHz. Thus, the sampling frequency may be only a little bit greater than 43.2 MHz, but the input bandwidth of the system must be at least 108 MHz. Similarly, the accuracy of the sampling timing, or aperture uncertainty of the sampler,

frequently the analog-to-digital converter, must be appropriate for the frequencies being sampled 108MHz, not the lower sample rate.
If the sampling theorem is interpreted as requiring twice the highest frequency, then the required sampling rate would be assumed to be greater than the *Nyquist rate* 216 MHz. While this does satisfy the last condition on the sampling rate, it is grossly oversampled.
Note that if a band is sampled with $n > 1$, then a band-pass filter is required for the anti-aliasing filter, instead of a lowpass filter.

As we have seen, the normal baseband condition for reversible sampling is that $X(f) = 0$ outside the open interval: $\left(-\frac{1}{2}f_s, \frac{1}{2}f_s\right)$,

and the reconstructive interpolation function, or lowpass filter impulse response, is $\operatorname{sinc}(t/T)$.

To accommodate undersampling, the bandpass condition is that $X(f) = 0$ outside the union of open positive and negative frequency bands

$$\left(-\frac{n}{2}f_s, -\frac{n-1}{2}f_s\right) \cup \left(\frac{n-1}{2}f_s, \frac{n}{2}f_s\right)$$ for some positive integer n.
which includes the normal baseband condition as case $n = 1$ (except that where the intervals come together at 0 frequency, they can be closed).

The corresponding interpolation function is the bandpass filter given by this difference of lowpass impulse responses:

$$n\operatorname{sinc}\left(\frac{nt}{T}\right) - (n-1)\operatorname{sinc}\left(\frac{(n-1)t}{T}\right).$$

On the other hand, reconstruction is not usually the goal with sampled IF or RF signals. Rather, the sample sequence can be treated as ordinary samples of the signal frequency-shifted to near baseband, and digital demodulation can proceed on that basis, recognizing the spectrum mirroring when n is even.

Further generalizations of undersampling for the case of signals with multiple bands are possible, and signals over multidimensional domains (space or space-time) and have been worked out in detail by Igor Kluvánek.

Upsampling

Upsampling is the process of increasing the sampling rate of a signal. For instance, upsampling raster images such as photographs means increasing the resolution of the image.

The upsampling factor (commonly denoted by *L*) is usually an integer or a rational fraction greater than unity. This factor multiplies the sampling rate or, equivalently, divides the sampling period. For example, if compact disc audio is upsampled by a factor of 5/4 then the resulting sampling rate goes from 44,100 Hz to 55,125 Hz.

Filtering

For an aesthetically pleasing upsample, an interpolation filter is required; in both upsampling and downsampling, such a low-pass filter implements anti-aliasing.

Upsampling process

Consider a discrete signal $f(k)$ on a radian frequency digital frequency range.

Upsampling by integer factor

Let L denote the upsampling factor.

1. Add $L-1$ zeros between each sample in $f(k)$. Or, equivalently define $g(k) = \begin{cases} f\left(\frac{k}{L}\right) & \text{if } \frac{k}{L} \text{ is an integer} \\ 0 & \text{otherwise} \end{cases}$
2. Filter with a low-pass filter which, theoretically, should be the sinc filter with frequency cut off at $\frac{\pi}{L}$

The second step calls for the use of a perfect low-pass filter, which is not implementable. When choosing a realizable low-pass filter this will have to be considered and it will have aliasing effects. These aliases can be removed to a reasonable extent by a finite impulse response low pass filter. The presence of zeros in the sequence which is passed through the filter can be used to reduce the complexity of the filter implementation. The original filter can be split to L subfilters and the output of each of these subfilters is sequentially tapped to obtain the filtered output sequence.

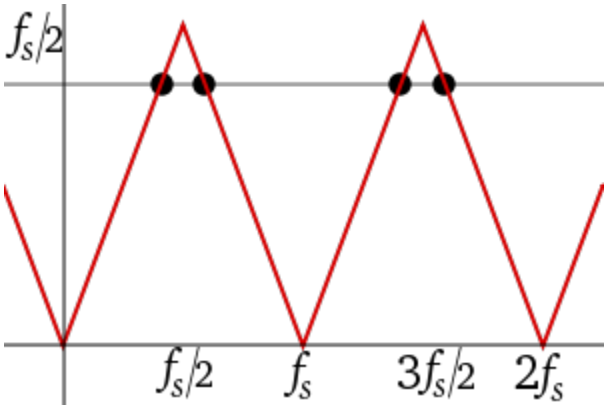
Upsampling by rational fraction

Let L/M denote the upsampling factor.

1. Upsample by a factor of L
2. Downsample by a factor of M

Note that upsampling requires an interpolation filter after increasing the data rate and that downsampling requires a filter before decimation. These two filters can be combined into a single filter. Since both interpolation and anti-aliasing filters are low-pass filters, the filter with the smallest bandwidth is more restrictive and, thus, can be used in place of both filters. Since the rational fraction L/M is greater than unity when $M < L$, the single low-pass filter should have cutoff at $1 / 2L$ cycles per intermediate sample, the Nyquist frequency of the input sample rate.

Nyquist frequency



Frequencies above $f_s/2$ (the Nyquist frequency) have an alias below $f_s/2$, whose value is given by this graph. $f_s/2$ is also called folding frequency, because of the symmetry between 0 and f_s.

The **Nyquist frequency**, named after the Swedish-American engineer Harry Nyquist or the Nyquist–Shannon sampling theorem, is half the sampling frequency of a discrete signal processing system. It is sometimes known as the folding frequency of a sampling system.

The sampling theorem shows that aliasing can be avoided if the Nyquist frequency is greater than the bandwidth, or maximum component frequency, of the signal being sampled.

The Nyquist frequency should not be confused with the *Nyquist rate*, which is the lower bound of the sampling frequency that satisfies the Nyquist sampling criterion for a given signal or family of signals. This lower bound is twice the bandwidth or maximum component frequency of the signal. *Nyquist rate*, as commonly used with respect to sampling, is a property of a continuous-time signal, not of a system, whereas *Nyquist frequency* is a property of a discrete-time system, not of a signal. The domain of the signals does not have to be time, though that is common, leading to Nyquist frequency in

hertz; for example, an image sampling system has a Nyquist frequency expressed in units such as cycles per meter.

The aliasing problem

In theory, a Nyquist frequency just larger than the signal bandwidth is sufficient to allow perfect reconstruction of the signal from the samples. However, this reconstruction requires an ideal filter that passes some frequencies unchanged while suppressing all others completely (commonly called a brick-wall filter). In practice, perfect reconstruction is unattainable. Some amount of aliasing is unavoidable.

Signal frequencies higher than the Nyquist frequency will encounter a "folding" about the Nyquist frequency, back into lower frequencies. For example, if the sample rate is 20 kHz, the Nyquist frequency is 10 kHz, and an 11 kHz signal will fold, or alias, to 9 kHz. However, a 9 kHz signal can also fold up to 11 kHz in that case if the reconstruction filter is not adequate. Both types of aliasing can be important.

When attainable filters are used, some degree of oversampling is necessary to accommodate the practical constraints on anti-aliasing filters: instead of a brickwall, one has flat response in the passband up to a point called the cutoff frequency or corner frequency, (pass all frequencies below there unchanged), then gradual rolloff in a transition band, finally suppressing signals above a certain point completely or almost completely in the stopband. Thus, frequencies close to the Nyquist frequency may be distorted in the sampling and reconstruction process, so the bandwidth should be kept below the Nyquist frequency by some margin (frequency headroom) that depends on the actual filters used.

For example, audio CDs have a sampling frequency of 44100 Hz. The Nyquist frequency is therefore 22050 Hz, which is an upper bound on the highest frequency the data can unambiguously represent. If the chosen anti-aliasing filter (a low-pass filter in this case) has a transition band of 2000 Hz, then the cut-off frequency should be no higher than 20050 Hz to yield a signal with negligible power at frequencies of 22050 Hz and greater.

Other meanings

Early uses of the term *Nyquist frequency*, such as those cited above, are all consistent with the definition presented here. Some later publications, including some respectable textbooks, call twice the signal bandwidth (the Nyquist rate) as Nyquist frequency; this is a distinctly minority usage.

