Data Converters

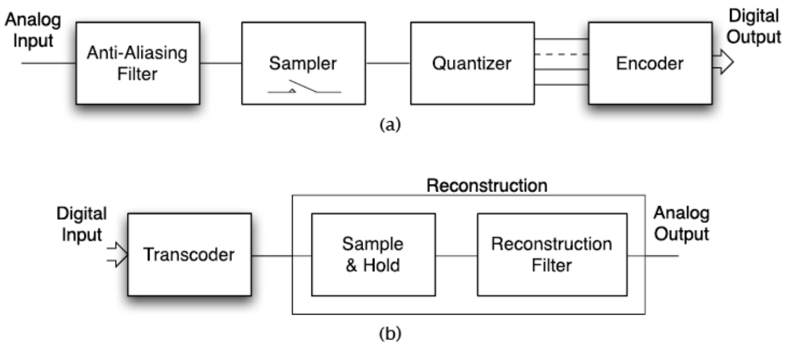
Introduction

Pietro Andreani
Dept. of Electrical and Information Technology
Lund University, Sweden

Overview

- Introduction
- The ideal data converter
- Sampling
- Amplitude quantization
- Quantization noise
- kT/C noise
- Discrete Fourier Transform and Fast Fourier Transform (FFT)
- The D/A converter
- *z*-transform

The ideal data converter

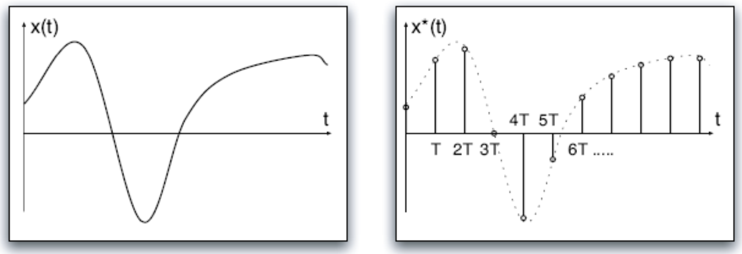


Transformation from continuous-amplitude and continuous-time to discrete-amplitude and discrete-time (and viceversa)

Sampling

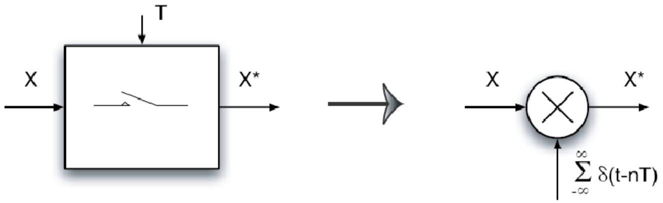
A sampler transforms a continuous-time signal into a sampled-data equivalent:

$$x^*(t) = x^*(nT) = \sum x(t) \cdot \delta(t - nT)$$



Only the value at the sampling instant matters

Sampling as a non-linear process



with Laplace/Fourier transform we obtain

$$L\left[\sum \delta(t-nT)\right] = \sum e^{-nsT} \quad \rightarrow \quad F\left[x^*(nT)\right] = \sum_{n=-\infty}^{\infty} x(nT)e^{-jn\omega T}$$

and alternatively

$$F\left[x^*(nT)\right] = \frac{1}{T}\sum_{n=-\infty}^{\infty} X(\omega - n\omega_s), \quad \omega_s = \frac{2\pi}{T}$$

Time and frequency domain –I

Sampled signal: $f^*(t) = f(t)\cdot \sum_{n=-\infty}^{\infty} \delta(t-nT)$

We define: $g(t) = \sum_{n=-\infty}^{\infty} \delta(t-nT) \quad \rightarrow \quad f^*(t) = f(t)\cdot g(t)$

It is known that the transform of the product of two functions of time is given by the convolution of the respective transforms:

$$f(t)\cdot g(t) \leftrightarrow F(\omega) * G(\omega)$$

Train of deltas with period T → can be represented with Fourier series as

$$g(t) = \sum_{n=-\infty}^{\infty} \delta(t-nT) = \sum_{k=-\infty}^{\infty} a_k e^{k\omega_s t + \phi_k}, \quad \omega_s = \frac{2\pi}{T}$$

Time and frequency domain –II

Using the complex Fourier series, we obtain:

$$a_k = \frac{1}{T}\int_0^T g(t)e^{jk\omega_s t}dt = \frac{1}{T}\int_0^T \delta(t)e^{jk\omega_s t}dt = \frac{1}{T}$$

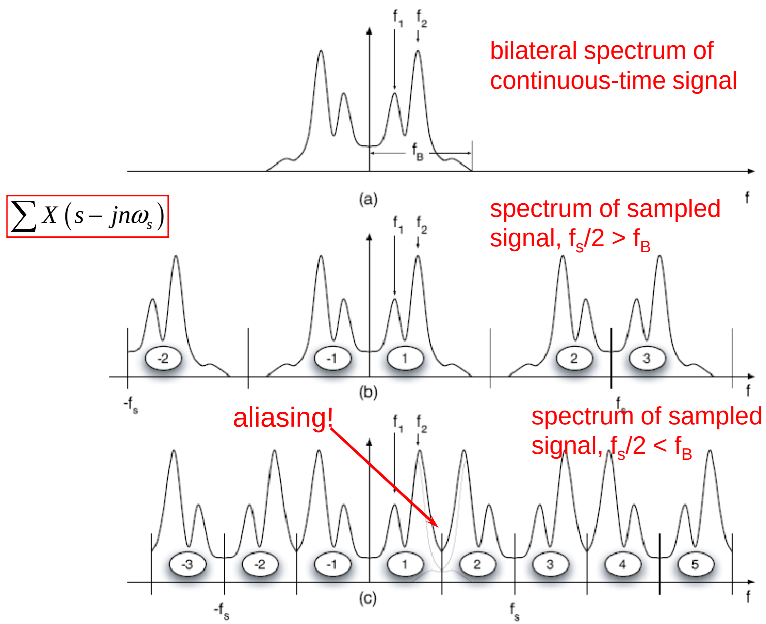
which means that the spectrum *G(ω)* of *g(t)* is a series of lines at frequencies $k\omega_s$, all with amplitude *1/T*. This means that we can write *G(ω)* as

$$G(\omega) = \frac{1}{T}\sum_{k=-\infty}^{\infty} \delta(\omega - k\omega_s)$$

Finally:

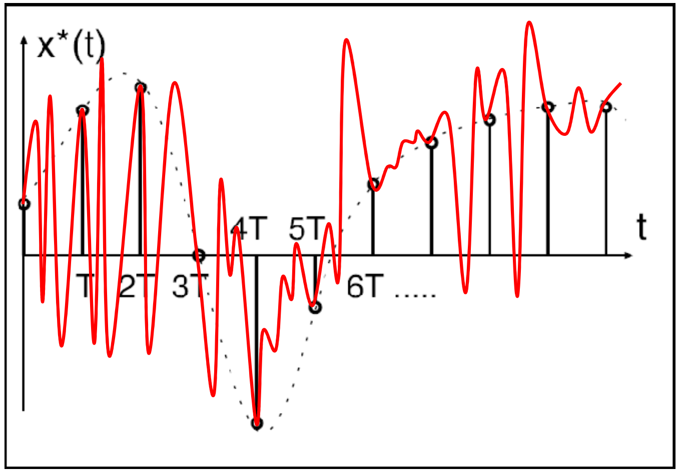
$$F^*(\omega) = F(\omega) * G(\omega) = F(\omega) * \frac{1}{T}\sum_{k=-\infty}^{\infty} \delta(\omega - k\omega_s) = \frac{1}{T}\sum_{k=-\infty}^{\infty} F(\omega - k\omega_s)$$

Is it possible to recover the original signal?

