ESE 3400: Medical Devices Lab

Lec 13: October 31, 2022

Data Converters Pt 3: ADC Oversampling and Noise Shaping

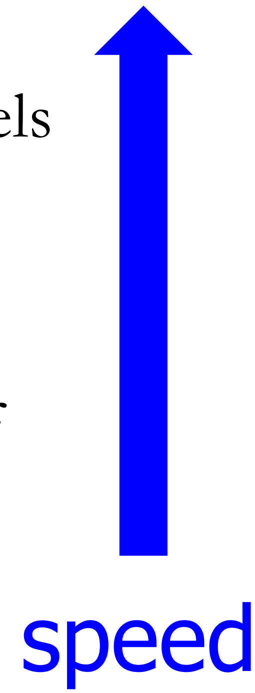
Lecture Outline

- Oversampling
- Noise Shaping
- Sigma-Delta ADCs

ADC Architectures

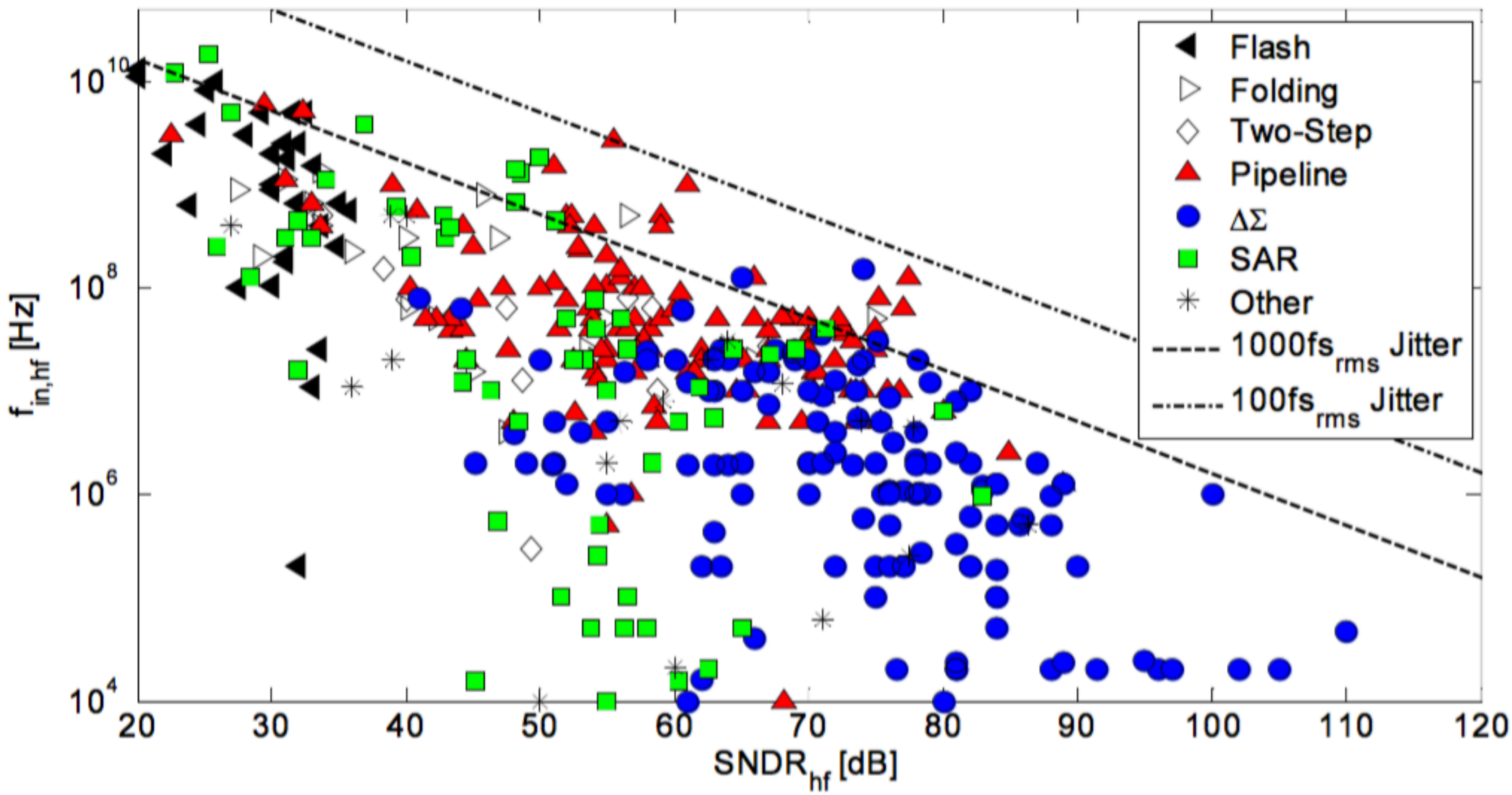
Nyquist ADC Architectures

- Word-at-a-time
  - E.g. flash ADC
  - Instantaneous comparison with 2^B-1 reference levels
- Multi-step
  - E.g. pipeline ADCs
  - Coarse conversion, followed by fine conversion of residuals
- Bit-at-a-time
  - E.g. successive approximation ADCs
  - Conversion via a binary search algorithm



ADC Survey (ISSCC & VLSI 1997-2013)