Chapter- 11

Continuous Signal & Discrete Signal

Continuous signal

A **continuous signal** or a **continuous-time signal** is a varying quantity (a signal) whose domain, which is often time, is a continuum (e.g., a connected interval of the reals). That is, the function's domain is an uncountable set. The function itself need not be continuous. To contrast, a discrete time signal has a countable domain, like the natural numbers.

The signal is defined over a domain, which may or may not be finite, and there is a functional mapping from the domain to the value of the signal. The continuity of the time variable, in connection with the law of density of real numbers, means that the signal value can be found at any arbitrary point in time.

A typical example of an infinite duration signal is:

$$f(t) = \sin(t), \quad t \in \mathbb{R}$$

A finite duration counterpart of the above signal could be:

$$f(t) = \sin(t), \quad t \in [-\pi, \pi]$$ and $f(t) = 0$ otherwise.

The value of a finite (or infinite) duration signal may or may not be finite. For example,

$$f(t) = \frac{1}{t}, \quad t \in [0, 1]$$ and $f(t) = 0$ otherwise,

is a finite duration signal but it takes an infinite value for $t = 0$.

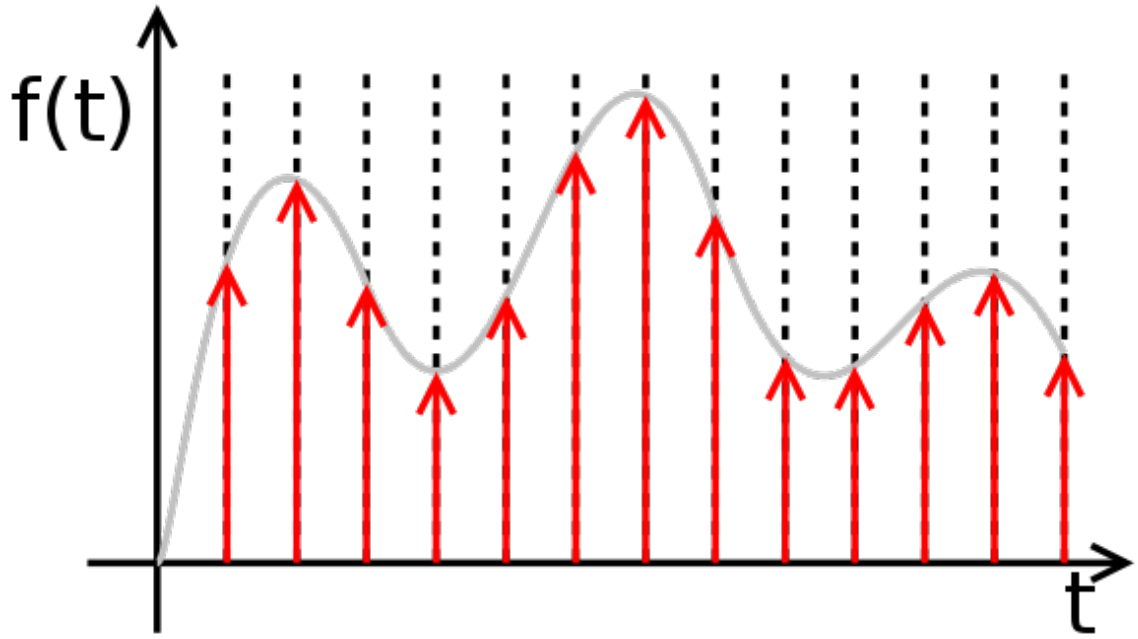
In many disciplines, the convention is that a continuous signal must always have a finite value, which makes more sense in the case of physical signals.

For some purposes, infinite singularities are acceptable as long as the signal is integrable over any finite interval (for example, the t^{-1} signal is not integrable, but t^{-2} is).

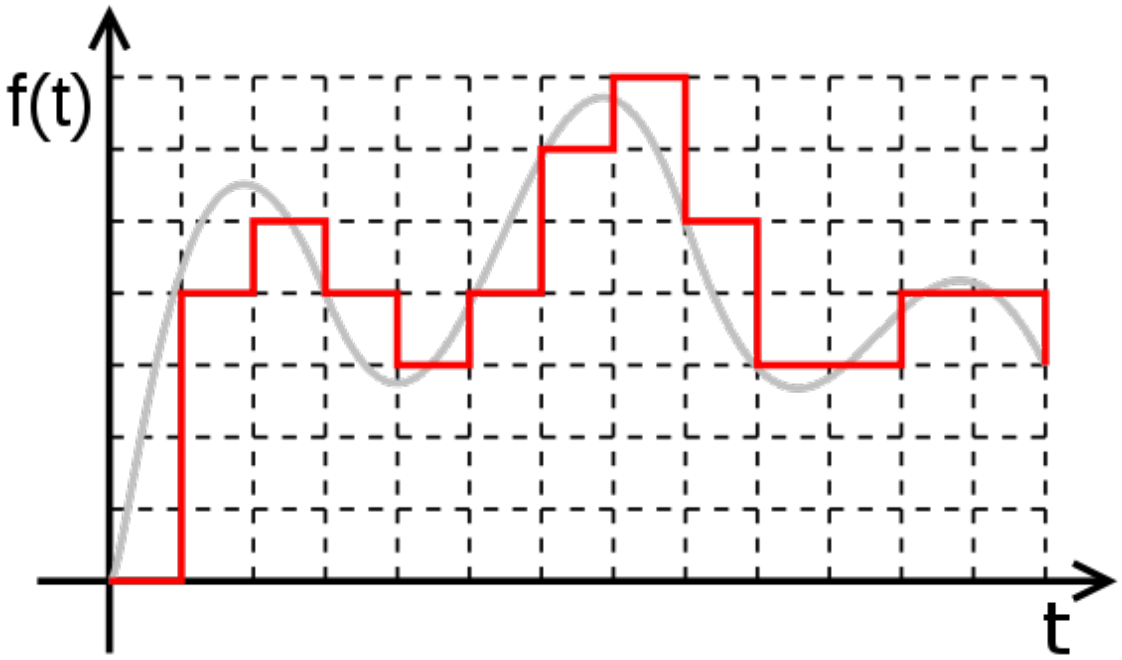
Any analogue signal is continuous by nature. Discrete signals, used in digital signal processing, can be obtained by sampling and quantization of continuous signals.

Continuous signal may also be defined over an independent variable other than time. Another very common independent variable is space and is particularly useful in image processing, where two space dimensions are used.

Discrete signal



Discrete sampled signal



Digital signal

A **discrete signal** or **discrete-time signal** is a time series consisting of a sequence of quantities. In other words, it is a time series that is a function over a domain of discrete integers.

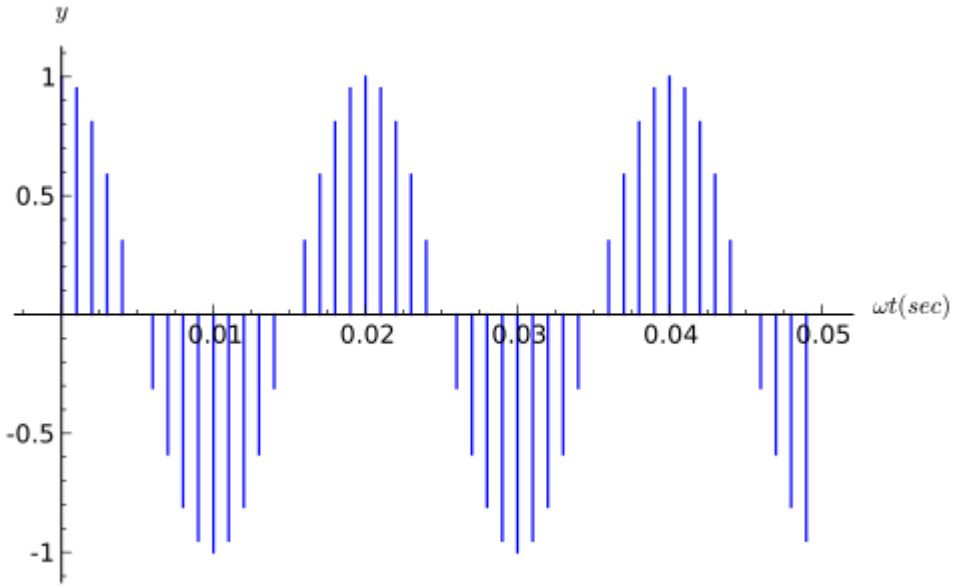
Unlike a continuous-time signal, a discrete-time signal is not a function of a continuous argument; however, it may have been obtained by sampling from a continuous-time signal, and then each value in the sequence is called a sample. When a discrete-time signal obtained by sampling is a sequence corresponding to uniformly spaced times, it has an associated sampling rate; the sampling rate is not apparent in the data sequence, and so needs to be associated as a separate data item.

