EE P 538
Analog Circuits for Sensor Systems

Spring 2020

Instructor: Jason Silver, PhD

ELECTRICAL ENGINEERING
UNIVERSITY *of* WASHINGTON

Announcements

- Design Project Phase 1 due Saturday, June 30 at midnight

Week 9

- AoE Chapter 13 – Analog Meets Digital
- Analog Devices: Mixed Signal Electronic Systems
  - MT-002: Nyquist Criterion
  - MT-001: Quantization Noise
  - MT-009: Data Converter Codes

Overview

- Last time…
  - Higher order filters
  - Highpass filters, bandpass filters
  - Multiple-feedback (MFB) architecture
  - Biquad filter
- Today…
  - Representation of signals
  - Sampling/aliasing
  - Quantization

Lecture 9 – Sampling and Quantization

Analog vs Digital

Analog Systems

- Continuous (time and amplitude) quantities, typically voltages between the supply rails
- One wire/node represents many “bits” of information at a given time
- Susceptible to offset, mismatch
- Limit of resolution is physical noise (e.g. thermal, flicker), which accumulates with the number of stages

Digital Systems

- Discrete (time and amplitude) quantities, equal to either the upper (1) or lower (0) supply rail
- One wire/node represents only a single bit (0 or 1) of information at a given time
- Computation is exact (no offset), and no sensitivity to mismatch
- Limited by number of bits used to represent a signal, no accumulation of noise

Fourier Series

- The Fourier Series of a periodic time-domain signal is given by

$$g(t) = \sum_{n=-\infty}^{\infty} c_n \cdot e^{j2\pi nft}$$

which represents an infinite sum sinusoids with frequencies nf and amplitudes c_n

- The coefficients c_n are given by

$$c_n = \frac{1}{T}\int_{t}^{t+T} g(t) \cdot e^{-j2\pi nft}$$

- The Fourier series is handy for dealing with periodic signals such as sine waves and square waves, but it can't be used to represent the majority of real-world signals, which are in general aperiodic (i.e. not repeating with time)

Fourier Transform

- To represent aperiodic signals, we can use the Fourier Transform, which for a time-domain function $f(t)$ is given as

$$F(j\omega) = \int_{-\infty}^{\infty} f(t) \cdot e^{-j\omega t} dt$$

  where ω is frequency in rads/sec and j is the imaginary number ($j^2 = -1$)
- Whereas the Laplace Transform allows us to transition between the time and frequency domains for *systems*, the Fourier Transform is our link between the two domains for *signals*
- The *inverse* Fourier Transform allows us to perform the reverse operation:

$$f(t) = \int_{-\infty}^{\infty} F(j\omega) \cdot e^{j\omega t} df$$

Unit Impulse

- The time-domain discrete Dirac delta function, also known as the unit impulse, is defined as

$$\delta(t - nt_0) = 1 \text{ when } t = nt_0 \text{ and } 0 \text{ otherwise}$$

- Similarly, in the frequency domain, we have

$$\delta(\omega - n\omega_0) = 1 \text{ when } \omega = n\omega_0 \text{ and } 0 \text{ otherwise}$$

- The Fourier Transform of a complex exponential is an impulse in the frequency domain

$$F(j\omega) = \int_{-\infty}^{\infty} e^{j\omega_0 t} e^{-j\omega t} dt = \delta(\omega - \omega_0)$$

Fourier Transform of a Sine Wave

- By applying Euler's formula, we can express a sine wave as

$$V_p \cos(\omega_1 t) = V_p \cdot \frac{e^{j\omega_1 t} + e^{-j\omega_1 t}}{2}$$

- The Fourier Transform of the sine wave is thus

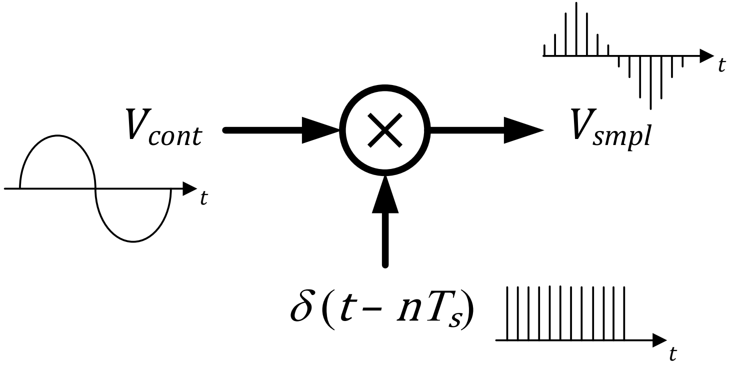
$$F(j\omega) = V_p \int_{-\infty}^{\infty} \frac{e^{j\omega_1 t} + e^{-j\omega_1 t}}{2} e^{-j\omega t} dt = \frac{V_p}{2}\left[\int_{-\infty}^{\infty} e^{j\omega_1 t} e^{-j\omega t} dt + \int_{-\infty}^{\infty} e^{-j\omega_1 t} e^{-j\omega t}\, dt\right]$$

- Which, given the Fourier transform of a complex exponential, is

$$F(j\omega) = \frac{V_p}{2} \cdot [\delta(\omega + \omega_1) + \delta(\omega - \omega_1)]$$

- Which is just two unit impulses at $2\pi f_1$ and $-2\pi f_1$

Impulse Sampling



$$V_{cont}(t) = V_p \cos(2\pi f_{in} t)$$

$$\delta(t - nT_s) = 1 \text{ when } t = nT_s \text{ and } 0 \text{ otherwise}$$

$$V_{smpl}(t) = V_{cont}(t) \cdot \delta(t - nT_s)$$

- The output of an ideal ‘impulse’ sampler consists only of the values of the input occurring at integer multiples of the sampling clock period, T_s
- Sampling is the operation of *quantizing* a signal in time, i.e. constraining the output to only have values at *specific* time points
- Typically, these time points are spaced evenly in time