Data: http://www.stanford.edu/~murmann/adcsurvey.html



Signal-to-Quantization-Noise Ratio

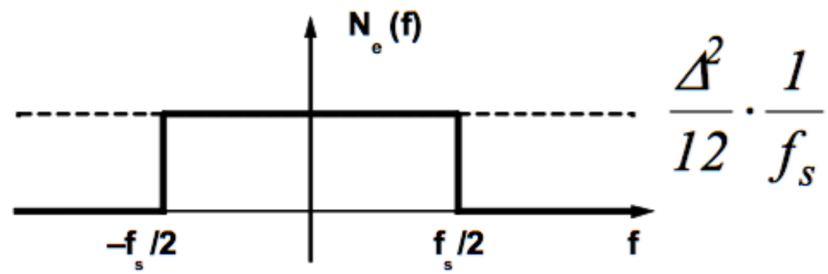
- Assuming full-scale sinusoidal input, we have

$$SNR_Q = 6.02B + 1.76 \text{ dB}$$

B (Number of Bits)	SQNR
8	50dB
12	74dB
16	98dB
20	122dB

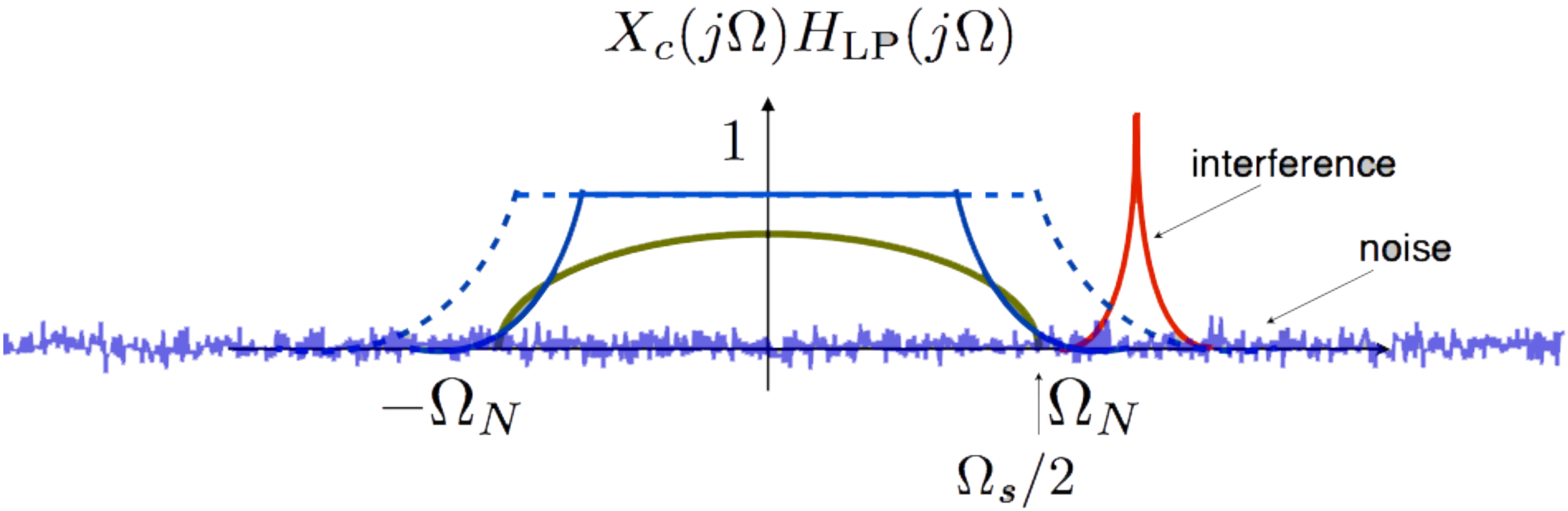
Quantization Noise Spectrum

- If the quantization error is "sufficiently random", it also follows that the noise power is uniformly distributed in frequency