Acquisition

Discrete signals may have several origins, but can usually be classified into one of two groups:

- By acquiring values of an analog signal at constant or variable rate. This process is called sampling.
- By recording the number of events of a given kind over finite time periods. For example, this could be the number of people taking a certain elevator every day.

Digital signals



Discrete cosine waveform with frequency of 50 Hz and a sampling rate of 1000 samples/sec, easily satisfying the sampling theorem for reconstruction of the original cosine function from samples.

A digital signal is a discrete-time signal for which not only the time but also the amplitude has been made discrete; in other words, its samples take on only values from a discrete set.

The process of converting a continuous-valued discrete-time signal to a digital (discrete-valued discrete-time) signal is known as analog-to-digital conversion. It usually proceeds by replacing each original sample value by an approximation selected from a given discrete set (for example by truncating or rounding, but much more sophisticated methods exist), a process known as quantization. This process loses information, and so discrete-valued signals are only an approximation of the converted continuous-valued discrete-time signal, itself only an approximation of the original continuous-valued continuous-time signal.

Common practical digital signals are represented as 8-bit (256 levels), 16-bit (65,536 levels), 32-bit (4.3 billion levels), and so on, though any number of quantization levels is possible, not just powers of two.

Chapter- 12

Nyquist–Shannon Sampling Theorem

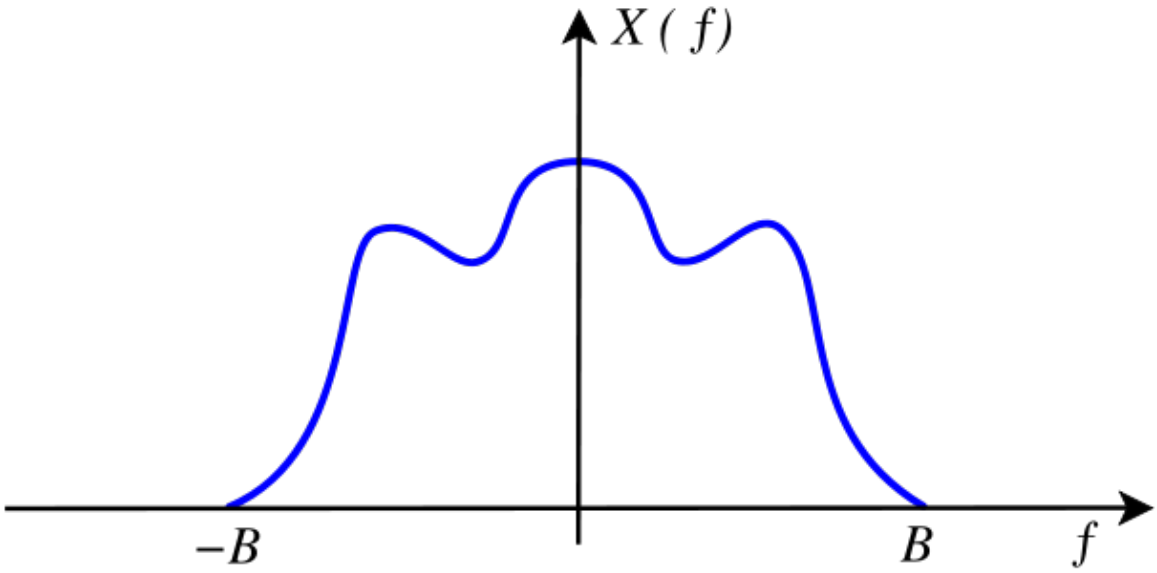


Fig.1: Hypothetical spectrum of a **bandlimited signal** as a function of frequency

The **Nyquist–Shannon sampling theorem**, after Harry Nyquist and Claude Shannon, is a fundamental result in the field of information theory, in particular telecommunications and signal processing. Sampling is the process of converting a signal (for example, a function of continuous time or space) into a numeric sequence (a function of discrete time or space). Shannon's version of the theorem states:

If a function $x(t)$ contains no frequencies higher than B hertz, it is completely determined by giving its ordinates at a series of points spaced $1/(2B)$ seconds apart.

The theorem is commonly called the **Nyquist sampling theorem**; since it was also discovered independently by E. T. Whittaker, by Vladimir Kotelnikov, and by others, it is also known as **Nyquist–Shannon–Kotelnikov**, **Whittaker–Shannon–Kotelnikov**, **Whittaker–Nyquist–Kotelnikov–Shannon**, **WKS**, etc., sampling theorem, as well as

the **Cardinal Theorem of Interpolation Theory**. It is often referred to simply as ***the sampling theorem***.

In essence, the theorem shows that a bandlimited analog signal that has been sampled can be perfectly reconstructed from an infinite sequence of samples if the sampling rate exceeds $2B$ samples per second, where B is the highest frequency in the original signal. If a signal contains a component at exactly B hertz, then samples spaced at exactly $1/(2B)$ seconds do not completely determine the signal, Shannon's statement notwithstanding. This sufficient condition can be weakened, as discussed at Sampling of non-baseband signals below.

More recent statements of the theorem are sometimes careful to exclude the equality condition; that is, the condition is if $x(t)$ contains no frequencies higher than *or equal to* B; this condition is equivalent to Shannon's except when the function includes a steady sinusoidal component at exactly frequency B.

The theorem assumes an idealization of any real-world situation, as it only applies to signals that are sampled for infinite time; any time-limited $x(t)$ cannot be perfectly bandlimited. Perfect reconstruction is mathematically possible for the idealized model but only an approximation for real-world signals and sampling techniques, albeit in practice often a very good one.

The theorem also leads to a formula for reconstruction of the original signal. The constructive proof of the theorem leads to an understanding of the aliasing that can occur when a sampling system does not satisfy the conditions of the theorem.

The sampling theorem provides a sufficient condition, but not a necessary one, for perfect reconstruction. The field of compressed sensing provides a stricter sampling condition when the underlying signal is known to be sparse. Compressed sensing specifically yields a sub-Nyquist sampling criterion.

Introduction

A signal or function is bandlimited if it contains no energy at frequencies higher than some bandlimit or bandwidth B. A signal that is bandlimited is constrained in how rapidly it changes in time, and therefore how much detail it can convey in an interval of time. The sampling theorem asserts that the uniformly spaced discrete samples are a complete representation of the signal if this bandwidth is less than half the sampling rate. To formalize these concepts, let $x(t)$ represent a continuous-time signal and $X(f)$ be the continuous Fourier transform of that signal:

$$X(f) \ \stackrel{\text{def}}{=} \ \int_{-\infty}^{\infty} x(t)\ e^{-i2\pi f t}\ dt.$$

The signal $x(t)$ is said to be bandlimited to a one-sided baseband bandwidth, B, if

$$X(f) = 0 \text{ for all } |f| > B,$$

or, equivalently, supp(X) $\subseteq [-B, B]$. Then the sufficient condition for exact reconstructability from samples at a uniform sampling rate f_s (in samples per unit time) is:

$$f_s > 2B.$$

The quantity $2B$ is called the *Nyquist rate* and is a property of the bandlimited signal, while $f_s/2$ is called the *Nyquist frequency* and is a property of this sampling system.

The time interval between successive samples is referred to as the *sampling interval*:

$$T \stackrel{\text{def}}{=} \frac{1}{f_s},$$

and the samples of $x(t)$ are denoted by:

$$x[n] = x(nT),$$

where n is an integer. The sampling theorem leads to a procedure for reconstructing the original $x(t)$ from the samples and states sufficient conditions for such a reconstruction to be exact.

The sampling process

The theorem describes two processes in signal processing: a sampling process, in which a continuous time signal is converted to a discrete time signal, and a reconstruction process, in which the original continuous signal is recovered from the discrete time signal.

The continuous signal varies over *time* (or *space* in a digitized image, or another independent variable in some other application) and the sampling process is performed by measuring the continuous signal's value every T units of time (or space), which is called the *sampling interval*. In practice, for signals that are a function of time, the sampling interval is typically quite small, on the order of milliseconds, microseconds, or less. This results in a sequence of numbers, called *samples*, to represent the original signal. Each sample value is associated with the instant in time when it was measured. The reciprocal of the sampling interval ($1/T$) is the sampling frequency denoted f_s, which is measured in samples per unit of time. If T is expressed in seconds, then f_s is expressed in Hz.