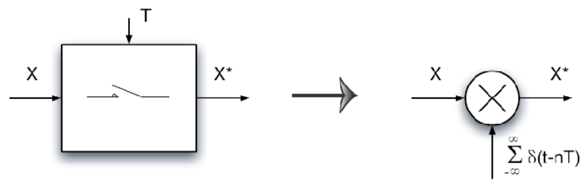


Aliasing

An infinite number of (higher frequency) signals can result in the same sampled data!



Nyquist theorem

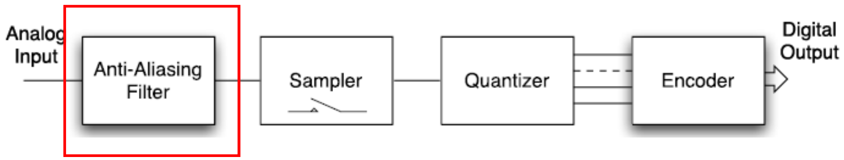


A band-limited signal x(t), whose spectrum X(jω) vanishes for $|\omega| > \omega_s/2$, is fully described by a uniform sampling x(nT), where $T=2\pi/\omega_s$. The band-limited signal x(t) is reconstructed by:

$$x(t) = \sum x(nT) \frac{\sin\left(\omega_s\left(t-nT\right)/2\right)}{\omega_s\left(t-nT\right)/2}$$

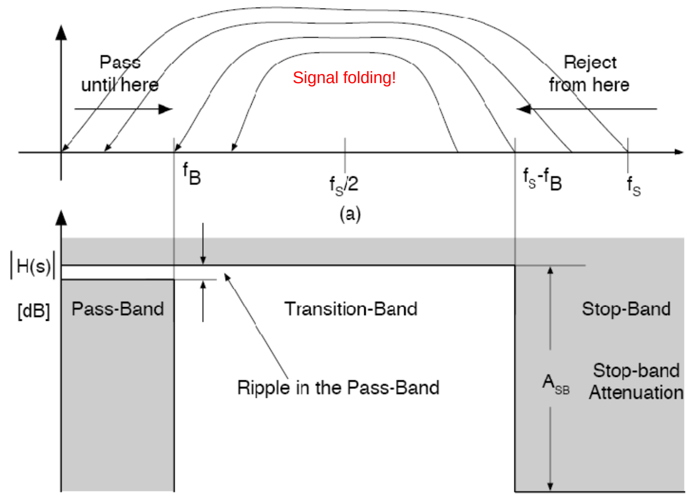
Half the sampling frequency, $f_s/2 = 1/2T$, is referred to as the Nyquist frequency. The frequency interval 0.. $f_s/2$ is referred to as the Nyquist band (or base-band), while the intervals $f_s/2..f_s$, $f_s..3f_s/2$ etc are called second, third etc Nyquist zones.

Anti-aliasing filter – I



- The anti-alias filter protects the information content of the signal
- Use an anti-aliasing filter in front of every quantizer to reject unwanted interferences from outside the band of interest

Anti-aliasing filter – II

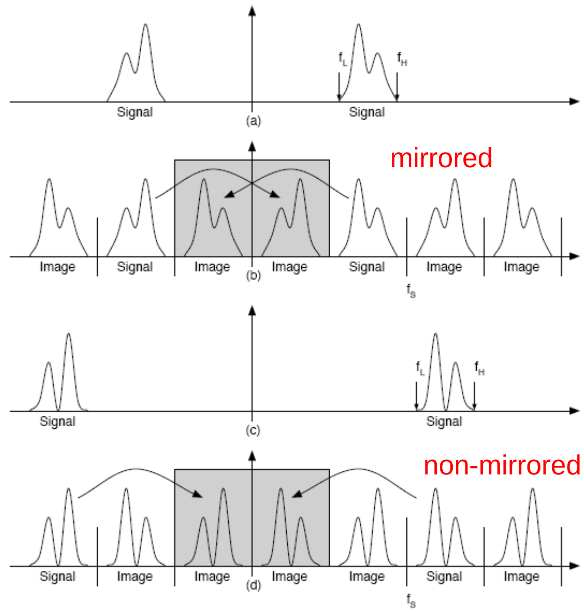


Complexity depends on band-pass ripple, transition-region steepness, and stop-band attenuation.

Under-sampling – I

- Exploit the base-band folding of signals at frequencies higher than the Nyquist limit ($f_s/2$) → brings high-frequency spectra into base band.
- Also known as harmonic sampling, base-band sampling, IF sampling, RF-to-digital conversion, etc
- Attractive as RF demodulator, but: bandwidth, distortion, and jitter performance must meet the specs at RF! Further, wide-band noise is also sampled!

Under-sampling – II

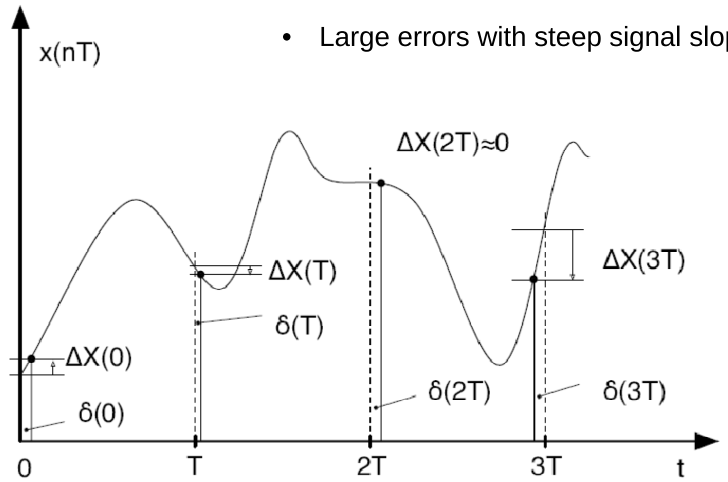


Under-sampling – III

- Also under-sampling requires an anti-aliasing filter
- Removes unwanted signals occurring at base-band or in any other Nyquist zone
- In the case of under-sampling, the anti-aliasing filter is a band-pass filter around the signal band