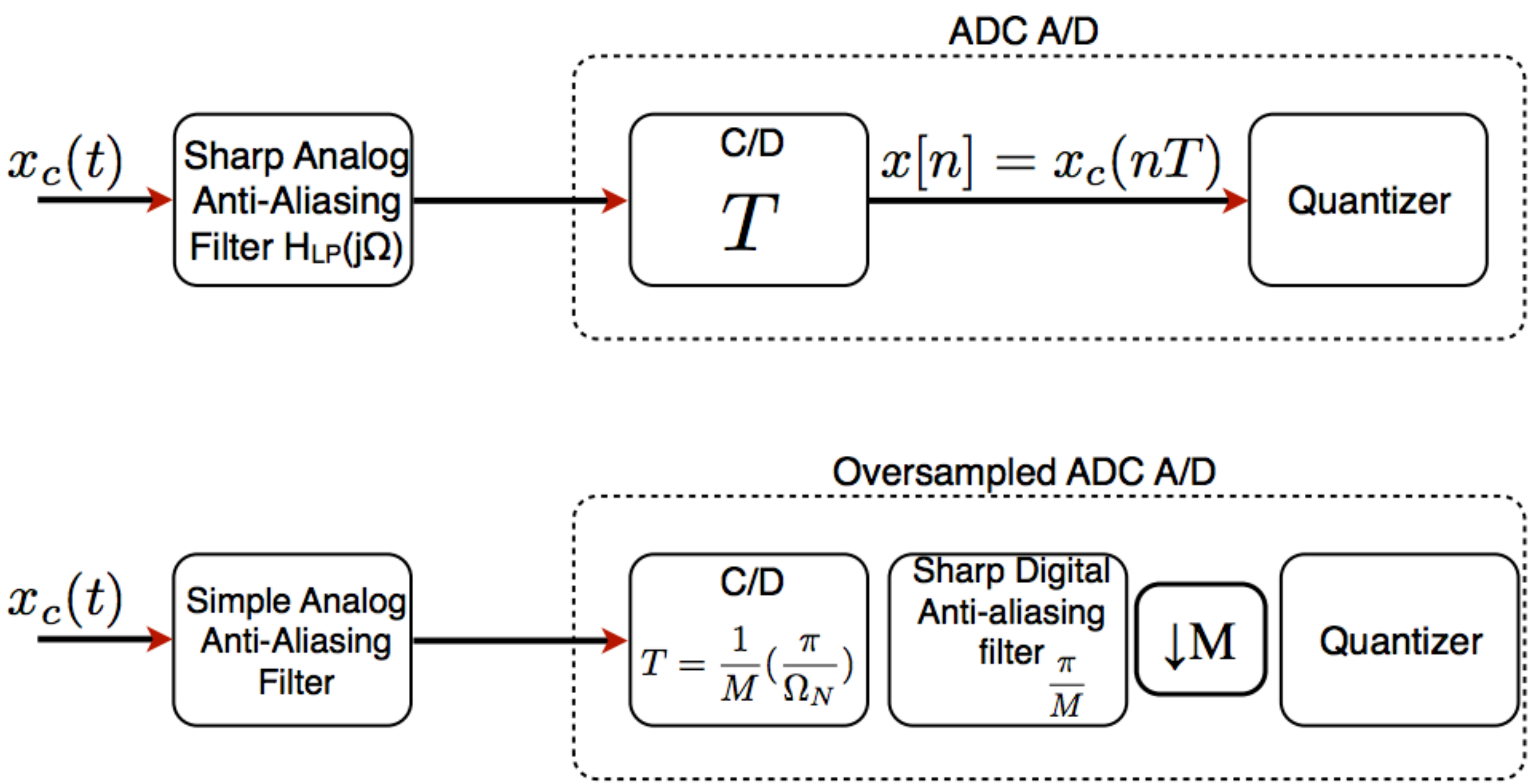
- References
  - W. R. Bennett, "Spectra of quantized signals," Bell Syst. Tech. J., pp. 446-72, July 1988.
  - B. Widrow, "A study of rough amplitude quantization by means of Nyquist sampling theory," IRE Trans. Circuit Theory, vol. CT-3, pp. 266-76, 1956.