Reconstruction

Reconstruction of the original signal is an interpolation process that mathematically defines a continuous-time signal $x(t)$ from the discrete samples $x[n]$ and at times in between the sample instants nT.

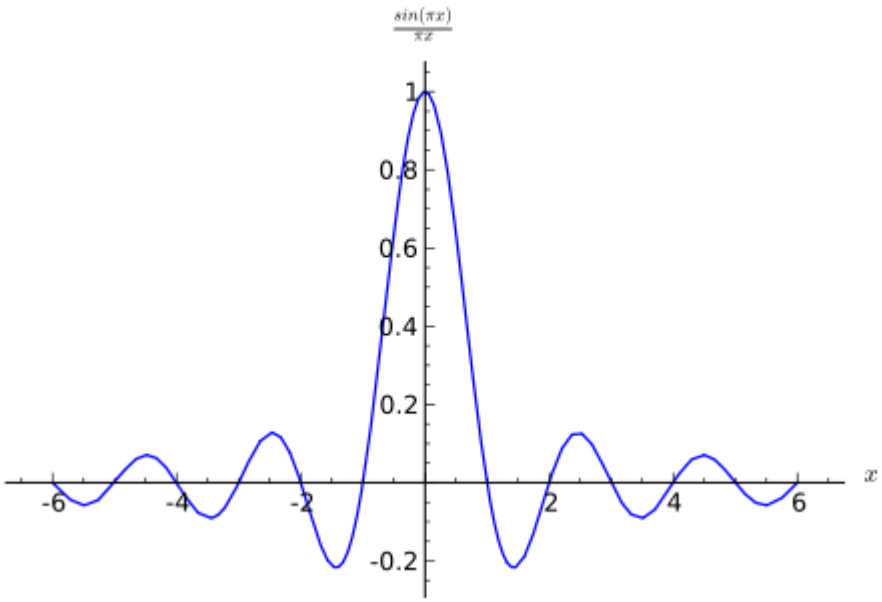


Fig.2: The normalized sinc function: $\sin(\pi x) / (\pi x)$ ... showing the central peak at $x = 0$, and zero-crossings at the other integer values of x.

- **The procedure:** Each sample value is multiplied by the sinc function scaled so that the zero-crossings of the sinc function occur at the sampling instants and that the sinc function's central point is shifted to the time of that sample, nT. All of these shifted and scaled functions are then added together to recover the original signal. The scaled and time-shifted sinc functions are continuous making the sum of these also continuous, so the result of this operation is a continuous signal. This procedure is represented by the Whittaker–Shannon interpolation formula.

- **The condition:** The signal obtained from this reconstruction process can have no frequencies higher than one-half the sampling frequency. According to the theorem, the reconstructed signal will match the original signal provided that the original signal contains no frequencies at or above this limit. This condition is called the *Nyquist criterion*, or sometimes the *Raabe condition.*

If the original signal contains a frequency component equal to one-half the sampling rate, the condition is not satisfied. The resulting reconstructed signal may have a component at that frequency, but the amplitude and phase of that component generally will not match the original component.

This reconstruction or interpolation using sinc functions is not the only interpolation scheme. Indeed, it is impossible in practice because it requires summing an infinite number of terms. However, it is the interpolation method that in theory exactly reconstructs *any* given bandlimited $x(t)$ with *any* bandlimit $B < 1/(2T)$; any other method that does so is formally equivalent to it.

Practical considerations

A few consequences can be drawn from the theorem:

- If the highest frequency B in the original signal is known, the theorem gives the lower bound on the sampling frequency for which perfect reconstruction can be assured. This lower bound to the sampling frequency, $2B$, is called the Nyquist rate.

- If instead the sampling frequency is known, the theorem gives us an upper bound for frequency components, $B<f_s/2$, of the signal to allow for perfect reconstruction. This upper bound is the Nyquist frequency, denoted f_N.

- Both of these cases imply that the signal to be sampled must be bandlimited; that is, any component of this signal which has a frequency above a certain bound should be zero, or at least sufficiently close to zero to allow us to neglect its influence on the resulting reconstruction. In the first case, the condition of bandlimitation of the sampled signal can be accomplished by assuming a model of the signal which can be analysed in terms of the frequency components it contains; for example, sounds that are made by a speaking human normally contain very small frequency components at or above 10 kHz and it is then sufficient to sample such an audio signal with a sampling frequency of at least 20 kHz. For the second case, we have to assure that the sampled signal is bandlimited such that frequency components at or above half of the sampling frequency can be neglected. This is usually accomplished by means of a suitable low-pass filter; for example, if it is desired to sample speech waveforms at 8 kHz, the signals should first be lowpass filtered to below 4 kHz.

- In practice, neither of the two statements of the sampling theorem described above can be completely satisfied, and neither can the reconstruction formula be precisely implemented. The reconstruction process that involves scaled and delayed sinc functions can be described as *ideal*. It cannot be realized in practice since it implies that each sample contributes to the reconstructed signal at almost all time points, requiring summing an infinite number of terms. Instead, some type of approximation of the sinc functions, finite in length, has to be used. The error that corresponds to the sinc-function approximation is referred to as *interpolation error*. Practical digital-to-analog converters produce neither scaled and delayed sinc functions nor ideal impulses (that if ideally low-pass filtered would yield the original signal), but a sequence of scaled and delayed rectangular pulses. This practical piecewise-constant output can be modeled as a zero-order

hold filter driven by the sequence of scaled and delayed dirac impulses referred to in the mathematical basis section below. A shaping filter is sometimes used after the DAC with zero-order hold to make a better overall approximation.

- Furthermore, in practice, a signal can never be perfectly bandlimited, since ideal "brick-wall" filters cannot be realized. All practical filters can only attenuate frequencies outside a certain range, not remove them entirely. In addition to this, a "time-limited" signal can never be bandlimited. This means that even if an ideal reconstruction could be made, the reconstructed signal would not be exactly the original signal. The error that corresponds to the failure of bandlimitation is referred to as *aliasing*.

- The sampling theorem does not say what happens when the conditions and procedures are not exactly met, but its proof suggests an analytical framework in which the non-ideality can be studied. A designer of a system that deals with sampling and reconstruction processes needs a thorough understanding of the signal to be sampled, in particular its frequency content, the sampling frequency, how the signal is reconstructed in terms of interpolation, and the requirement for the total reconstruction error, including aliasing, sampling, interpolation and other errors. These properties and parameters may need to be carefully tuned in order to obtain a useful system.

Aliasing

The Poisson summation formula shows that the samples, $x[n]=x(nT)$, of function $x(t)$ are sufficient to create a periodic summation of function $X(f)$. The result is:

$$X_s(f) \ \stackrel{\text{def}}{=} \ \sum_{k=-\infty}^{\infty} X\left(f - k f_s\right) = T \sum_{n=-\infty}^{\infty} x(nT)\, e^{-i2\pi nTf}. \qquad \textbf{(Eq.1)}$$

As depicted in Figures 3, 4, and 8, copies of $X(f)$ are shifted by multiples of f_s and combined by addition.

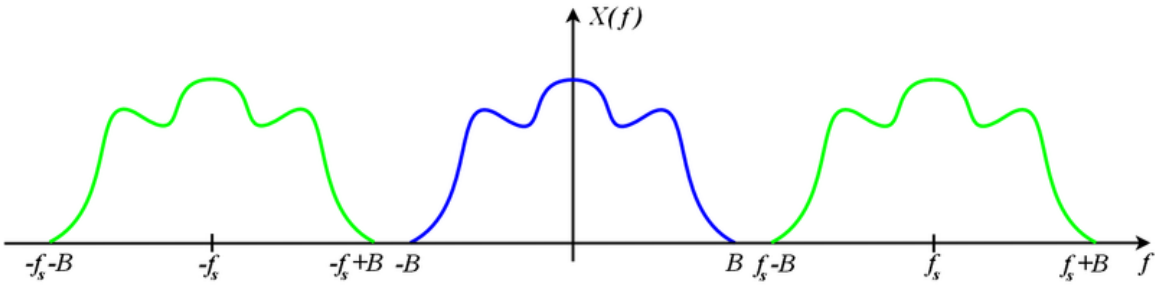


Fig.3: Hypothetical spectrum of a properly sampled bandlimited signal (blue) and images (green) that do not overlap. A "*brick-wall*" low-pass filter can remove the images and leave the original spectrum, thus recovering the original signal from the samples.