Sampling-time jitter

- Any clock signal is affected by jitter (noise on the “zero crossings”)
- Large errors with steep signal slopes



Sampling error caused by jitter

For a sine wave $x_{in}(t) = A\sin(\omega_{in} t)$, the jitter-induced error $\Delta x(nT)$ is

$$\Delta x(nT) = A\omega_{in}\delta(nT)\cos(\omega_{in} nT)$$

Assume that $\delta(nT)$ is the sampling of a random variable $\delta_{ji}(t)$; we obtain:

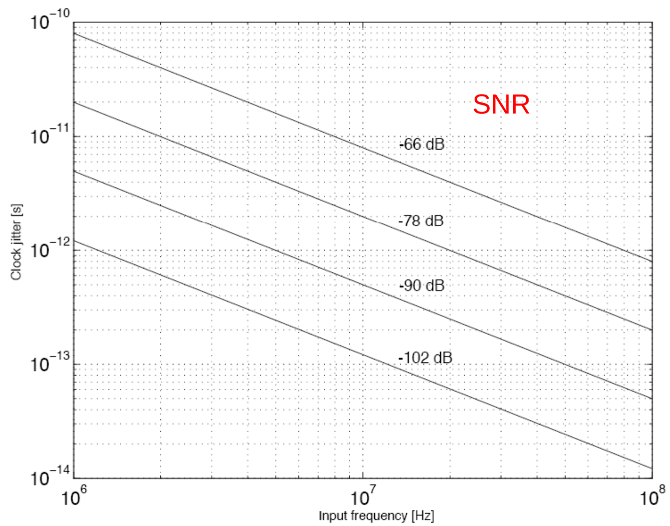
$$\left\langle x_{ji}^2(t)\right\rangle = \left\langle\left[A\omega_{in}\cos(\omega_{in} nT)\right]^2\right\rangle\left\langle\delta^2(nT)\right\rangle = \frac{A^2\omega_{in}^2}{2}\left\langle\delta^2(nT)\right\rangle$$

The jitter-limited SNR is therefore:

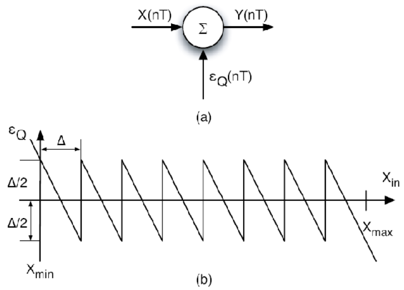
$$SNR = 10\log\frac{S}{N} = -20\log\left(\omega_{in}\left\langle\delta(nT)\right\rangle\right)$$

Jitter may be the limiting factor in data conversion: if SNR=90dB and f_{in}=100MHz, then the clock jitter must be below 50fs

Maximum jitter vs. input frequency and SNR



Amplitude quantization – quantization error



$$Y = X_{in} + \varepsilon_Q$$

$$n\Delta < X_{in} < (n+1)\Delta$$

$$-\Delta/2 \le \varepsilon_Q \le \Delta/2$$

$$\Delta = \frac{X_{FS}}{M}$$

$$X_{FS} = X_{\max} - X_{\min}$$

$$M = \text{\# of quantization levels}$$

$$M = 2^N$$

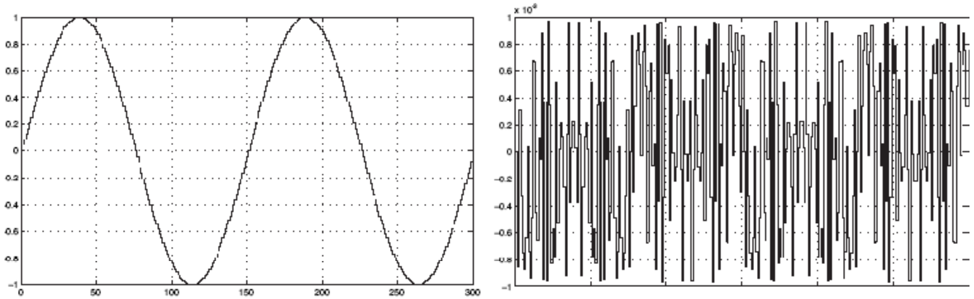
$$N = \text{\# of bits}$$

The quantization error is a form of data corruption fundamentally unavoidable in data conversion (unless N is infinite)

Quantization noise

- The quantization error can be viewed as a form of noise
- Frequent code transitions decorrelate successive samples of the quantization error, spreading its spectrum and making it resemble white noise
- Necessary conditions for this very convenient assumptions are:
  1. All quantization levels are exercised with equal probability (true if input signal is large)
  2. A large number of quantization levels are used (usually true, except for ΔΣ converters)
  3. The quantization steps are uniform (usually true)
  4. The quantization error is not correlated with the input (usually true)

An example



quantized signal
(10 bits)

quantization error
(f_s=150f_{in})

Notice that in this case (f_s multiple of f_{in}) the quantization error is obviously correlated with the input sine wave

Quantization noise – properties

The quantization error is confined between -Δ/2 and Δ/2, and may assume any value between -Δ/2 and Δ/2 with equal probability *p* →

$$p\left(\varepsilon_Q\right)=\begin{cases}\frac{1}{\Delta} & \text{if } -\frac{\Delta}{2}<\varepsilon_Q<\frac{\Delta}{2}\\ 0 & \text{otherwise}\end{cases}$$

The time average power becomes:

$$P_Q=\int_{-\infty}^{\infty}\varepsilon_Q^2\,p\left(\varepsilon_Q\right)d\varepsilon_Q=\int_{-\Delta/2}^{\Delta/2}\frac{\varepsilon_Q^2}{\Delta}d\varepsilon_Q=\frac{\Delta^2}{12}$$

Quantization noise and SNR

Within a dynamic range of X_{FS}, the power of a maximum-amplitude sine wave is

$$P_{\sin}=\frac{1}{T}\int_0^T\frac{X_{FS}^2}{4}\sin^2\left(\omega t\right)dt=\frac{X_{FS}^2}{8}=\frac{\left(2^n\Delta\right)^2}{8}$$

and for a triangular wave is $$P_{trian}=\frac{X_{FS}^2}{12}=\frac{\left(2^n\Delta\right)^2}{12}$$

$$SNR_{\sin}=10\cdot\log\frac{P_{\sin}}{P_Q}=\left(6.02\cdot n+1.76\right)dB$$

The maximum SNR becomes:

$$SNR_{trian}=10\cdot\log\frac{P_{trian}}{P_Q}=\left(6.02\cdot n\right)dB$$

Equivalent number of bits (ENOB)