Sampling Operation

- Assume we sample a sinusoidal signal with an ideal sampler

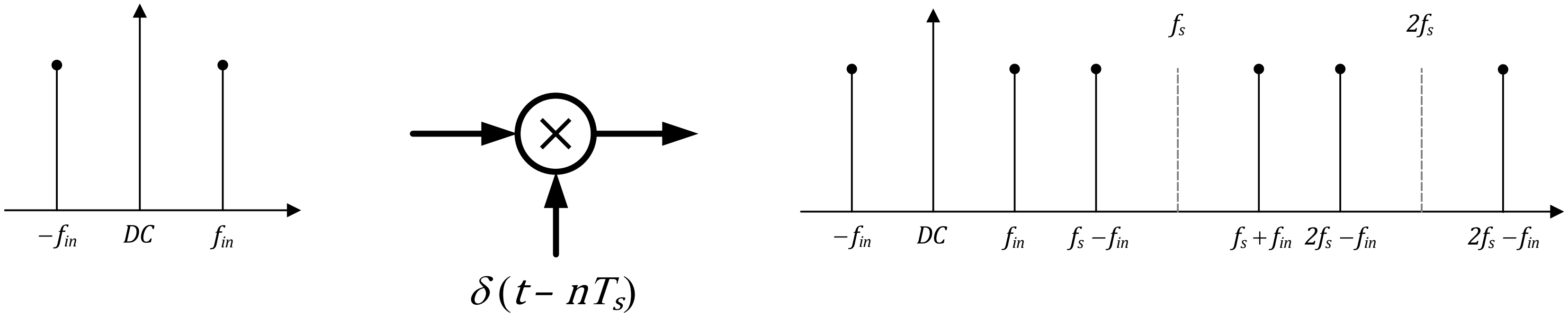
$$y(t) = \sum_{n=-\infty}^{\infty} V_p \cos(2\pi f_{in} t) \cdot \delta(t - nT_s)$$

  where f_{in} is the frequency of the input signal and f_s = $1/T_s$ is the sampling frequency
- The Fourier series of the resulting sequence, $y(t)$, is

$$Y(f) = \frac{V_p}{2T_s} \cdot \sum_{k=-\infty}^{\infty} [\delta(f - f_{in} - kf_s) + \delta(f + f_{in} - kf_s)]$$

  which is a series of impulses in the frequency domain occurring at multiples of the sampling frequency f_s

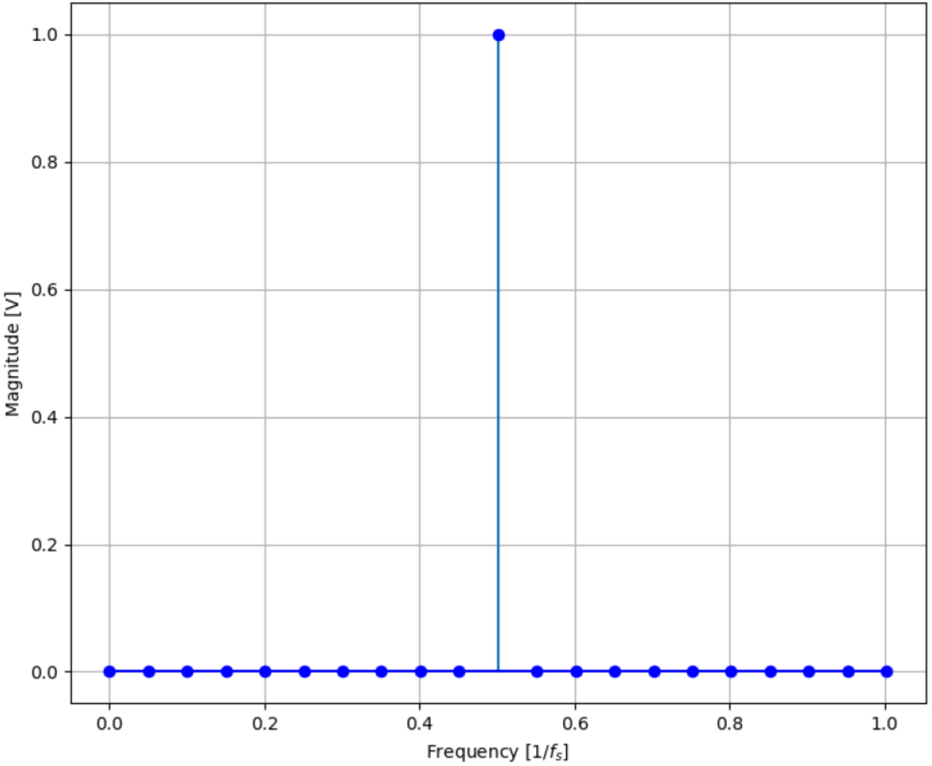
Spectrum of Sampled Signal



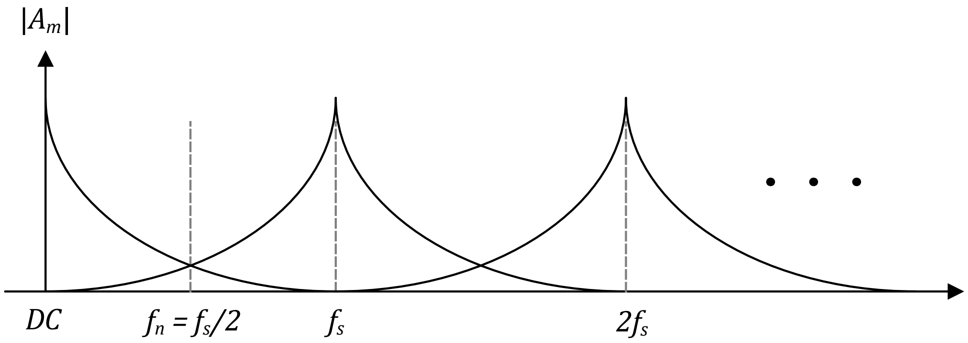
- The sampling process produces "images" of the input spectrum at multiples of the sampling clock frequency
- In order to avoid overlap between these images, the sampling frequency should be at least 2 times higher than the maximum frequency in the input (i.e. the Nyquist frequency)
- This requirement is often referred to as the "Nyquist criterion"

Resulting Spectrum

- As long as the Nyquist criterion isn't violated, we end up with a faithful representation of the sampled spectrum, up to the Nyquist frequency $f_n = f_s/2$
- It is important to remember that a digital signal can only *uniquely* represent signals with frequencies up to $f_s/2$
- Beyond $f_s/2$, we have only reflections/images of the Nyquist band