If the sampling condition is not satisfied, adjacent copies overlap, and it is not possible in general to discern an unambiguous $X(f)$. Any frequency component above $f_s/2$ is indistinguishable from a lower-frequency component, called an *alias*, associated with one of the copies. The reconstruction technique described below produces the alias, rather than the original component, in such cases.

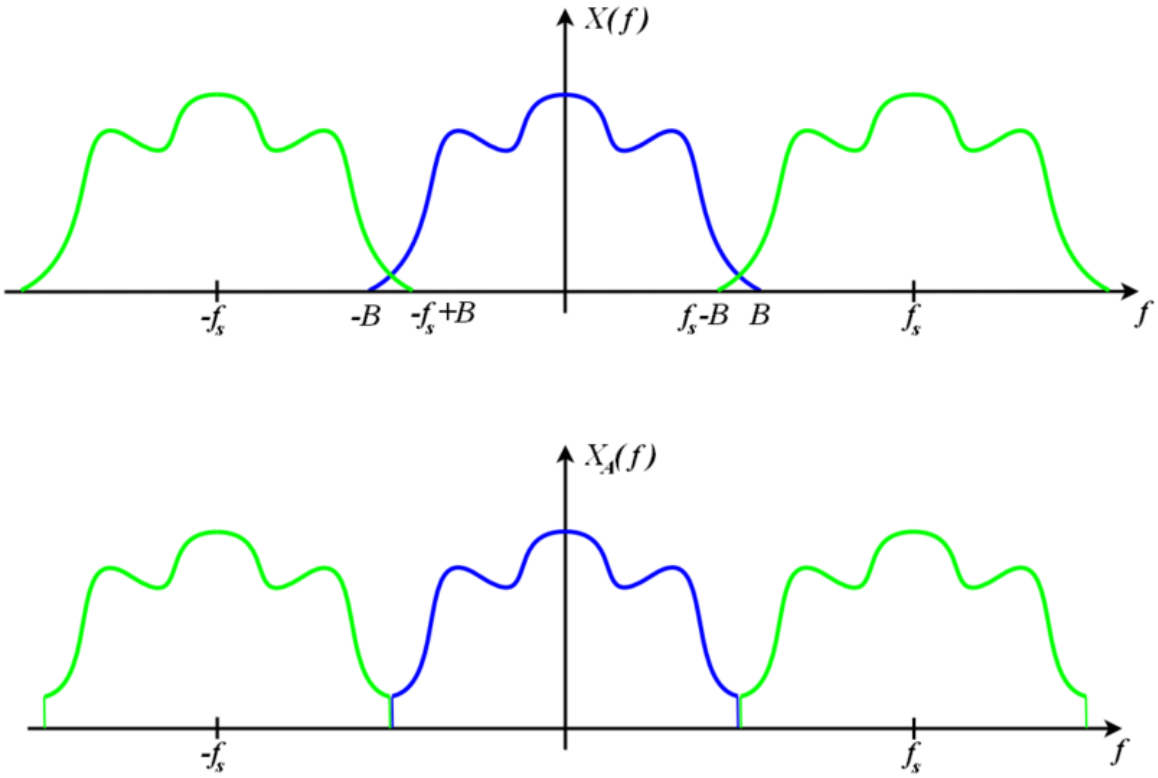


Fig.4 Top: Hypothetical spectrum of an insufficiently sampled bandlimited signal (blue), $X(f)$, where the images (green) overlap. These overlapping edges or "*tails*" of the images add, creating a spectrum unlike the original. **Bottom:** Hypothetical spectrum of a marginally sufficiently sampled bandlimited signal (blue), $X_A(f)$, where the images (green) narrowly do not overlap. But the overall sampled spectrum of $X_A(f)$ is identical to the overall inadequately sampled spectrum of $X(f)$ (top) because the sum of baseband and images are the same in both cases. The discrete sampled signals $x_A[n]$ and $x[n]$ are also identical. It is not possible, just from examining the spectra (or the sampled signals), to tell the two situations apart. If this were an audio signal, $x_A[n]$ and $x[n]$ would sound the same and the presumed "*properly*" sampled $x_A[n]$ would be the *alias* of $x[n]$ since the spectrum $X_A(f)$ masquerades as the spectrum $X(f)$.

For a sinusoidal component of exactly half the sampling frequency, the component will in general alias to another sinusoid of the same frequency, but with a different phase and amplitude.

To prevent or reduce aliasing, two things can be done:

1. Increase the sampling rate, to above twice some or all of the frequencies that are aliasing.
2. Introduce an anti-aliasing filter or make the anti-aliasing filter more stringent.

The anti-aliasing filter is to restrict the bandwidth of the signal to satisfy the condition for proper sampling. Such a restriction works in theory, but is not precisely satisfiable in reality, because realizable filters will always allow some *leakage* of high frequencies. However, the leakage energy can be made small enough so that the aliasing effects are negligible.

Application to multivariable signals and images

Fig.5: Subsampled image showing a Moiré pattern

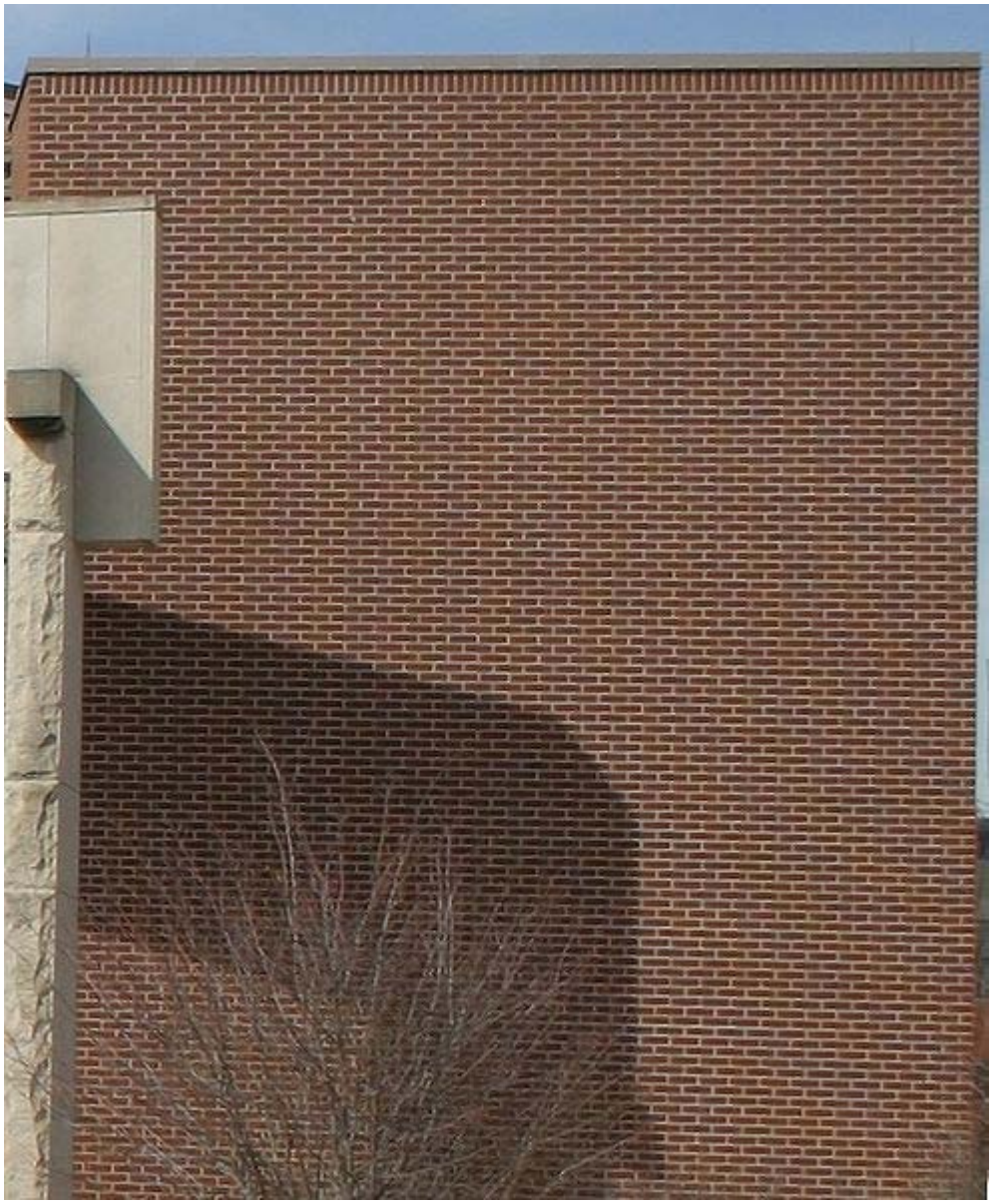


Fig.6

The sampling theorem is usually formulated for functions of a single variable. Consequently, the theorem is directly applicable to time-dependent signals and is normally formulated in that context. However, the sampling theorem can be extended in a straightforward way to functions of arbitrarily many variables. Grayscale images, for example, are often represented as two-dimensional arrays (or matrices) of real numbers representing the relative intensities of pixels (picture elements) located at the intersections of row and column sample locations. As a result, images require two independent variables, or indices, to specify each pixel uniquely — one for the row, and one for the column.