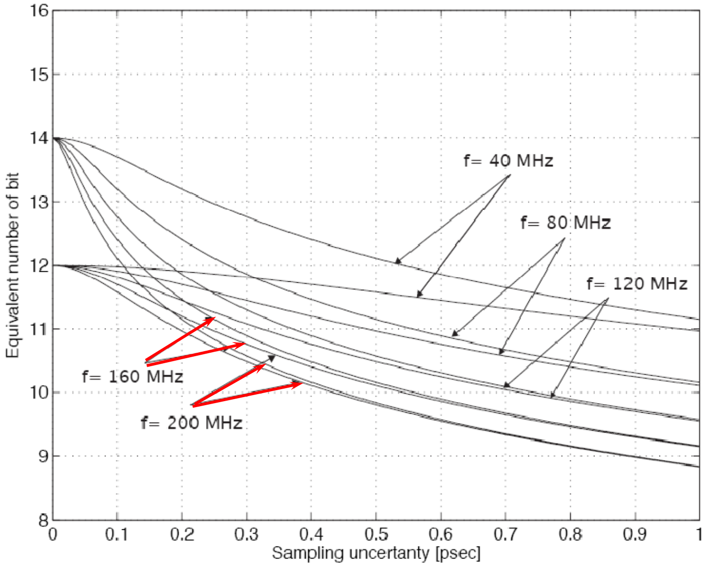
The effective SNR is the real measure of the resolution of a data converter. Measured in bits, it is called the ENOB:

$$ENOB_{\sin}=\frac{SNR_{eff}\big|_{dB}-1.76}{6.02}$$

If, for example, there is a sampling jitter δ_{ji}, the effective SNR becomes

$$SNR_{eff}=10\log\frac{X_{FS}^2/8}{\Delta^2/12+\left(\omega_{in}\delta_{ji}X_{FS}\right)^2/8}=10\log\frac{1}{8/12\cdot2^{-2N}+\left(\omega_{in}\delta_{ji}\right)^2}$$

ENOB with jitter



Quantization noise – noise power spectrum

Laplace transform of the noise auto-correlation function:

$$p_\varepsilon(f) = \int_{-\infty}^{\infty} R_\varepsilon(\tau) e^{-j2\pi f\tau} d\tau = \sum_{-\infty}^{\infty} R_\varepsilon(nT) e^{-j2\pi fnT}$$

We assume that $R_\varepsilon(\tau)$ goes rapidly to zero for |*n*|>0, i.e., that only $R_\varepsilon(0)$ is different from zero. Thus, $R_\varepsilon(\tau)$ is a delta in the time domain, and a constant-valued function in the frequency domain. The unilateral power spectrum becomes

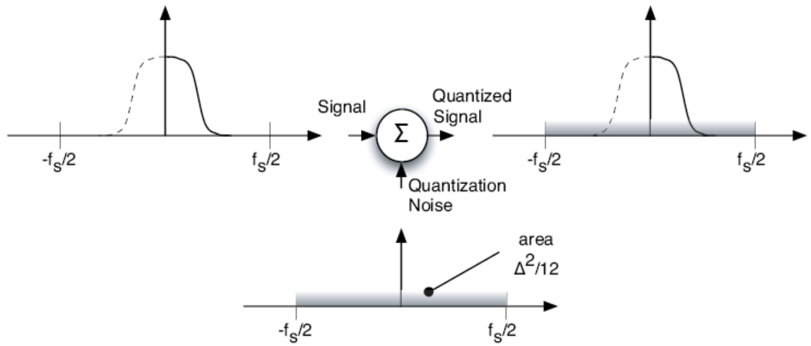
$$p_\varepsilon(f) = \frac{\Delta^2}{6 f_s}$$

since the total noise power $\Delta^2/12$ is uniformly spread over the unilateral Nyquist interval $0 \dots f_s/2$.

Bilateral noise power spectrum

In a bilateral representation, we have (in order to keep the total noise power constant):

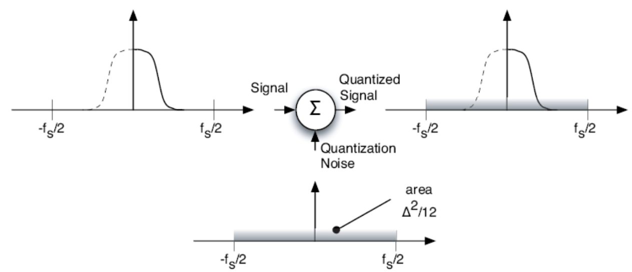
$$p_\varepsilon(f) = \frac{\Delta^2}{12 f_s}$$



A remark about quantization noise

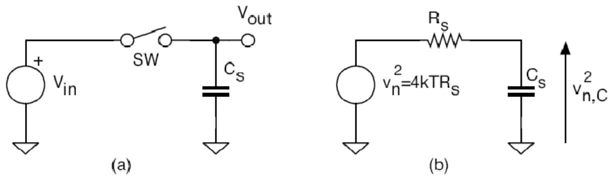
We have seen that the sampled signal is band-limited between $-f_s/2$ and $f_s/2$ (of course, there is an infinite number of replicas centered around multiples of $\pm f_s$)

After quantization, the original band-limited signal is expressed as the sum of the quantized signal and quantization noise; therefore, the bandwidth of the quantization noise must be the same as that of the original sampled signal, i.e., between $-f_s/2$ and $f_s/2$



kT/C noise

An unavoidable limit to the SNR comes from the thermal noise associated with a sampling switch, and depends only on the sampling capacitance (and not on the switch resistance!)



$$\overline{v_{n,C_s}^2}(\omega) = \frac{4kTR_s}{1+(\omega R_s C_s)^2}$$

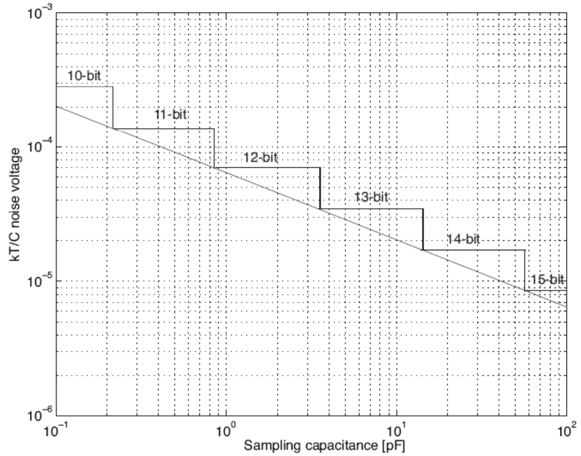
$$P_{n,C_s} = \int_0^\infty \frac{4kTR_s}{1+(\omega R_s C_s)^2} df = 4kTR_s \int_0^\infty \frac{1}{1+(\omega R_s C_s)^2} df = \frac{kT}{C_s}$$

$$C_s = 1\text{pF} \quad \rightarrow \quad \sqrt{kT/C_s} = 64.5\mu\text{V}$$

ENOB limitations from kT/C noise

Thermal noise vs. sampling capacitance, compared to the quantization noise $\Delta/\sqrt{12}$ (reference = 1V). If both noise sources have the same power, the SNR decreases by 3dB = 0.5LSB

$$\sqrt{\frac{kT}{C_s}}$$



Thermal noise: an example – I

A pipeline A/D uses a cascade of two S/H circuits in the first stage. The jitter is 1ps, X_{FS} is 1V, f_{in} is 5MHz. Determine the minimum C_s for ENOB = 12 bits.