Aliasing



Nyquist Criterion

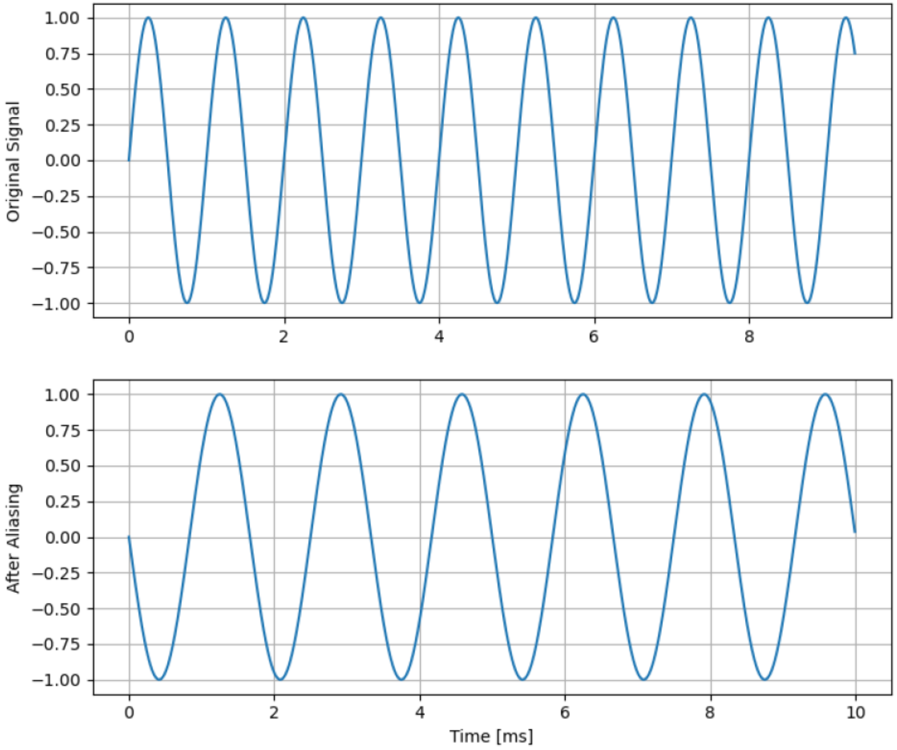
$$f_{in} \leq f_n$$

$$f_n = f_s/2$$

- Sampling causes images/reflections of the Nyquist band to occur at multiples of the sampling frequency
- If the Nyquist criterion is violated, these “aliases” of the sampled spectrum fall into the Nyquist band, corrupting the sampled signal
- The resulting in-band aliases produce distortion in the sampled waveform

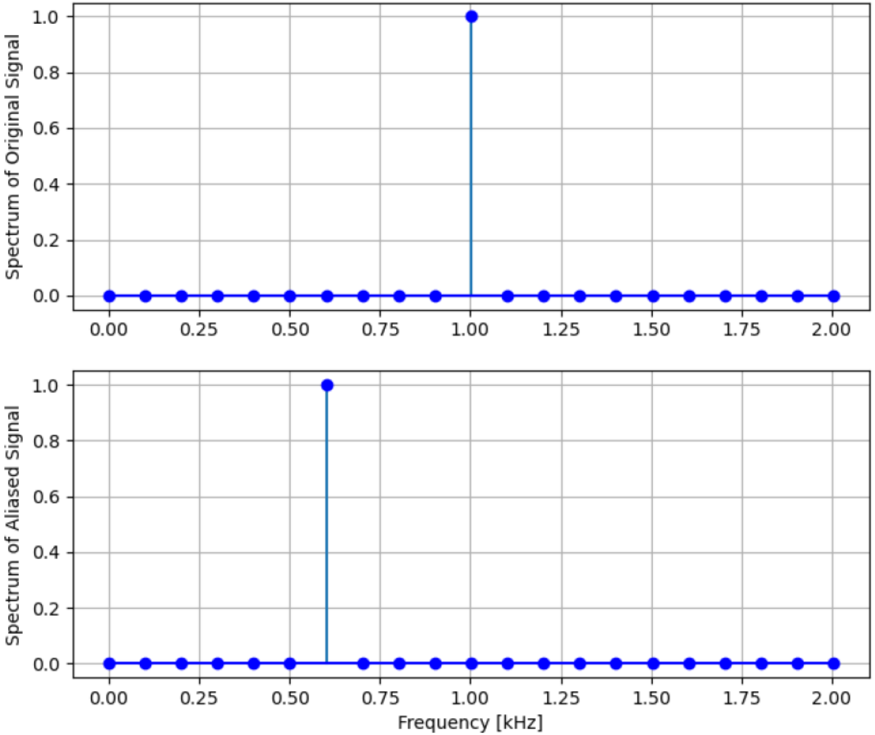
Aliasing (time domain)

- The "analog" input is a sinusoidal signal with $f_{in} = 1kHz$
- The input waveform is sampled at $f_s = 1.6kHz$, violating the Nyquist criterion
- The result is a sinusoid with frequency $f_s - f_{in} = 600Hz$, where f_s is the sampling frequency
- Due to "undersampling," information in the original signal is lost (forever!)