Non-Ideal Anti-Aliasing Filter

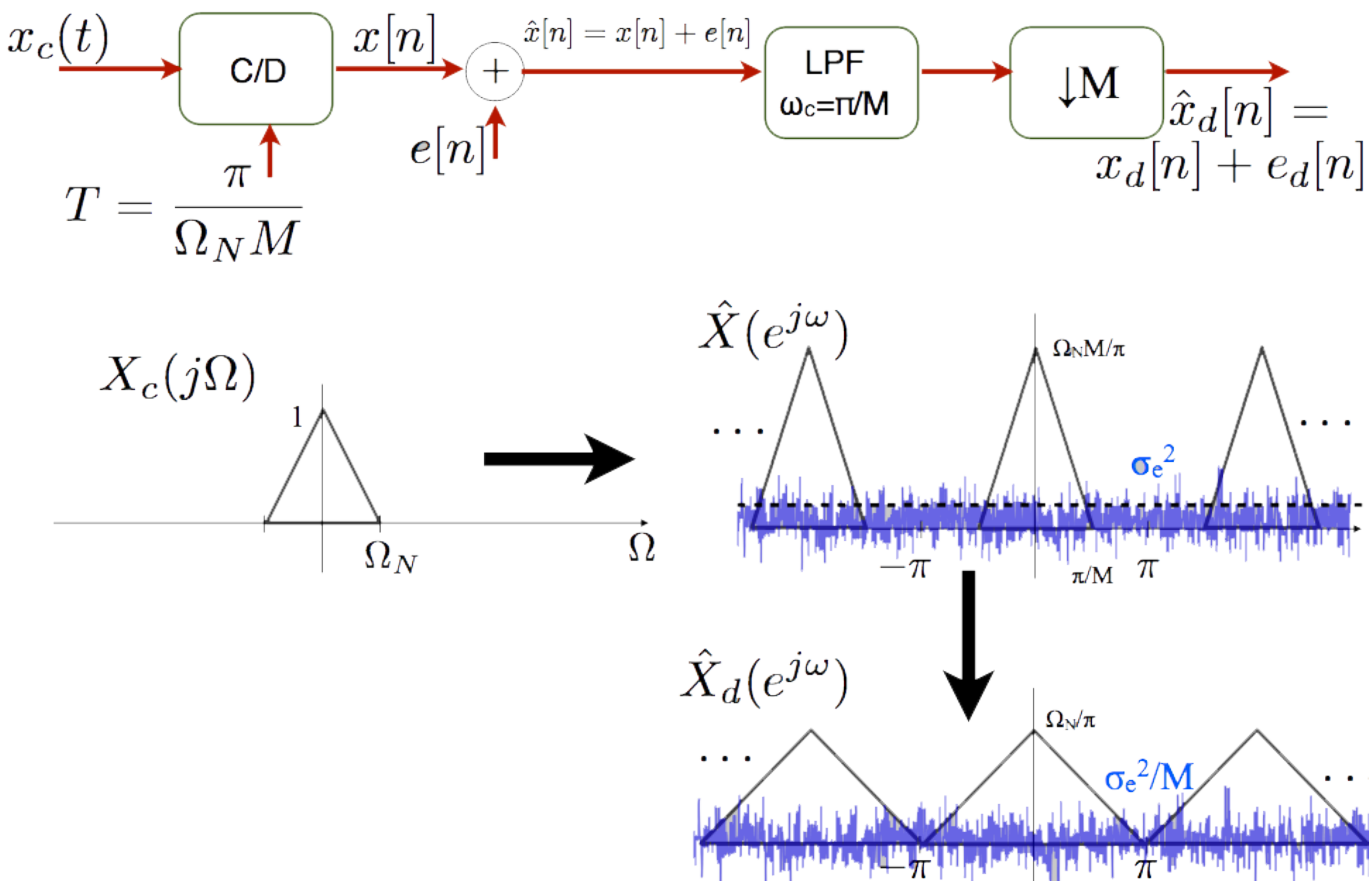


- Problem: Hard to implement sharp analog filter
- Consequence: Crop part of the signal and suffer from noise and interference

Oversampled ADC

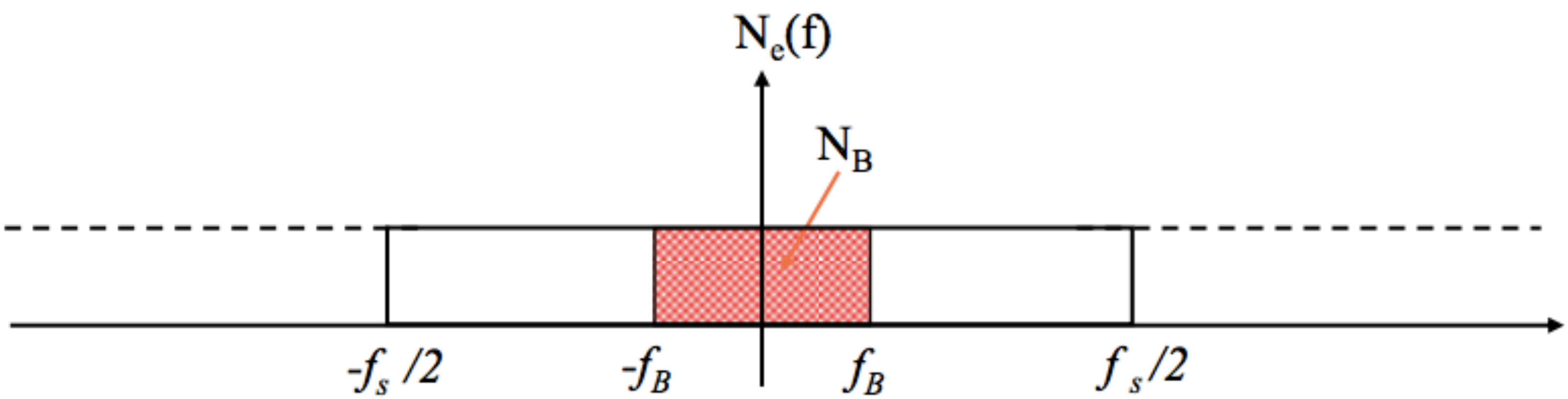


Quantization Noise with Oversampling



ADC Converters Baseband Noise

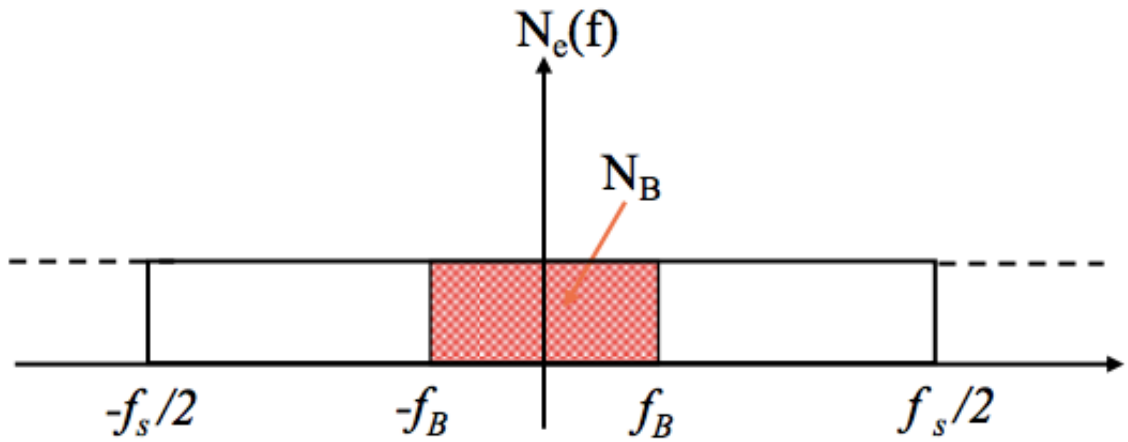
- For a quantizer with quantization step size Δ and sampling rate f_S:
  - Quantization noise power is distributed uniformly across Nyquist bandwidth ($f_S/2$)