The q-noise power is $\Delta^2/12$. We assume that an extra 50% total noise is ok (SNR decreases by 1.76dB = 0.29LSB). The noise budget for kT/C and jitter noise together is then:

$$v_{n,budget}^2 = \frac{1}{2} v_Q^2 = \frac{1}{2} \cdot \frac{\Delta^2}{12} = \frac{X_{FS}^2}{24 \cdot 2^{2N}} = \frac{1}{24 \cdot 2^{24}} = 2.48 \cdot 10^{-9}\,\text{V}^2$$

The jitter affects only the first stage (the second stage samples the signal held by the first). The jitter noise is

$$v_{n,ji}^2 = \frac{1}{2}\left(\frac{X_{FS}}{2} 2\pi f_{in} \delta_{ji}\right)^2 = 1.23 \cdot 10^{-10}\,\text{V}^2$$

Thermal noise: an example – II

Thus, the total noise power that can be tolerated by the samplers is

$$v_{n,samplers}^2 = 2.36 \cdot 10^{-9}\,\text{V}^2$$

Assuming equal capacitance in both S/H circuits, the noise for each is

$$v_{n,C}^2 = 1.18 \cdot 10^{-9}\,\text{V}^2$$

and the minimum sampling capacitance value is

$$C_s = \frac{kT}{v_{n,C}^2} = \frac{4.14 \cdot 10^{-21}}{1.12 \cdot 10^{-9}} = 3.51\text{pF}$$

Notice that the jitter noise establishes a maximum achievable resolution independently of C_s; this limit is, in this case, 14.7 bits

Discrete and Fast Fourier Transform – I

The spectrum of a sampled-data signal is estimated using

$$L\left[x^*(nT)\right] = \sum x(nT)\cdot e^{-nsT}$$

$$F\left[x^*(nT)\right] = X^*(j\omega) = \sum_{-\infty}^{\infty} x(nT)\cdot e^{-j\omega nT}$$

This requires an infinite number of steps; a convenient approximation is the Discrete Fourier Transform (DFT), which assume the input to be N-periodic:

$$X(f_k) = \sum_{n=0}^{N-1} x(nT)\cdot e^{-j2\pi kn/(N-1)}$$

Spectrum made of N lines (bins) located at $f_k = \dfrac{k}{T(N-1)}, \quad 0 \le k \le N-1$

Discrete and Fast Fourier Transform – II

The DFT is (of course) a complex function:

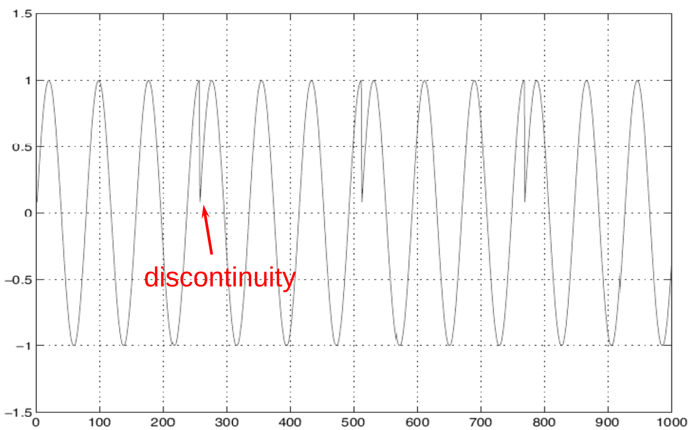
$$\left|X(f_k)\right| = \sqrt{\mathrm{Re}^2\left(X(f_k)\right) + \mathrm{Im}^2\left(X(f_k)\right)}$$

$$\varphi\left(X(f_k)\right) = \arctan\frac{\mathrm{Im}\left(X(f_k)\right)}{\mathrm{Re}\left(X(f_k)\right)}$$

The DFT algorithm requires N^2 computations. For long sample series, the Fast Fourier Transform (FFT) algorithm is more effective, as it only requires $N\cdot\log_2(N)$ computations. In the FFT case, N must be a power of 2 (which is therefore the usual choice).

Periodicity assumption in DFT/FFT

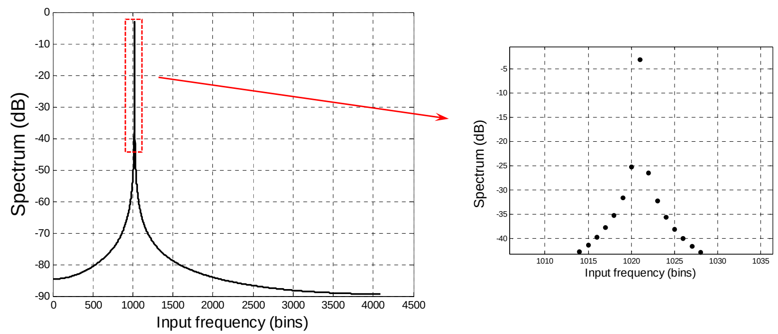
The DFT and FFT assume the input is N-periodic. Real signals are never periodic, and the N-periodicity assumption lead to discontinuities between the last and first samples of successive sequences.



Spectral leakage

If the number of sampled signal periods is NOT an integer number, as in the example below → the Fourier coefficients are non-zero at all harmonics!

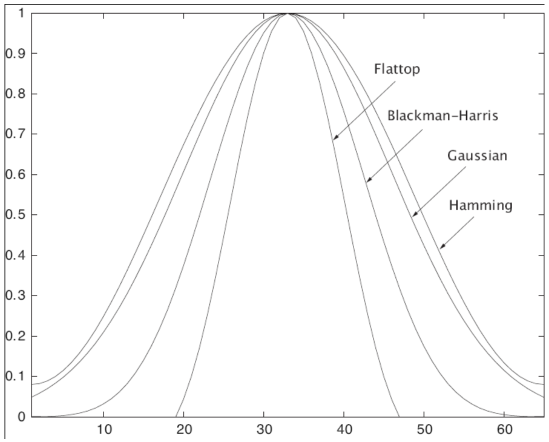
This is referred to as *spectral leakage* → it looks as if the signal has long skirts → this is NOT to be confused with a proper noise floor!