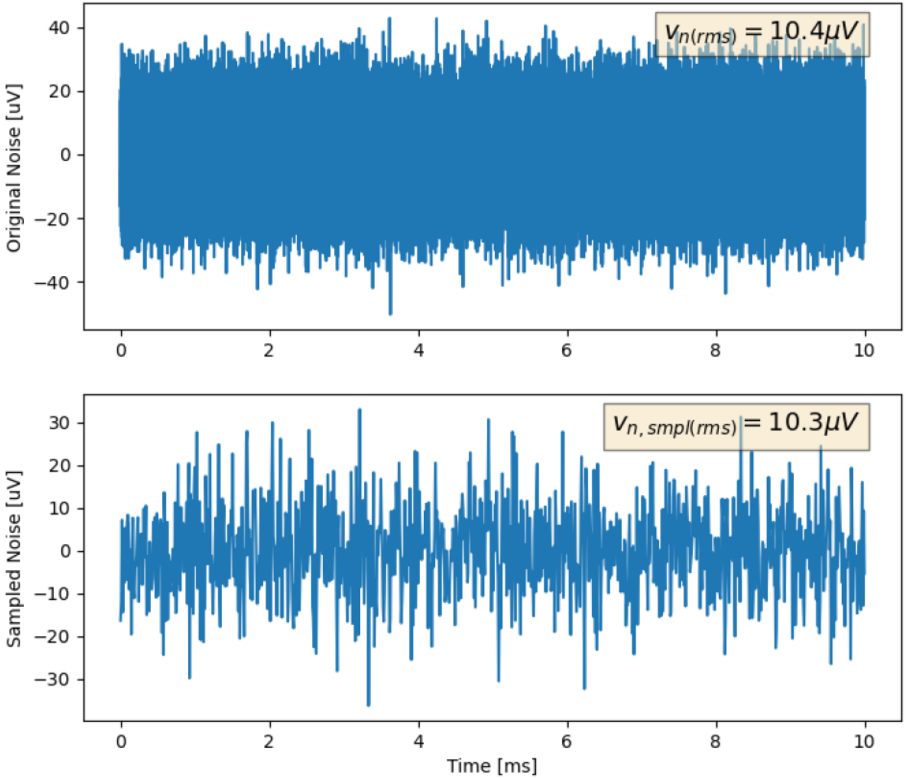
Aliasing (frequency domain)

- To quantify the effects of aliasing, it is typically more informative to use the frequency domain
- Here we see the spectrum of the same 1-kHz sinusoid, both before and after aliasing
- Because the sampled data only contains information up to the Nyquist frequency, the original 1kHz tone is gone and we are left with only the aliased version



Sampled White Noise

- When we sample broadband noise, it gets aliased just like any other signal
- We can see this in the time domain in that we have fewer points (more “sparse”), but the rms value remains the same
- Thus, the same rms noise is present over a smaller bandwidth, and the equivalent noise bandwidth of the sampled noise is the Nyquist bandwidth, $f_s/2$



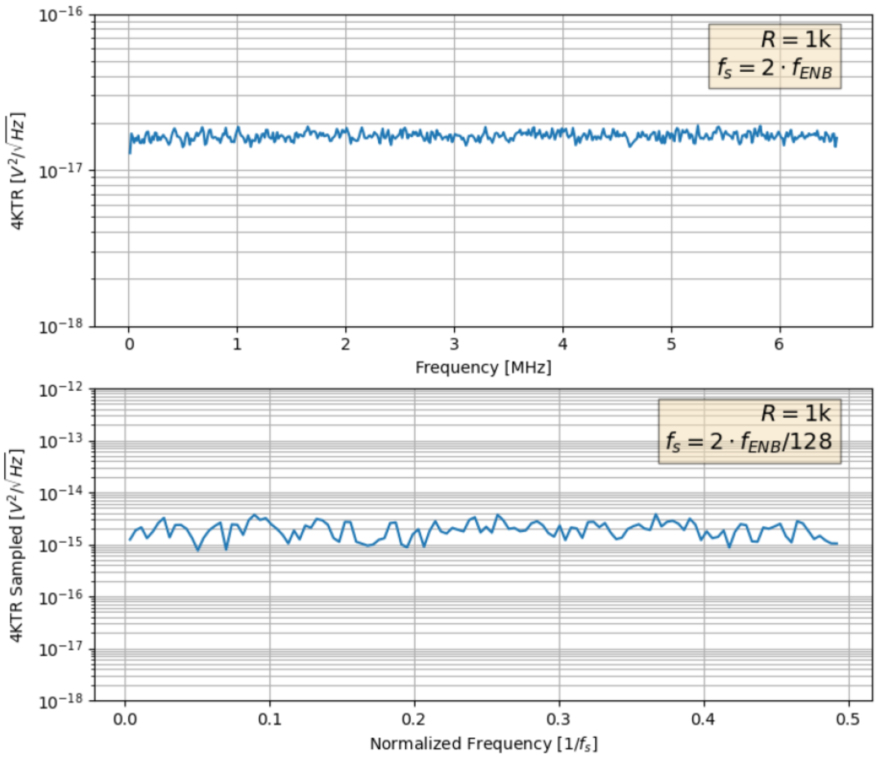
Spectrum of Sampled Noise

- In the frequency domain, we can see the effect of sampling on the power spectral density of the sampled noise
- The analog noise has a power spectral density given by

$$e_{nR}^2 = 4kTR$$

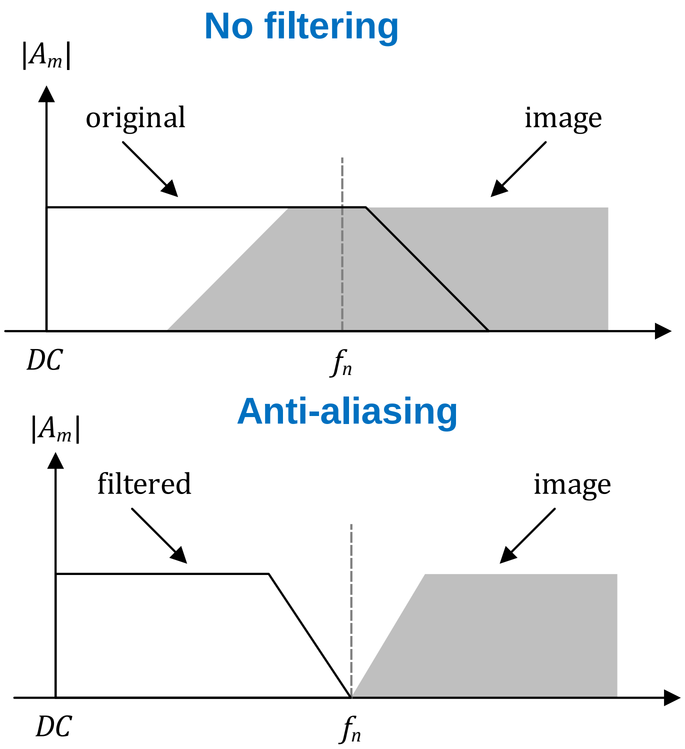
- Due to aliasing, the broadband components of the noise fall into the Nyquist band:

$$e_{nR,smpl}^2 = 4kTR\frac{f_{ENB}}{f_s/2}$$



Anti-aliasing

- To prevent information loss and other undesirable effects of aliasing, it is necessary to “band-limit” the signal to be sampled to minimize content above f_n
- In reality, no real-world signal is completely devoid of higher frequency content, and some amount of aliasing is inevitable
- As analog designers, we should ensure that *no signal* and *minimal noise* exists above the Nyquist frequency