  - Power spectral density: $$N_e(f) = \frac{\overline{e^2}}{f_s} = \left(\frac{\Delta^2}{12}\right)\frac{1}{f_s}$$
  - Noise is distributed over the Nyquist band $-f_S/2$ to $f_S/2$

Oversampled Converters Baseband Noise

$$S_B = \int_{-f_B}^{f_B} N_e(f)\,df =$$



Oversampled Converters Baseband Noise

$$S_B = \int_{-f_B}^{f_B} N_e(f)\,df = \int_{-f_B}^{f_B} \left(\frac{\Delta^2}{12}\right)\frac{1}{f_s}\,df$$

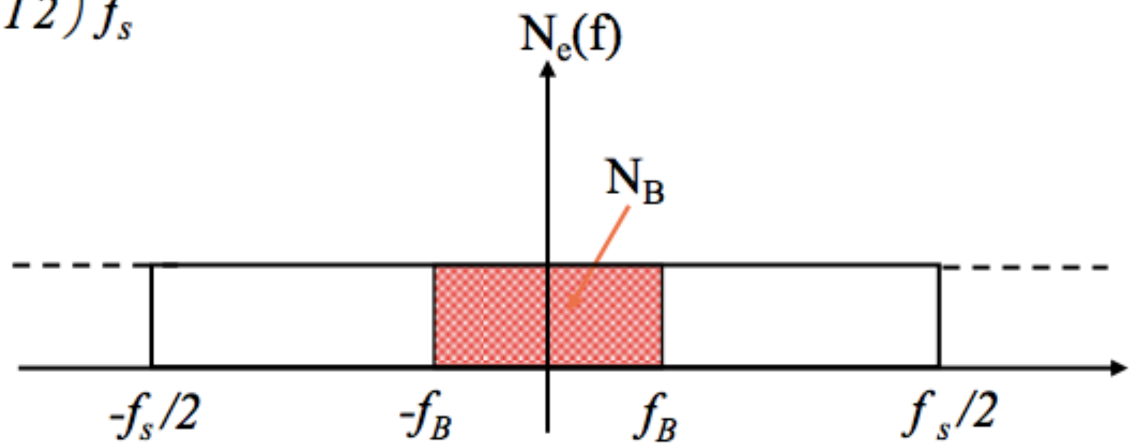
$$= \frac{\Delta^2}{12}\left(\frac{2f_B}{f_s}\right)$$

$$where\ for\ f_B = f_s/2$$

$$S_{B0} = \frac{\Delta^2}{12}$$

$$S_B = S_{B0}\left(\frac{2f_B}{f_s}\right) = \frac{S_{B0}}{M}$$

$$where\ M = \frac{f_s}{2f_B} = oversampling\ ratio$$



Oversampled Converters Baseband Noise

$$S_B = S_{B0}\left(\frac{2f_B}{f_s}\right) = \frac{S_{B0}}{M}$$

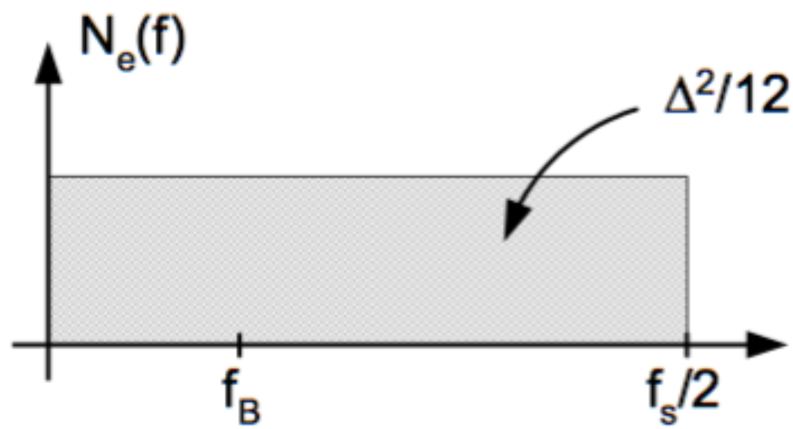
$$where\ M = \frac{f_s}{2f_B} = oversampling\ ratio$$