Windowing

Windowing addresses this problem by tapering the endings of the series

$$x_w(k) = x(k) \cdot W(k)$$



Useful tips

When using DFT or FFT, make sure that the sequence of the samples is N-periodic; otherwise use windowing

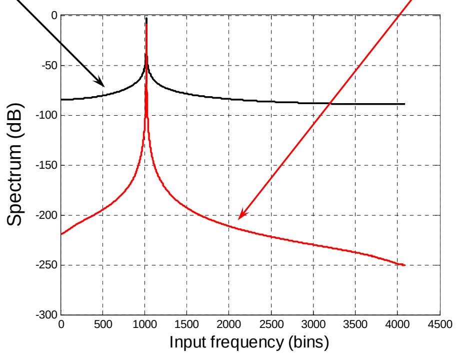
With sine-wave inputs, avoid repetitive patterns in the sequence: the ratio between the sine-wave period and the sampling period should be a prime number (otherwise, noise will appear concentrated in only a few bins, in a non-white fashion)

Windowing example – Hann window

The Hann window greatly reduces the impact of spectral leakage

Spectrum with rectangular window (previous example)

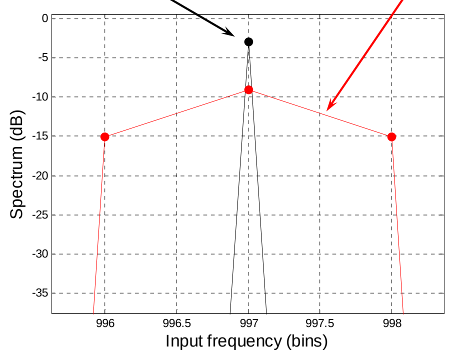
Spectrum with Hann window (would be below a reasonable noise floor)



Hann window – signal bins

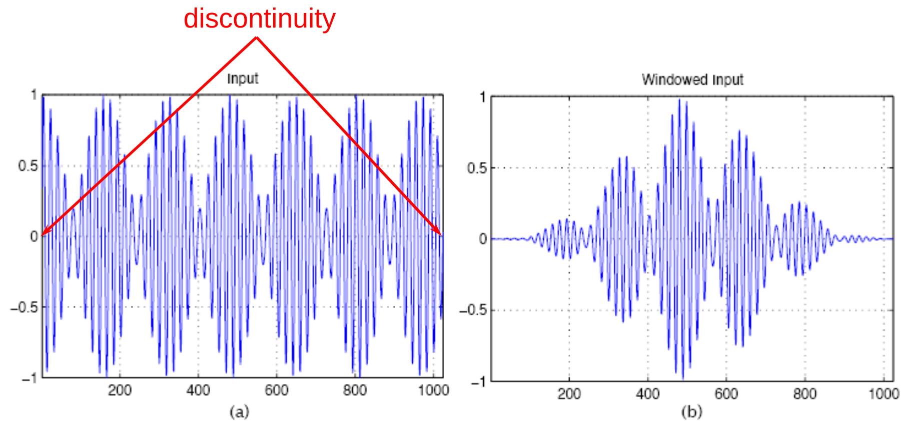
Rectangular window + integer number of signal periods → the signal is found over one frequency bin (as expected)

Hann → the signal is spread over three bins → signal + two sidebands
Further, the amplitude of the central bin is 6dB lower than with rectangular windowing



More on windowing – I

Time-domain input (two sine-waves) before and after windowing (Blackman window)

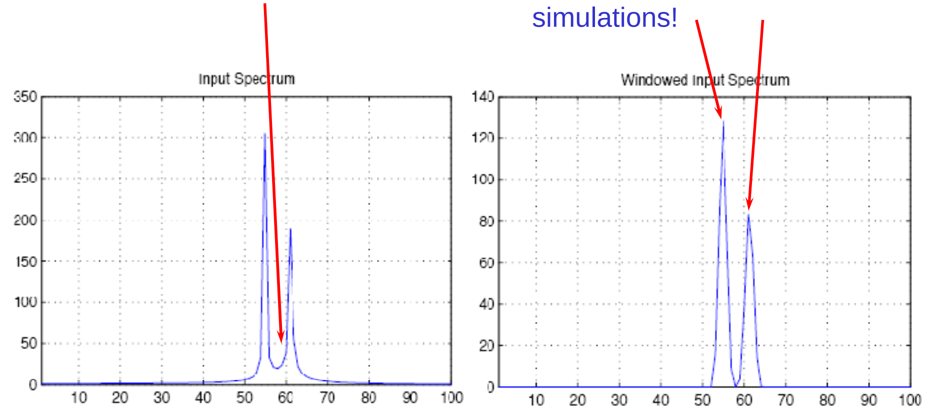


More on windowing – II

FFT on signal in previous slide:

Rectangular window → spectral leakage (Notice: amplitude is NOT in dB)

Blackman (or other) window → reduced amplitude, but constant ratio → ok for SNR simulations!



Is windowing avoidable?

Windowing is an amplitude modulation of the input → produces undesired effects as well

A spike at the beginning/end of the sequence is completely masked

Windowing gives rise to spectral leakage by its own (i.e., if the input is a sine wave, the spectrum is not a pure tone)

The SNR measurement can be accurate, but see the previous point

For input sine waves → use coherent sampling, for which an integer number *k* of signal periods fits into the sampling window; moreover, *k* must be chosen with care

Constraints on *k*

Coherent sampling: $m \cdot T_s = k \cdot T_{in}$ (*sampling period*: T_s; *signal period*: T_{in})

If: $m = a \cdot b$ and $k = a \cdot c$ → $a \cdot (b \cdot T_s) = a \cdot (c \cdot T_{in})$

i.e., the q-noise sequence repeats itself a times, identically → q-noise is not white

To avoid this, *m* and *k* should be prime to each other; in particular, if *m* is a power of *N*, *k* should be odd:

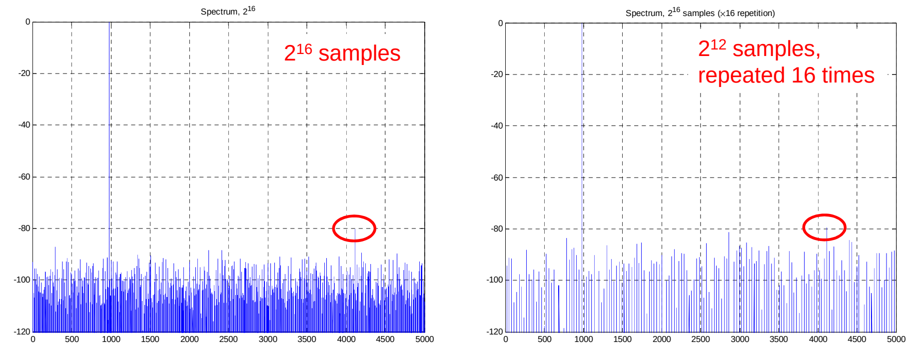
$$f_{in} = \frac{k}{2^N} f_s$$

sine-wave freq.: f_{in}; odd number: k; sampling freq.: f_s; # of samples: 2^N

k and q-noise

If *k* in $f_{in} = f_s \cdot k/2^N$ is not odd, the quantization noise repeats itself in exactly the same way two or more times across the sampling sequence → *q*-noise is not white, but shows repetitive patterns → q-noise is concentrated at certain frequencies (bins)!