Color images typically consist of a composite of three separate grayscale images, one to represent each of the three primary colors — red, green, and blue, or *RGB* for short.

Other colorspaces using 3-vectors for colors include HSV, LAB, XYZ, etc. Some colorspaces such as cyan, magenta, yellow, and black (CMYK) may represent color by four dimensions. All of these are treated as vector-valued functions over a two-dimensional sampled domain.

Similar to one-dimensional discrete-time signals, images can also suffer from aliasing if the sampling resolution, or pixel density, is inadequate. For example, a digital photograph of a striped shirt with high frequencies (in other words, the distance between the stripes is small), can cause aliasing of the shirt when it is sampled by the camera's image sensor. The aliasing appears as a moiré pattern. The "solution" to higher sampling in the spatial domain for this case would be to move closer to the shirt, use a higher resolution sensor, or to optically blur the image before acquiring it with the sensor.

Another example is shown to the left in the brick patterns. The top image shows the effects when the sampling theorem's condition is not satisfied. When software rescales an image (the same process that creates the thumbnail shown in the lower image) it, in effect, runs the image through a low-pass filter first and then downsamples the image to result in a smaller image that does not exhibit the moiré pattern. The top image is what happens when the image is downsampled without low-pass filtering: aliasing results.

The application of the sampling theorem to images should be made with care. For example, the sampling process in any standard image sensor (CCD or CMOS camera) is relatively far from the ideal sampling which would measure the image intensity at a single point. Instead these devices have a relatively large sensor area at each sample point in order to obtain sufficient amount of light. In other words, any detector has a finite-width point spread function. The analog optical image intensity function which is sampled by the sensor device is not in general bandlimited, and the non-ideal sampling is itself a useful type of low-pass filter, though not always sufficient to remove enough high frequencies to sufficiently reduce aliasing. When the area of the sampling spot (the size of the pixel sensor) is not large enough to provide sufficient anti-aliasing, a separate anti-aliasing filter (optical low-pass filter) is typically included in a camera system to further blur the optical image. Despite images having these problems in relation to the sampling theorem, the theorem can be used to describe the basics of down and up sampling of images.

Downsampling

When a signal is downsampled, the sampling theorem can be invoked via the artifice of resampling a hypothetical continuous-time reconstruction. The Nyquist criterion must still be satisfied with respect to the new lower sampling frequency in order to avoid aliasing. To meet the requirements of the theorem, the signal must usually pass through a low-pass filter of appropriate cutoff frequency as part of the downsampling operation. This low-pass filter, which prevents aliasing, is called an anti-aliasing filter.

Critical frequency

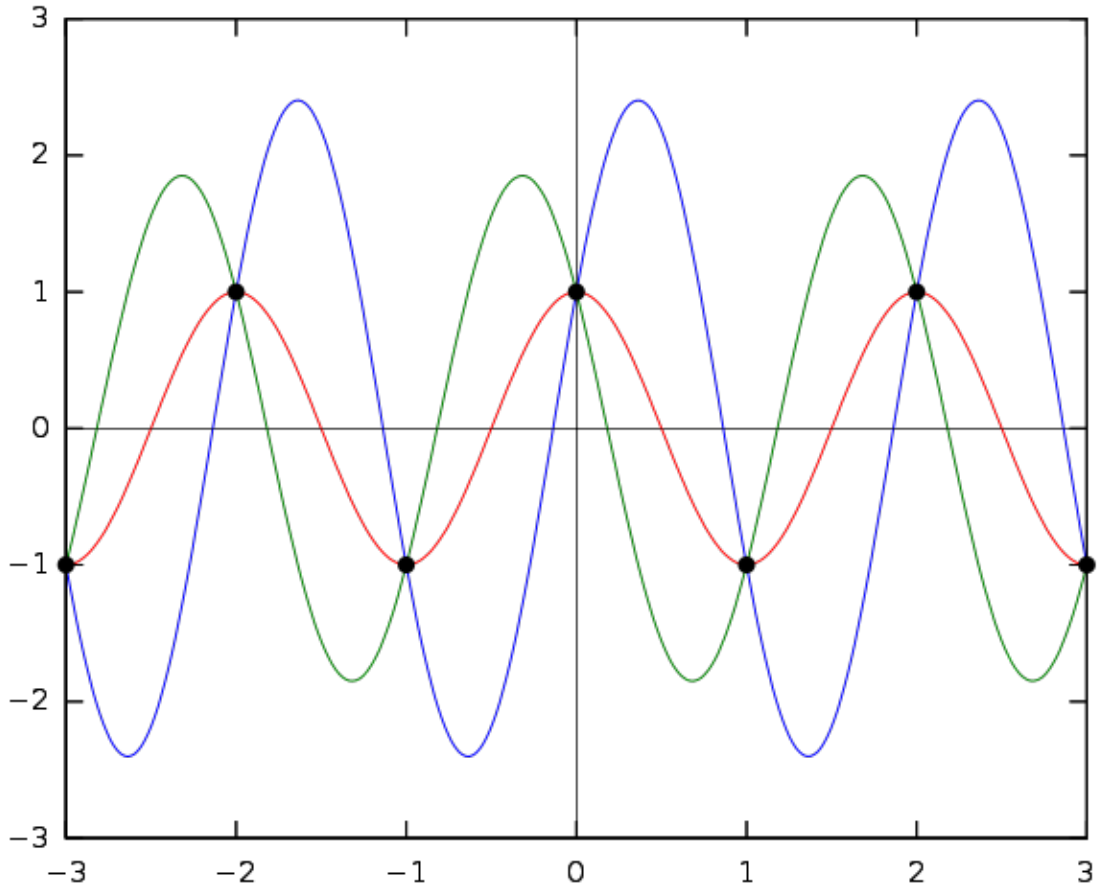


Fig.7: A family of sinusoids at the critical frequency, all having the same sample sequences of alternating +1 and –1. That is, they all are aliases of each other, even though their frequency is not above half the sample rate.

To illustrate the necessity of $f_s > 2B$, consider the sinusoid:

$$x(t) = \cos(2\pi Bt + \theta) \ = \ \cos(2\pi Bt)\cos(\theta) - \sin(2\pi Bt)\sin(\theta).$$

With $f_s = 2B$ or equivalently $T = 1/(2B)$, the samples are given by:

$$x(nT) = \cos(\pi n)\cos(\theta) - \underbrace{\sin(\pi n)}_{0}\sin(\theta) = \cos(\pi n)\cos(\theta).$$

Those samples cannot be distinguished from the samples of:

$$x_A(t) = \cos(2\pi Bt)\cos(\theta).$$

But for any θ such that $\sin(\theta) \neq 0$, $x(t)$ and $x_A(t)$ have different amplitudes and different phase. This and other ambiguities are the reason for the *strict* inequality of the sampling theorem's condition.