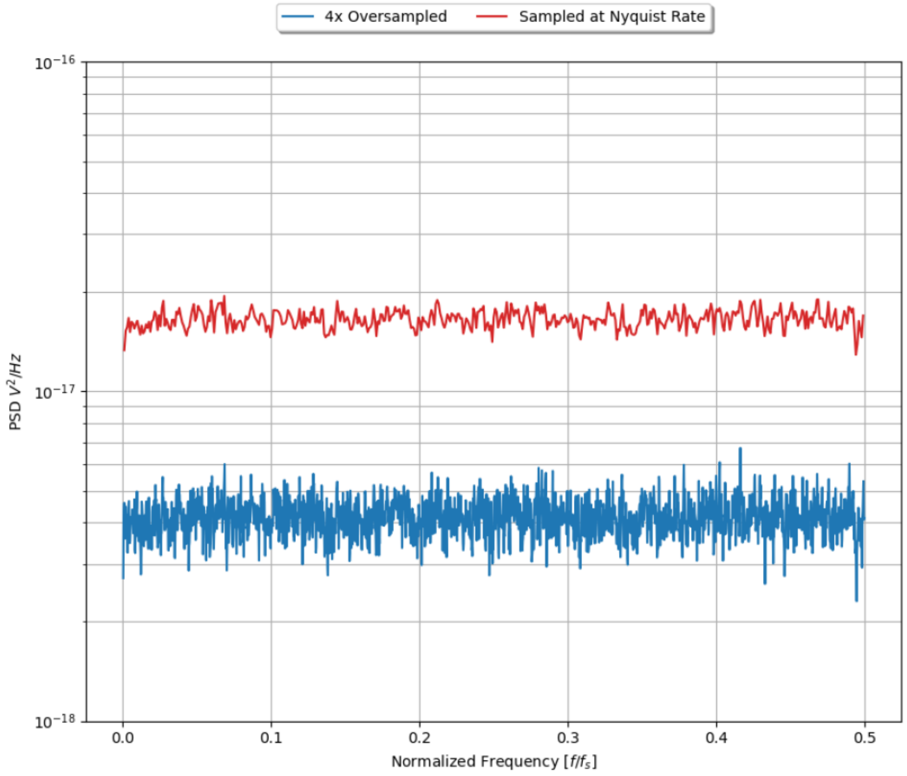


Oversampling

- If we increase the sampling clock frequency above the Nyquist rate, the white noise gets distributed over a wider frequency range, decreasing the noise PSD
- As a result, the *sampled noise density* becomes

$$e^2_{n,smpl} = e^2_{n,orig} \frac{f_{ENB}}{f_s/2}$$

- Where the *oversampling rate* (OSR) is the ratio of f_s to $2f_n$



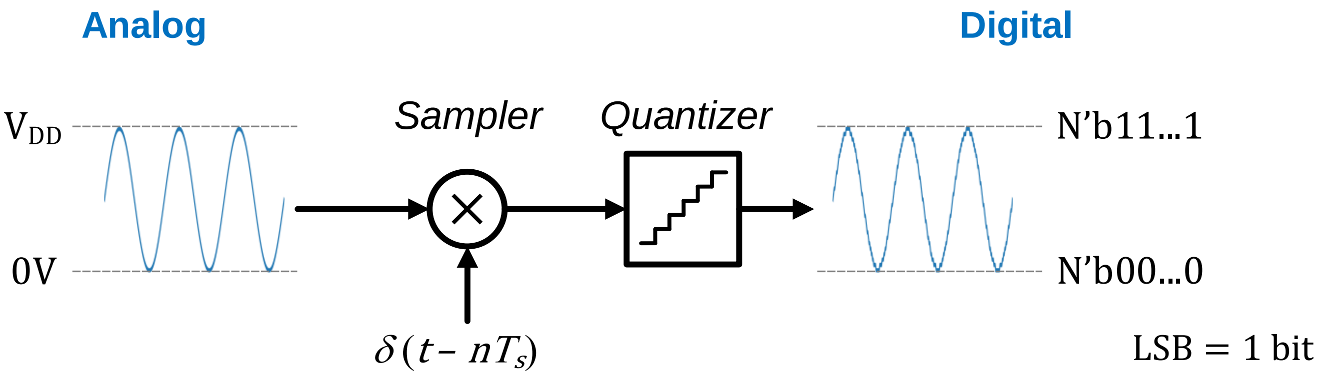
Improvement to SNR

- We can now apply filtering in the digital domain, where it is much easier (and cheaper) to realize high-order filters with steep transitions
- Let's assume we apply a filter cutoff of f_c to the oversampled noise, setting the *new* equivalent noise bandwidth to the Nyquist frequency
- The rms noise then becomes:

$$v_{n(rms)} = \sqrt{e_{n,analog}^2 \frac{f_{ENB,orig}}{f_s/2} f_c} = \sqrt{\frac{e_n^2 f_c}{OSR}}$$

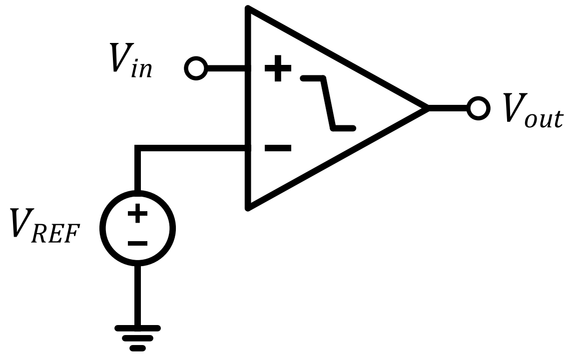
- Thus, the rms noise is reduced by the square root of the oversampling rate!
- Note that we *must apply filtering to the sampled noise* in order to reap the benefit of oversampling