The two simulations below use the same amount of samples, but notice how in the plot to the right the q-noise is much less white, with much higher peaks (possibly hiding a higher harmonic)



About the FFT – I

The FFT of an N-sample sequence is made of N discrete lines equally spaced in the frequency interval $0..f_s$

The spectrum in the second Nyquist zone ($f_s/2..f_s$) mirrors the base band → only half of the FFT computations are represented ($0..f_s/2$) → N/2 spectral lines (bins) in the base band

Each bin gives the spectral power falling in the bandwidth f_s/N and centered around the line itself

Thus, the FFT operates like a spectrum analyzer with N channels each with bandwidth f_s/N

If the number of points in the series increases, the channel bandwidth of the equivalent spectrum analyzer decreases and each channel will contain less noise power

About the FFT – II

An M-bit quantization gives a noise power of $\frac{X_{FS}^2}{12 \cdot 2^{2M}}$

The power of the full-scale sine wave is $\frac{X_{FS}^2}{8}$

The FFT of the quantization noise is, on average, a factor $\frac{3}{2} \cdot 2^{2M} \cdot \frac{N}{2}$ below the full scale, where N is the number of points in the FFT series. The overall noise floor becomes

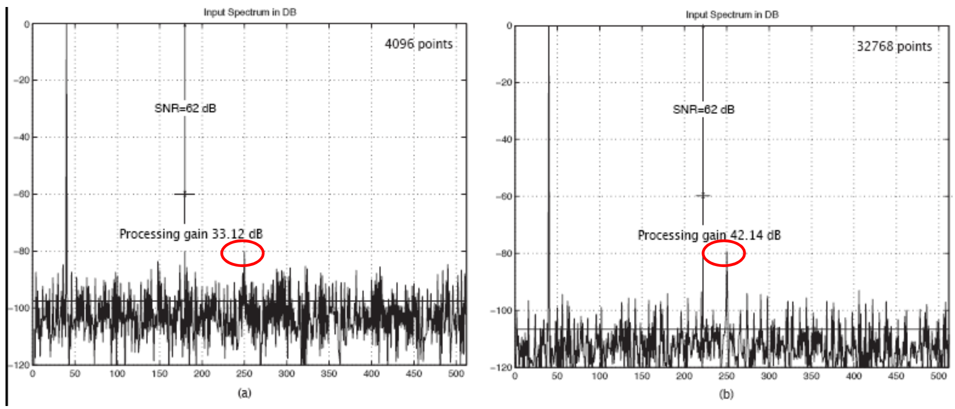
$$x_{noise}^2\Big|_{dB} = P_{sig} - 1.76 - 6.02 \cdot M - 10 \cdot \log\frac{N}{2}$$

The term $10 \cdot \log\frac{N}{2}$ is called *processing gain* of the FFT

About the FFT – III

If the length of the input series increases by a factor 2, the floor of the FFT noise spectrum diminishes by 3dB; however, tones caused by harmonic distortion do not change.

A long-enough input series is able to reveal a small distortion tone above the noise floor.



Coding schemes – I

USB – Unipolar Straight Binary: simplest binary scheme, used for unipolar signals. The lowest level ($-V_{ref}$ +1/2 V_{LSB}) is represented with all zeros (00….0), and the highest (V_{ref} -1/2 V_{LSB}) with all ones (11..1). The quantization range is ($-V_{ref}$ .. + V_{ref}) .

CSB – Complementary Straight Binary: also for unipolar signals, opposite of USB. (00..0) represents full scale, and (11..1) the first quantization level.

BOB – Bipolar Offset Binary: for bipolar (i.e., both positive and negative). The MSB is the sign of the signal (1 for positive and 0 for negative). Therefore, (00…0) represents the full negative scale, (11..1) is the full positive scale, and (01..11) is the zero crossing.

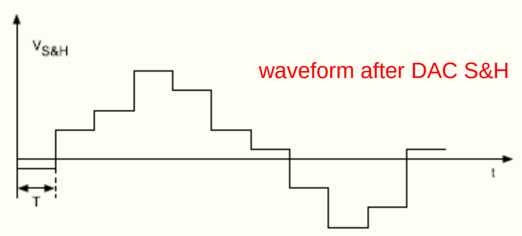
Coding schemes – II

COB – Complementary Offset Binary: complementary to BOB. The zero crossing is therefore at (10…00).

BTC – Binary Two's Complement: one of the most used coding schemes. The MSB indicates the sign in a complemented way: it is 0 for positive and 1 for negative inputs. Zero crossing is at (0..00). For positive signals, the digital code increases for increasing analog values, and the positive full scale is therefore (01….11). For negative signals, the digital code is the two's complement of the positive counterpart, which means that (10…00) is the negative full scale. BCT coding is standard in microprocessors and digital audio, and is in general used in the implementation of mathematical algorithms.

BTC – Complementary Two's Complement: complementary to BTC. Negative full scale is (01..11) and positive full scale is (10..00).

D/A conversion



The (ideal) reconstruction filter smooths the staircase-like waveforms, removing the high-frequency components:

$$H_{R,ideal}(f)=\begin{cases}1 & \text{for } -\frac{f_s}{2}<f<\frac{f_s}{2}\\ 0 & \text{otherwise}\end{cases}$$

In the time domain:

$$r(t)=\frac{\sin(\omega_s t/2)}{\omega_s t/2}$$

"sinc" function

It is well known that this transfer function is not implementable, as it is not causal (it is different from zero for t<0)

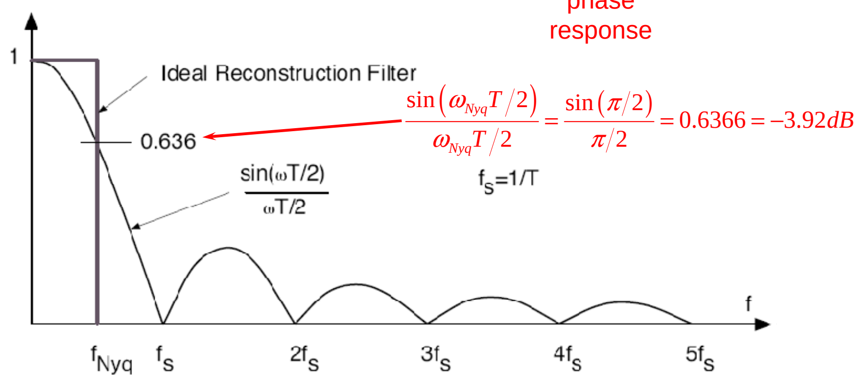
Real reconstruction

The S&H transfer function is:

$$H_{S/H}(s)=\frac{1-e^{-sT}}{s\tau} \rightarrow H_{S/H}(j\omega)=j\frac{T}{\tau}e^{-j\omega T/2}\frac{\sin(\omega T/2)}{\omega T/2}$$