Mathematical basis for the theorem

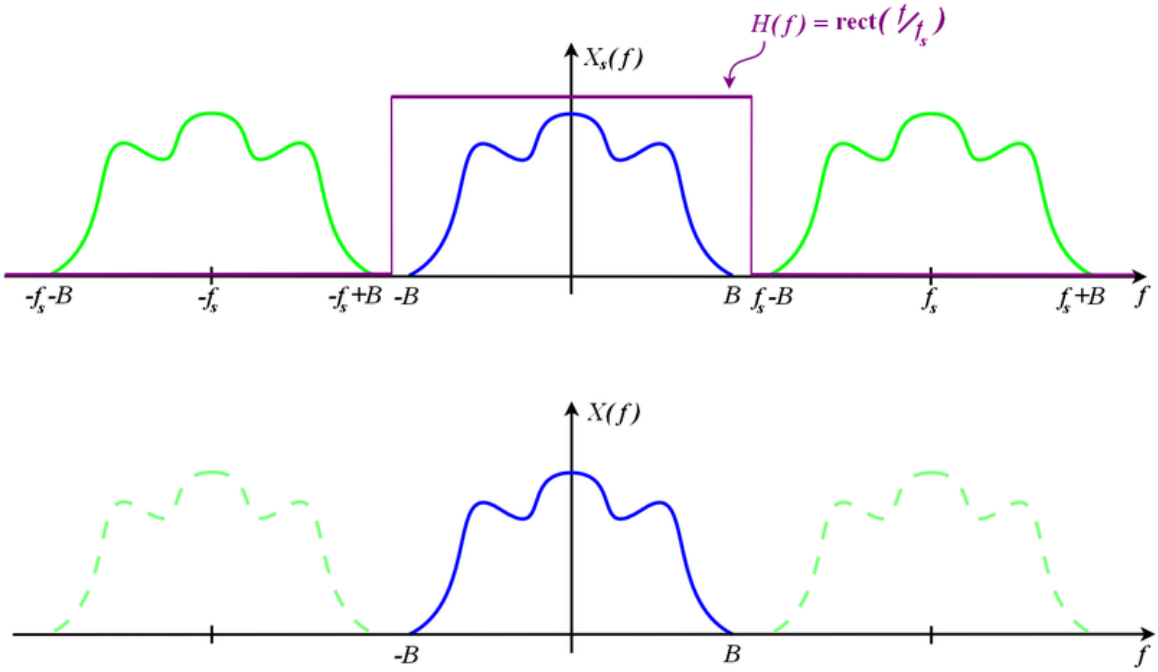


Fig.8: Spectrum, $X_s(f)$, of a properly sampled bandlimited signal (blue) and images (green) that do not overlap. A *brick-wall* low-pass filter, $H(f)$, removes the images, leaves the original spectrum, $X(f)$, and recovers the original signal from the samples.

From Figures 3 and 8, it is apparent that when there is no overlap of the copies (aka "images") of $X(f)$, the $k = 0$ term of $X_s(f)$ can be recovered by the product:

$$X(f) = H(f) \cdot X_s(f), \quad \text{where:}$$
$$H(f) = \begin{cases} 1 & |f| < B \\ 0 & |f| > f_s - B. \end{cases}$$

$H(f)$ need not be precisely defined in the region $[B, f_s - B]$ because $X_s(f)$ is zero in that region. However, the worst case is when $B = f_s/2$, the Nyquist frequency. A function that is sufficient for that and all less severe cases is:

$$H(f) = \operatorname{rect}\left(\frac{f}{f_s}\right) = \begin{cases} 1 & |f| < \frac{f_s}{2} \\ 0 & |f| > \frac{f_s}{2}, \end{cases}$$

where $\operatorname{rect}(u)$ is the rectangular function.

Therefore:

$$
\begin{aligned}
X(f) &= \operatorname{rect}\left(\frac{f}{f_s}\right) \cdot X_s(f) \\
&= \operatorname{rect}(Tf) \cdot T \sum_{n=-\infty}^{\infty} x(nT)\ e^{-i2\pi nTf} && \text{(from } \textbf{Eq.1}\text{, above).} \\
&= T \sum_{n=-\infty}^{\infty} x(nT) \cdot \operatorname{rect}(Tf) \cdot e^{-i2\pi nTf}.
\end{aligned}
$$

The original function that was sampled can be recovered by an inverse Fourier transform:

$$
\begin{aligned}
x(t) &= \mathcal{F}^{-1}\left\{ T \sum_{n=-\infty}^{\infty} x(nT) \cdot \operatorname{rect}(Tf) \cdot e^{-i2\pi nTf} \right\} \\
&= T \sum_{n=-\infty}^{\infty} x(nT) \cdot \underbrace{\mathcal{F}^{-1}\left\{ \operatorname{rect}(Tf) \cdot e^{-i2\pi nTf} \right\}}_{\frac{1}{T}\cdot\operatorname{sinc}\left(\frac{t-nT}{T}\right)} \\
&= \sum_{n=-\infty}^{\infty} x(nT) \cdot \operatorname{sinc}\left(\frac{t-nT}{T}\right),
\end{aligned}
$$

which is the Whittaker–Shannon interpolation formula. It shows explicitly how the samples, $x(nT)$, can be combined to reconstruct $x(t)$.

- From Figure 8, it is clear that larger-than-necessary values of f_s (smaller values of T), called *oversampling*, have no effect on the outcome of the reconstruction and have the benefit of leaving room for a *transition band* in which $H(f)$ is free to take intermediate values. Undersampling, which causes aliasing, is not in general a reversible operation.
- Theoretically, the interpolation formula can be implemented as a low pass filter, whose impulse response is $\operatorname{sinc}(t/T)$ and whose input is $\sum_{n=-\infty}^{\infty} x(nT) \cdot \delta(t-nT)$, which is a Dirac comb function modulated by the signal samples. Practical digital-to-analog converters (DAC) implement an approximation like the zero-order hold. In that case, oversampling can reduce the approximation error.

Shannon's original proof

The original proof presented by Shannon is elegant and quite brief, but it offers less intuitive insight into the subtleties of aliasing, both unintentional and intentional. Quoting

Shannon's original paper, which uses *f* for the function, *F* for the spectrum, and *W* for the bandwidth limit:

> Let $F(\omega)$ be the spectrum of $f(t)$. Then
> $$f(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} F(\omega)e^{i\omega t}\,d\omega$$
> $$= \frac{1}{2\pi}\int_{-2\pi W}^{2\pi W} F(\omega)e^{i\omega t}\,d\omega$$
> since $F(\omega)$ is assumed to be zero outside the band W. If we let
> $$t = \frac{n}{2W}$$
> where n is any positive or negative integer, we obtain
> $$f\left(\frac{n}{2W}\right) = \frac{1}{2\pi}\int_{-2\pi W}^{2\pi W} F(\omega)e^{i\omega\frac{n}{2W}}\,d\omega.$$
> On the left are values of $f(t)$ at the sampling points. The integral on the right will be recognized as essentially the nth coefficient in a Fourier-series expansion of the function $F(\omega)$, taking the interval $-W$ to W as a fundamental period. This means that the values of the samples $f(n / 2W)$ determine the Fourier coefficients in the series expansion of $F(\omega)$. Thus they determine $F(\omega)$, since $F(\omega)$ is zero for frequencies greater than W, and for lower frequencies $F(\omega)$ is determined if its Fourier coefficients are determined. But $F(\omega)$ determines the original function $f(t)$ completely, since a function is determined if its spectrum is known. Therefore the original samples determine the function $f(t)$ completely.

Shannon's proof of the theorem is complete at that point, but he goes on to discuss reconstruction via sinc functions, what we now call the Whittaker–Shannon interpolation formula as discussed above. He does not derive or prove the properties of the sinc function, but these would have been familiar to engineers reading his works at the time, since the Fourier pair relationship between rect (the rectangular function) and sinc was well known. Quoting Shannon:

> Let x_n be the nth sample. Then the function $f(t)$ is represented by:
> $$f(t) = \sum_{n=-\infty}^{\infty} x_n \frac{\sin\pi(2Wt - n)}{\pi(2Wt - n)}.$$

As in the other proof, the existence of the Fourier transform of the original signal is assumed, so the proof does not say whether the sampling theorem extends to bandlimited stationary random processes.

Sampling of non-baseband signals

As discussed by Shannon:

> A similar result is true if the band does not start at zero frequency but at some higher value, and can be proved by a linear translation (corresponding physically to single-sideband modulation) of the zero-frequency case. In this case the elementary pulse is obtained from sin(*x*)/*x* by single-side-band modulation.

That is, a sufficient no-loss condition for sampling signals that do not have baseband components exists that involves the *width* of the non-zero frequency interval as opposed to its highest frequency component.

A bandpass condition is that $X(f) = 0$, for all nonnegative f outside the open band of frequencies:

$$\left(\frac{N}{2} f_\mathrm{s}, \frac{N+1}{2} f_\mathrm{s}\right),$$

for some nonnegative integer N. This formulation includes the normal baseband condition as the case N=0.