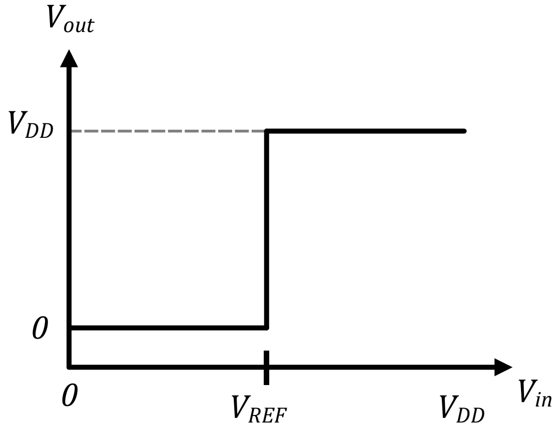
Quantization



- The conversion from an analog signal to a digital signal involves quantization in both *time* and *amplitude*
- Quantization in time is the result of the sampling process, while quantization in amplitude is the result of the finite numerical precision of mixed-signal/digital systems

1-bit Quantizer

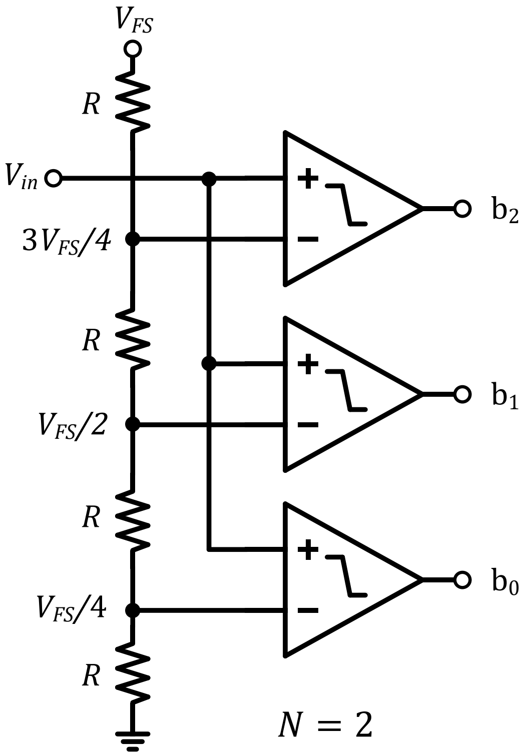




- A 1-bit quantizer is just a comparator, which is essentially a high-gain amplifier operated in open loop (no feedback)
- A small difference in the input voltage causes the output to swing high or low, depending on the polarity of the difference
- The resulting output is a binary (i.e. digital) signal, with V_{DD} representing a 1 and ground representing a 0

Multi-bit Quantizer

- Multi-bit comparisons can be performed in parallel, as in a “flash” ADC
- For example, if $V_{FS}/4 < V_{in} < V_{FS}/2$, bit b_0 is high (1), and bits b_{1-3} are low (0), resulting in an output code 3’b001
- For input values above $3V_{FS}/4$ or below $V_{FS}/4$, the converter is “saturated,” in that it cannot uniquely represent signals beyond these values



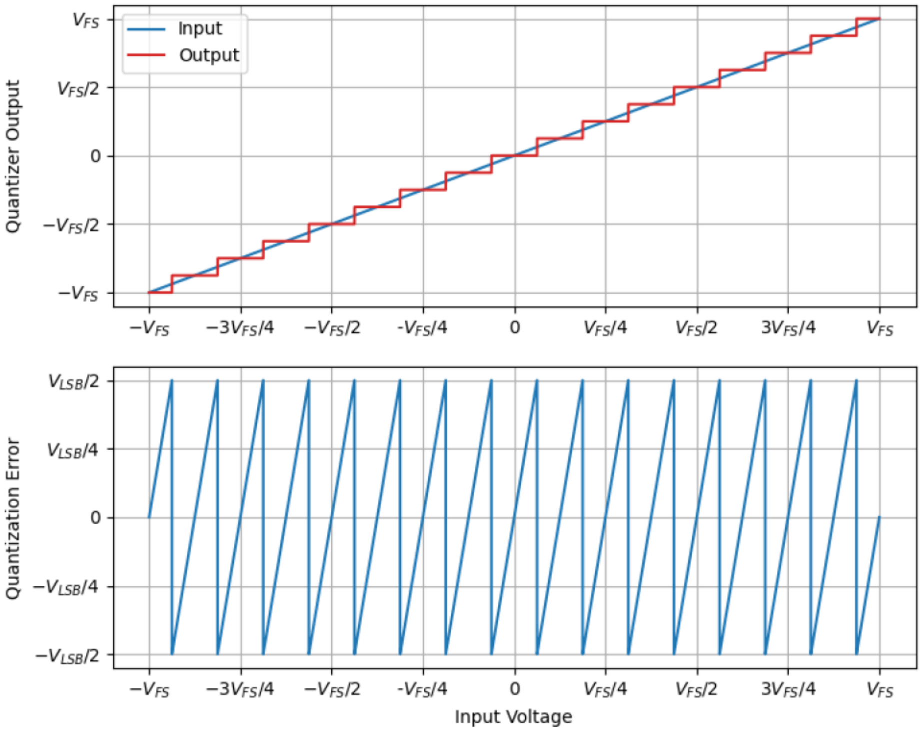
Analog	Therm.	Binary
$3V_{FS}/4$	111	11
$V_{FS}/2$	011	10
$V_{FS}/4$	001	01
0	000	00

$$V_{LSB} = \frac{V_{FS}}{2^N} = \frac{V_{FS}}{4}$$

Quantization Error

- As the input to the quantizer is swept over its range, the quantization error is the difference between the actual value and the quantized value
- The quantization error is limited by the quantization step size, V_{LSB}
- The rms value of the quantization error is given by

$$v_{nq,rms} = \sqrt{\frac{1}{T}\int_0^T \left(\frac{V_{LSB}}{2} - \frac{V_{LSB}}{T}\cdot t\right)^2 dt} = \frac{V_{LSB}}{\sqrt{12}}$$