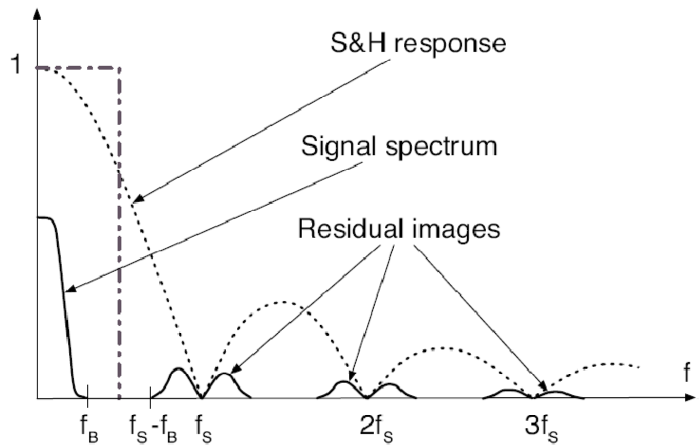
τ = gain factor

phase response



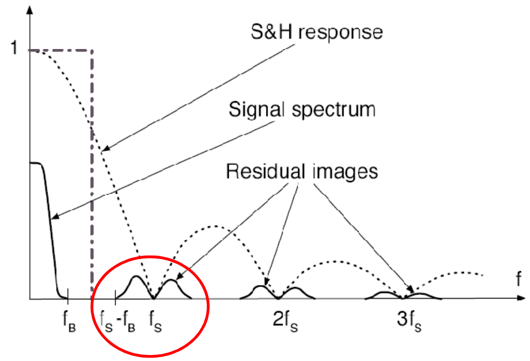
Spectrum after S/H

Rule of thumb: the reconstruction must use an in-band *x/sin(x)* compensation if the signal band is above ¼ of Nyquist



Attenuation of images

The transmission zeros of the sinc function shape the signal images at multiples of f_S. As an example, if the signal band is f_B=f_S/20, the value of the sinc at f=19/20f_S is 0.0498, yielding a 26dB suppression, alleviating the requirements on the reconstruction filter.



z-transform

Time-discrete counterpart of the Laplace transform:

$$Z\{x(nT)\}=\sum_{-\infty}^{\infty}x(nT)z^{-n}$$

Linear:

$$Z\{a_1\cdot x_1(nT)+a_2\cdot x_2(nT)\}=a_1\cdot Z\{x_1(nT)\}+a_2\cdot Z\{x_2(nT)\}$$

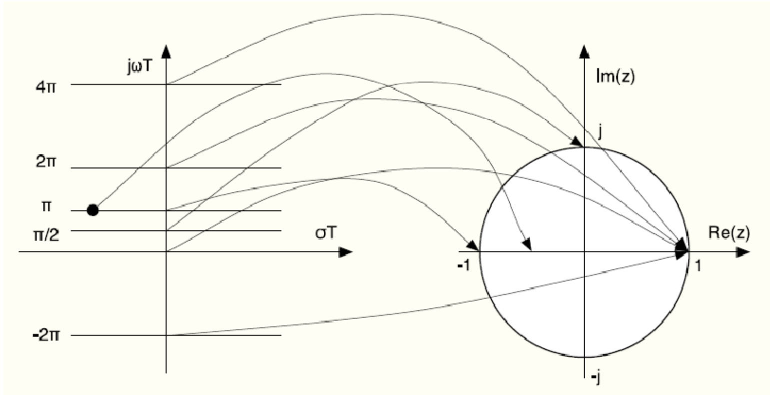
The z-transform of a delayed signal is:

$$Z\{x(nT-kT)\}=\sum_{-\infty}^{\infty}x(nT-kT)z^{-(n-k)}z^{-k}=X(z)z^{-k}$$

Mapping between s-plane and z-plane

A sampled-date system is stable if all the poles of its z transfer function are inside the unity circle. The frequency response of the system is the z transfer function calculate on the unity circle.

$$z\rightarrow e^{sT}$$



z-domain integration (I)

Time-continuous integration:

$$H_I(s) = \frac{1}{s\tau}$$

Approximate integration → the time-continuous equation is replaced by a finite increment; the equation

$$y(nT+T) = y(nT) + \int_{nT}^{nT+T} x(t)dt$$

can be discretized in three (actually more) ways:

$$1)\quad y(nT+T) = y(nT) + x(nT)\cdot T$$

$$2)\quad y(nT+T) = y(nT) + x(nT+T)\cdot T$$

$$3)\quad y(nT+T) = y(nT) + \frac{x(nT)+x(nT+T)}{2}T$$

z-domain integration (II)

Via the z-transform, we obtain:

$$1)\quad zY = Y + X\cdot T$$

$$2)\quad zY = Y + zX\cdot T$$

$$3)\quad zY = Y + X\frac{1+z}{2}T$$

Which results in three different approximate expressions for the z transfer function of the integral:

$$H_{I,Forward} = T\frac{1}{z-1}$$

$$H_{I,Backward} = T\frac{z}{z-1}$$

$$H_{I,Bilinear} = T\frac{z+1}{2(z-1)}$$

z-domain integration (III)

The corresponding mappings between s-plane and z-plane are

$$sT_{Forward} \to \frac{z-1}{T}$$

$$sT_{Backward} = \frac{z-1}{Tz}$$

$$sT_{Bilinear} = \frac{z-1}{2T(z+1)}$$

to be compared with the ideal mapping

$$s \to \frac{1}{T}\ln(z)$$

Forward → critical for stability (a stable system can become unstable)

Backward → problematic for conserving performance (and an unstable system can become stable)

Bilinear → best (stable → stable; unstable → unstable), but rarely used in data conversion

Remarks about *s*-to-*z* plane mappings

- Euler forward is never used because of the instability hazard
- In general, the mentioned mapping have been relevant in the design of sampled-data circuits such as switched-capacitor (SC) circuits
- Nowadays, the issue of *s*-plane to *z*-plane mapping is most relevant in ΔΣ converters, which merge filtering with data conversion
- A very popular modern approach is to equate the impulse response of the sampled-data circuit with the impulse response of the analog filter we wish to implement in the converter