2X increase in M

→ 3dB reduction in S_B

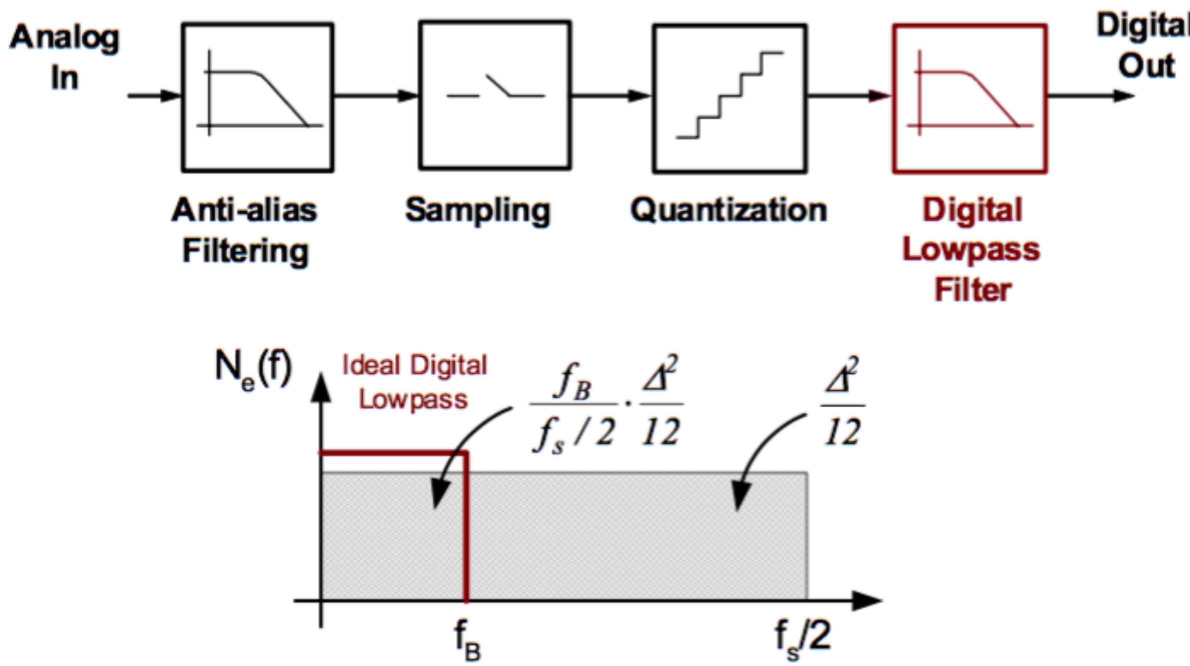
→ ½ bit increase in resolution/octave oversampling

Quantization Noise



- Recall that the "noise" introduced by quantizer is evenly distributed across all frequencies
  - Provided that quantization error sequence is "sufficiently random"
- Idea: Let's filter out the noise beyond $f=f_B$

Digital Noise Filter



- Total quantization noise at digital output is reduced proportional to "oversampling ratio" $M=(f_S/2)/f_B$

Digital Noise Filter

- Increasing M by 2x, means a 1/2 bit increase in resolution
  - "1/2 bit per octave" (octave means doubling in frequency)
- Is this useful?
- Reality check
  - Want 16-bit ADC, f_B=1MHz
  - Use oversampled 8-bit ADC with digital lowpass filter

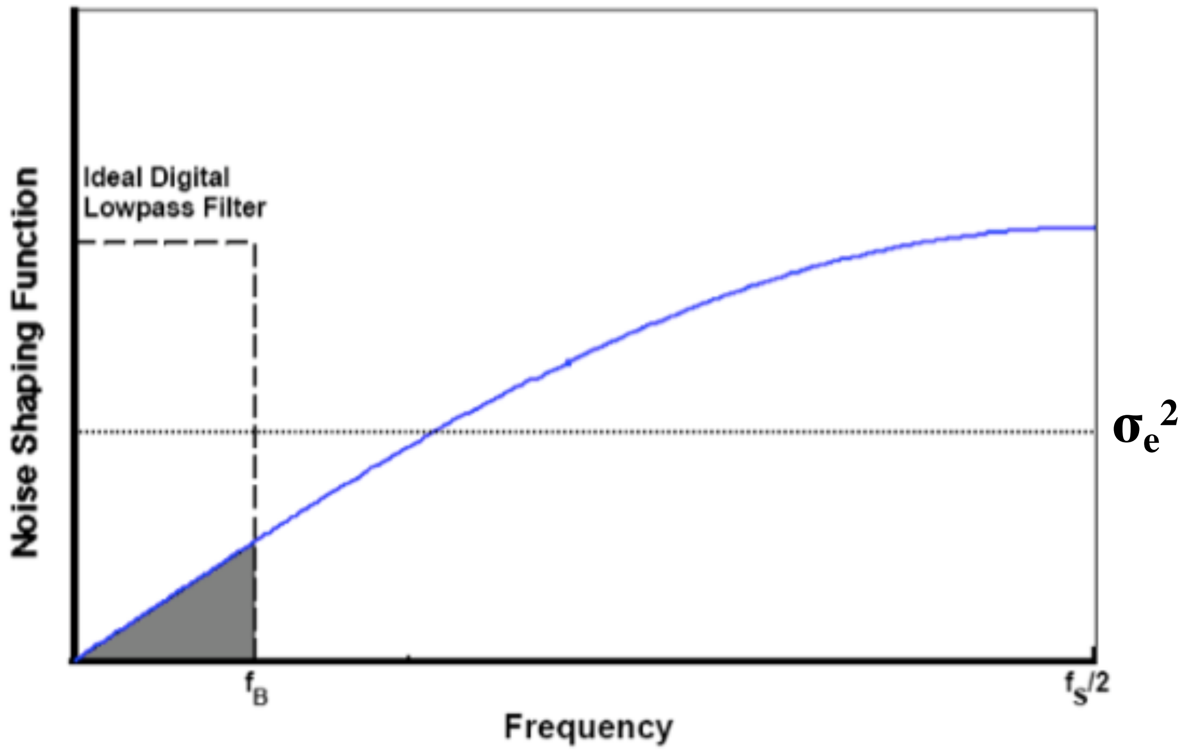
Digital Noise Filter

- Increasing M by 2x, means a 1/2 bit increase in resolution
  - "1/2 bit per octave" (octave means doubling in frequency)
- Is this useful?
- Reality check
  - Want 16-bit ADC, f_B=1MHz
  - Use oversampled 8-bit ADC with digital lowpass filter
  - 8-bit increase in resolution necessitates oversampling by 16 octaves