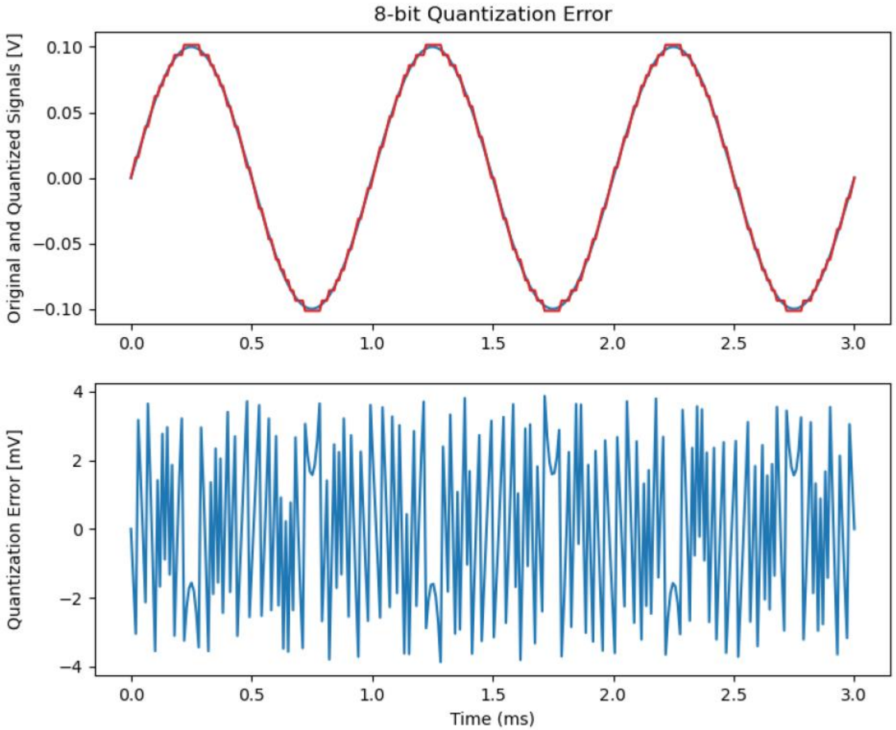


Quantization Noise

- Viewed as a function of time, the quantization error looks (almost) like a random signal
- The amplitude of the quantization noise signal at any given point in time is in the range

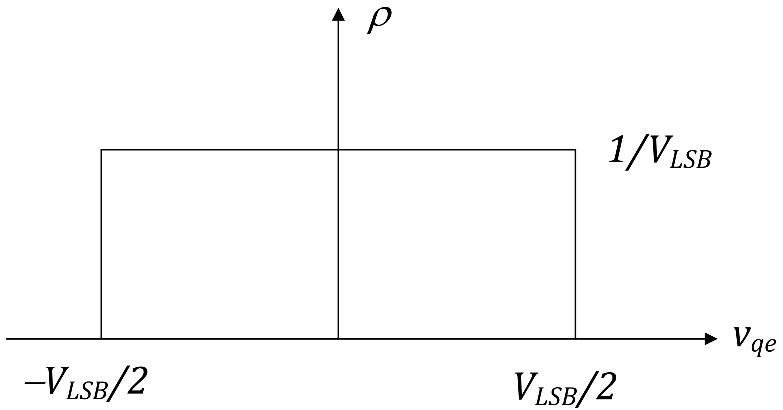
$$-\frac{V_{LSB}}{2} < v_{nq} < \frac{V_{LSB}}{2}$$

- Assuming the quantization noise is "random enough," the probability is the same for any value in this range



Quantization Noise Amplitude

- Unlike resistor thermal noise, which has a normal (Gaussian) probability density function, the probability density for quantization noise is *uniform* in the range $-V_{LSB}/2$ to $V_{LSB}/2$
- That is, at any given point in time the quantization error is equally likely to be any value in this range
- For a sufficiently "busy" input signal, the quantitation error appears as random noise in time



ρ: probability density

v_{qe}: quantization error

V_{LSB}: quantization step (least significant bit)

Quantization Noise Density

- This random noise signal exists over a bandwidth limited by the sample frequency (i.e. the sample frequency sets the noise bandwidth)
- The rms value of the noise signal is given by

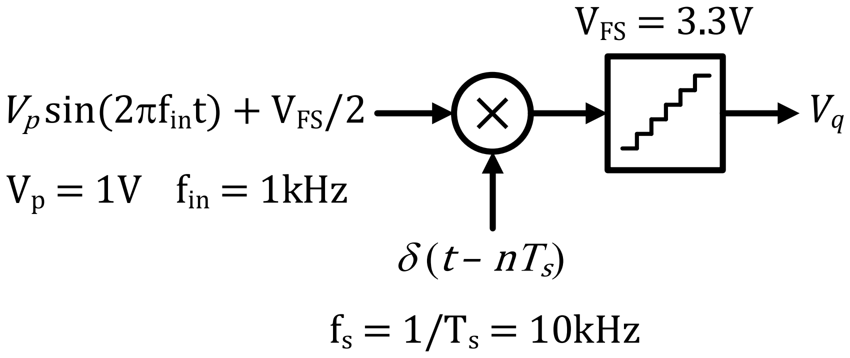
$$v_{nq(rms)} = \frac{V_{LSB}}{\sqrt{12}}$$

  and the noise bandwidth is just the Nyquist frequency $f_s/2$

- The *mean square quantization noise* is equal to the square of the rms noise divided by the Nyquist bandwidth of $f_s/2$