The corresponding interpolation function is the impulse response of an ideal brick-wall bandpass filter (as opposed to the ideal brick-wall lowpass filter used above) with cutoffs at the upper and lower edges of the specified band, which is the difference between a pair of lowpass impulse responses:

$$(N+1)\operatorname{sinc}\left(\frac{(N+1)t}{T}\right) - N\operatorname{sinc}\left(\frac{Nt}{T}\right).$$

Other generalizations, for example to signals occupying multiple non-contiguous bands, are possible as well. Even the most generalized form of the sampling theorem does not have a provably true converse. That is, one cannot conclude that information is necessarily lost just because the conditions of the sampling theorem are not satisfied; from an engineering perspective, however, it is generally safe to assume that if the sampling theorem is not satisfied then information will most likely be lost.

Nonuniform sampling

The sampling theory of Shannon can be generalized for the case of nonuniform samples, that is, samples not taken equally spaced in time. The Shannon sampling theory for non-uniform sampling states that a band-limited signal can be perfectly reconstructed from its samples if the average sampling rate satisfies the Nyquist condition . Therefore, although

uniformly spaced samples may result in easier reconstruction algorithms, it is not a necessary condition for perfect reconstruction.

The general theory for non-baseband and nonuniform samples was developed in 1967 by Landau . He proved that, to paraphrase roughly, the average sampling rate (uniform or otherwise) must be twice the *occupied* bandwidth of the signal, assuming it is *a priori* known what portion of the spectrum was occupied. In the late 1990s, this work was partially extended to cover signals of when the amount of occupied bandwidth was known, but the actual occupied portion of the spectrum was unknown . In the 2000s, a complete theory was developed using compressed sensing. In particular, the theory, using signal processing language, is described in this 2009 paper . They show, among other things, that if the frequency locations are unknown, then it is necessary to sample at least at twice the Nyquist criteria; in other words, you must pay at least a factor of 2 for not knowing the location of the spectrum. Note that minimum sampling requirements do not necessarily guarantee stability.

Beyond Nyquist

The Nyquist–Shannon sampling theorem provides a sufficient condition for the sampling and reconstruction of a band-limited signal. When reconstruction is done via the Whittaker–Shannon interpolation formula, the Nyquist criterion is also a necessary condition to avoid aliasing, in the sense that if samples are taken at a slower rate than twice the band limit, then there are some signals that will not be correctly reconstructed. However, if further restrictions are imposed on the signal, then the Nyquist criterion may no longer be a necessary condition.

A non-trivial example of exploiting extra assumptions about the signal is given by the recent field of compressed sensing, which allows for full reconstruction with a sub-Nyquist sampling rate. Specifically, this applies to signals that are sparse (or compressible) in some domain. As an example, compressed sensing deals with signals that may have a low over-all bandwidth (say, the *effective* bandwidth *EB*), but the frequency locations are unknown, rather than all together in a single band, so that the passband technique doesn't apply. In other words, the frequency spectrum is sparse. Traditionally, the necessary sampling rate is thus $B / 2$. Using compressed sensing techniques, the signal could be perfectly reconstructed if it is sampled at a rate slightly greater than the $EB / 2$. The downside of this approach is that reconstruction is no longer given by a formula, but instead by the solution to a convex optimization program which requires well-studied but nonlinear methods.

Historical background

The **sampling theorem** was implied by the work of Harry Nyquist in 1928 ("Certain topics in telegraph transmission theory"), in which he showed that up to $2B$ independent pulse samples could be sent through a system of bandwidth B; but he did not explicitly consider the problem of sampling and reconstruction of continuous signals. About the

same time, Karl Küpfmüller showed a similar result, and discussed the sinc-function impulse response of a band-limiting filter, via its integral, the step response *Integralsinus*; this bandlimiting and reconstruction filter that is so central to the sampling theorem is sometimes referred to as a *Küpfmüller filter* (but seldom so in English).

The sampling theorem, essentially a dual of Nyquist's result, was proved by Claude E. Shannon in 1949 ("Communication in the presence of noise"). V. A. Kotelnikov published similar results in 1933 ("On the transmission capacity of the 'ether' and of cables in electrical communications", translation from the Russian), as did the mathematician E. T. Whittaker in 1915 ("Expansions of the Interpolation-Theory", "Theorie der Kardinalfunktionen"), J. M. Whittaker in 1935 ("Interpolatory function theory"), and Gabor in 1946 ("Theory of communication").

Other discoverers

Others who have independently discovered or played roles in the development of the sampling theorem have been discussed in several historical articles, for example by Jerri and by Lüke. For example, Lüke points out that H. Raabe, an assistant to Küpfmüller, proved the theorem in his 1939 Ph.D. dissertation; the term *Raabe condition* came to be associated with the criterion for unambiguous representation (sampling rate greater than twice the bandwidth).

Meijering mentions several other discoverers and names in a paragraph and pair of footnotes:

As pointed out by Higgins [135], the sampling theorem should really be considered in two parts, as done above: the first stating the fact that a bandlimited function is completely determined by its samples, the second describing how to reconstruct the function using its samples. Both parts of the sampling theorem were given in a somewhat different form by J. M. Whittaker [350, 351, 353] and before him also by Ogura [241, 242]. They were probably not aware of the fact that the first part of the theorem had been stated as early as 1897 by Borel [25]. As we have seen, Borel also used around that time what became known as the cardinal series. However, he appears not to have made the link [135]. In later years it became known that the sampling theorem had been presented before Shannon to the Russian communication community by Kotel'nikov [173]. In more implicit, verbal form, it had also been described in the German literature by Raabe [257]. Several authors [33, 205] have mentioned that Someya [296] introduced the theorem in the Japanese literature parallel to Shannon. In the English literature, Weston [347] introduced it independently of Shannon around the same time.

Several authors, following Black , have claimed that this first part of the sampling theorem was stated even earlier by Cauchy, in a paper [41] published in 1841. However, the paper of Cauchy does not contain such a statement, as has been pointed out by Higgins [135].

As a consequence of the discovery of the several independent introductions of the sampling theorem, people started to refer to the theorem by including the names of the aforementioned authors, resulting in such catchphrases as “the Whittaker-Kotel’nikov-Shannon (WKS) sampling theorem" [155] or even "the Whittaker-Kotel'nikov-Raabe-Shannon-Someya sampling theorem" [33]. To avoid confusion, perhaps the best thing to do is to refer to it as the sampling theorem, "rather than trying to find a title that does justice to all claimants" [136].

Why Nyquist?

Exactly how, when, or why Harry Nyquist had his name attached to the sampling theorem remains obscure. The term *Nyquist Sampling Theorem* (capitalized thus) appeared as early as 1959 in a book from his former employer, Bell Labs, and appeared again in 1963, and not capitalized in 1965. It had been called the *Shannon Sampling Theorem* as early as 1954, but also just *the sampling theorem* by several other books in the early 1950s.

In 1958, Blackman and Tukey cited Nyquist's 1928 paper as a reference for *the sampling theorem of information theory*, even though that paper does not treat sampling and reconstruction of continuous signals as others did. Their glossary of terms includes these entries:

> *Sampling theorem* (*of information theory*)
> Nyquist's result that equi-spaced data, with two or more points per cycle of highest frequency, allows reconstruction of band-limited functions.
> *Cardinal theorem* (*of interpolation theory*)
> A precise statement of the conditions under which values given at a doubly infinite set of equally spaced points can be interpolated to yield a continuous band-limited function with the aid of the function
> $$\frac{\sin(x - x_i)}{x - x_i}.$$

Exactly what "Nyquist's result" they are referring to remains mysterious.

When Shannon stated and proved the sampling theorem in his 1949 paper, according to Meijering "he referred to the critical sampling interval $T = 1/(2W)$ as the *Nyquist interval* corresponding to the band W, in recognition of Nyquist’s discovery of the fundamental importance of this interval in connection with telegraphy." This explains Nyquist's name on the critical interval, but not on the theorem.

Similarly, Nyquist's name was attached to *Nyquist rate* in 1953 by Harold S. Black:

> "If the essential frequency range is limited to B cycles per second, $2B$ was given by Nyquist as the maximum number of code elements per second that could be unambiguously resolved, assuming the peak interference is less half a quantum step. This rate is generally referred to as **signaling at the Nyquist rate** and $1/(2B)$

> has been termed a *Nyquist interval*." (bold added for emphasis; italics as in the original)

According to the OED, this may be the origin of the term *Nyquist rate*. In Black's usage, it is not a sampling rate, but a signaling rate.