$$f_s \geq 2 \cdot f_B \cdot M = 2 \cdot 1MHz \cdot 2^{16}$$
$$\geq 131GHz$$

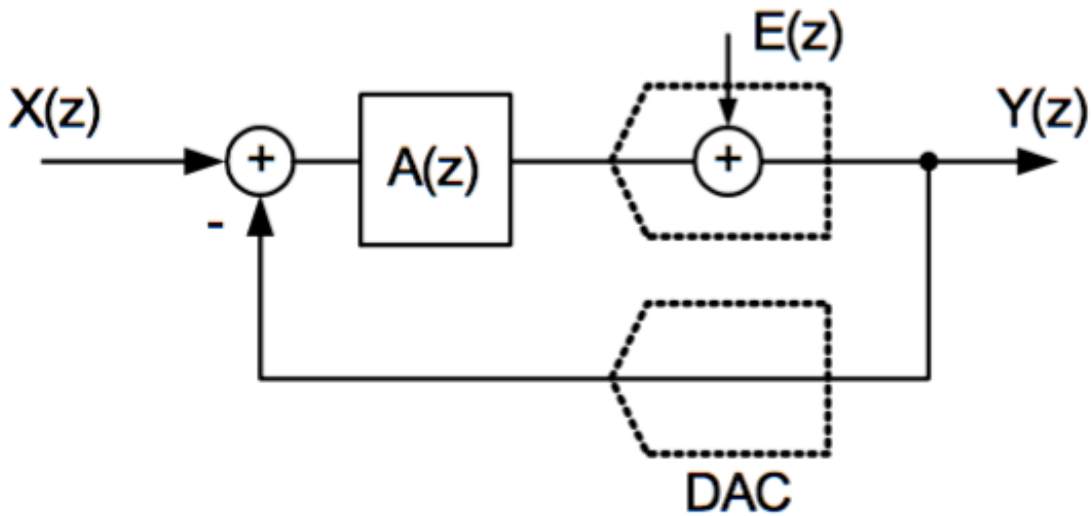
Noise Shaping

Noise Shaping



- Idea: "Somehow" build an ADC that has most of its quantization noise at high frequencies
- Key: Feedback

Noise Shaping Using Feedback



z-Transform

- Define the **forward z-transform** of x[n] as

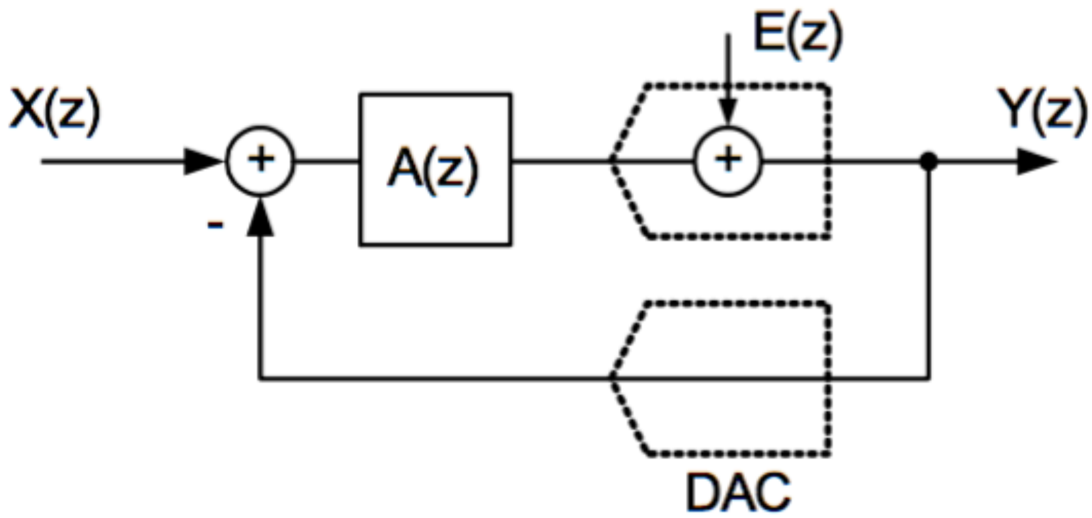
$$X(z) = \sum_{n=-\infty}^{\infty} x[n]\, z^{-n}$$

- The core “basis functions” of the z-transform are the complex exponentials z^n with arbitrary $z \in C$; these are the eigenfunctions of LTI systems for infinite-length signals