$$e_{nq}^2 = \frac{V_{LSB}^2}{12 \cdot (f_s/2)} = \frac{V_{LSB}^2}{6f_s}$$

Example: 16-bit Quantization



$$V_{LSB} = \frac{V_{FS}}{2^{16}} = \frac{3.3V}{2^{16}} = 50.3\mu V$$

$$e_{nq}^2 = \frac{V_{LSB}^2}{6f_s} = \frac{(0.21\mu V)^2}{Hz}$$

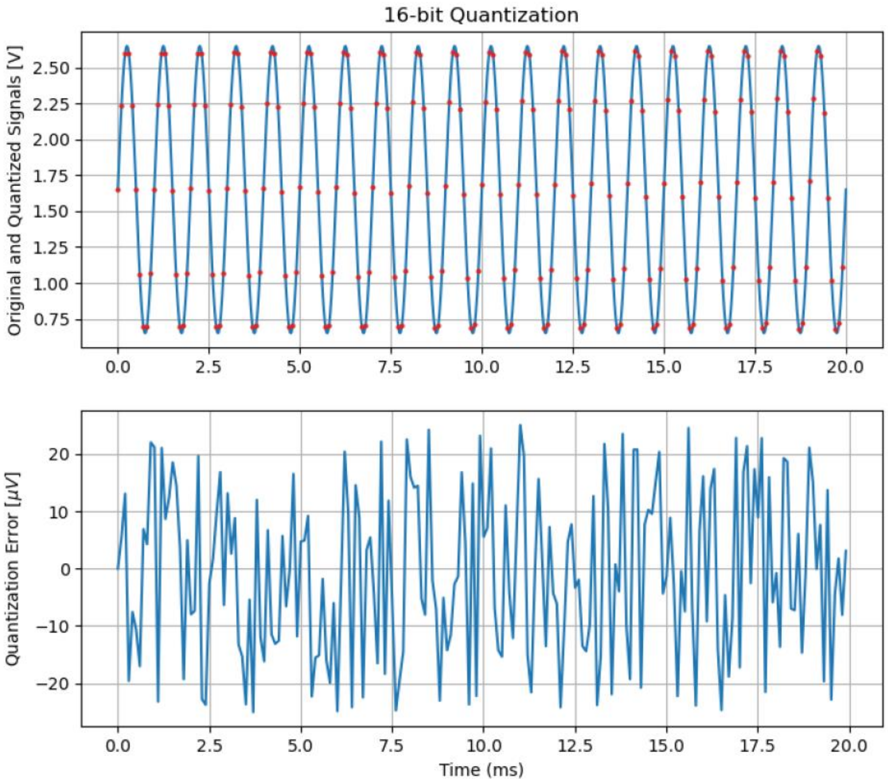
$$v_{nq(rms)} = \frac{V_{LSB}}{\sqrt{12}} = 14.5\mu V$$

- Suppose we have a 16-bit quantizer with a full-scale range of 3.3V and an impulse sampler sampling at 10ksps:

1. The input signal needs to fall within the range of the quantizer, and
2. The sampling frequency should satisfy the Nyquist criterion

Example: 16-bit Quantization

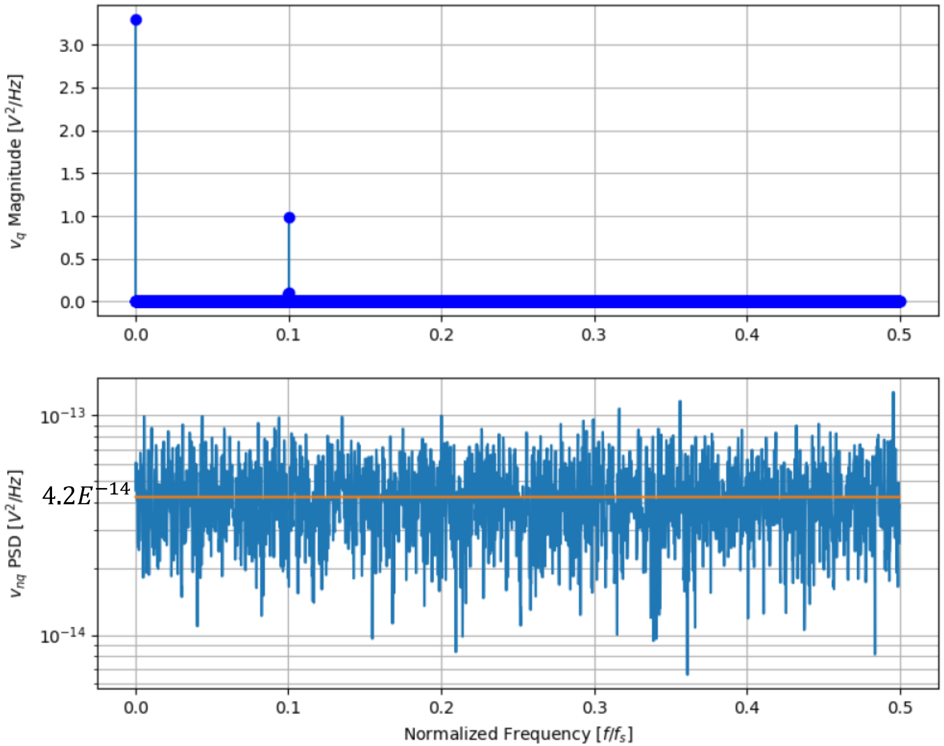
- The quantization error for each sample is the difference between the original signal and the sampled one
- Again, the quantization error appears as a random noise in time, as long as the input signal changes sufficiently fast relative to the sample rate
- Real signals are generally noisy enough to increase the randomness of the quantization noise



Quantized Signal Spectrum

- Because we've satisfied the Nyquist criterion, the input signal makes it through the sampling process intact
- The spectral density of the quantization noise is (almost) flat (i.e. constant with frequency)
- The rms quantization noise is simply

$$v_{nq(rms)} \approx \sqrt{4.2E^{-14}\frac{V^2}{Hz} \cdot 5kHz} = 14.1\mu V$$



Signal-to-Quantization-Noise Ratio

- If we change the amplitude of the signal being quantized, the ratio between the rms signal and rms quantization noise increases, improving the signal-to-quantization-noise ratio (SQNR)
- The SQNR is just the ratio of the rms signal to the rms quantization noise:

$$SQNR = 20\log_{10}\frac{v_{sig(rms)}}{V_{LSB}/\sqrt{12}}$$

- Similarly, if the quantization step size is reduced, the SQNR is increased
- Typical designs are not limited by SQNR, which is generally high
- In the case of the previous 16-bit design example,

$$SQNR = 20\log_{10}\frac{0.707V}{14.2\mu V} = 94dB$$

SQNR with Oversampling

- As with uniform white noise, the power spectral density of quantization noise can be decreased by oversampling
- The quantization noise density is given by

$$e_{nq}^2 = \frac{V_{LSB}^2}{12 f_s/2}$$

  which, like sampled white noise, depends on the sample frequency f_s
- By increasing the sampling frequency beyond the Nyquist rate and then filtering up to the Nyquist bandwidth, we decrease the RMS quantization noise:

$$v_{nq(rms)} = \frac{V_{LSB}}{\sqrt{12}} \cdot \sqrt{\frac{2f_n}{f_s}} = \frac{V_{LSB}}{\sqrt{12}} \cdot \sqrt{OSR}$$