- **Notation abuse alert:** We use X(•) to represent both the DTFT $X(e^{j\omega})$ and the z-transform X(z); they are, in fact, intimately related

$$X_z(z)|_{z=e^{j\omega}} = X_z(e^{j\omega})$$

Noise Shaping Using Feedback



$$Y(z) = E(z) + A(z)X(z) - A(z)Y(z)$$

$$= E(z)\frac{1}{1+A(z)} + X(z)\frac{A(z)}{1+A(z)}$$

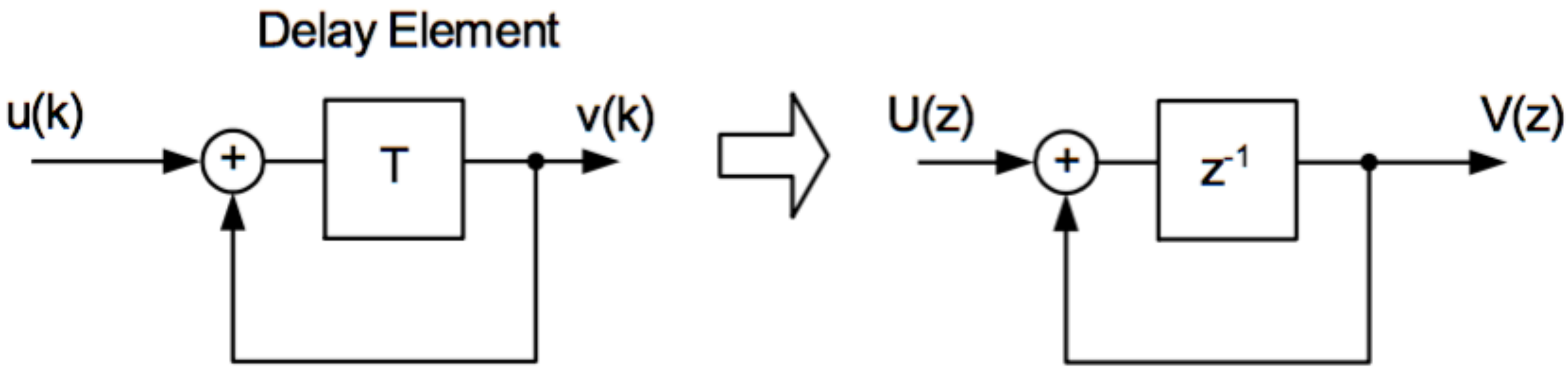
$$= E(z)\underbrace{H_E(z)}_{\substack{\text{Noise}\\\text{Transfer}\\\text{Function}}} + X(z)\underbrace{H_X(z)}_{\substack{\text{Signal}\\\text{Transfer}\\\text{Function}}}$$

Noise Shaping Using Feedback

$$Y(z) = E(z)\underbrace{\frac{1}{1+A(z)}}_{\substack{Noise\\Transfer\\Function}} + X(z)\underbrace{\frac{A(z)}{1+A(z)}}_{\substack{Signal\\Transfer\\Function}}$$

- Objective
  - Want to make STF unity in the signal frequency band
  - Want to make NTF "small" in the signal frequency band
- If the frequency band of interest is around DC ($0 \ldots f_B$) we achieve this by making $|A(z)| >> 1$ at low frequencies
  - Means that NTF << 1
  - Means that STF ≅ 1

Discrete Time Integrator



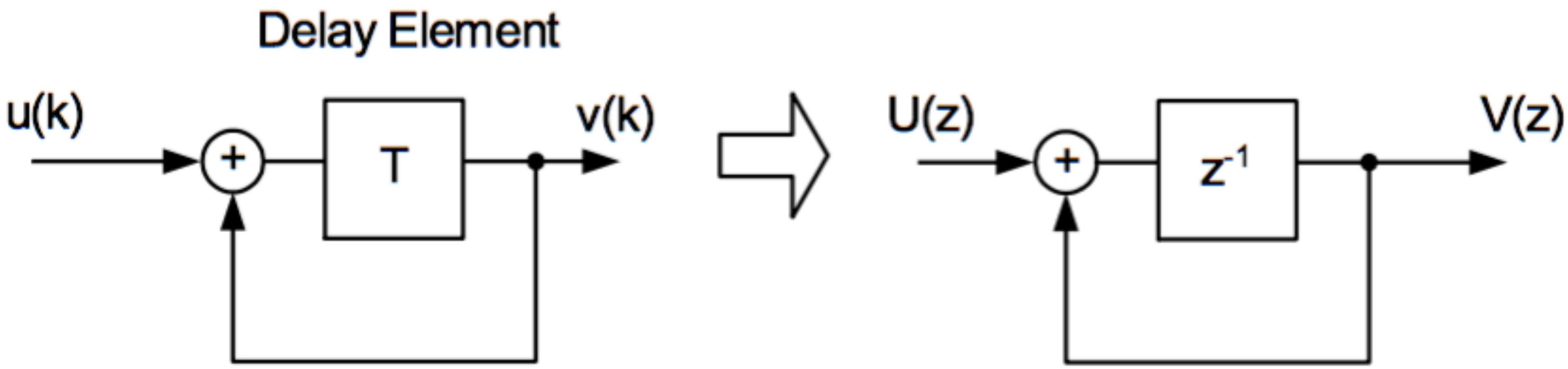
$$v(k) = u(k-1) + v(k-1)$$

$$V(z) = z^{-1}U(z) + z^{-1}V(z)$$

Z-Transform Properties

Property	Time Domain	z-Domain	ROC
Notation:	$x(n)$	$X(z)$	ROC: $r_2 < \|z\| < r_1$
	$x_1(n)$	$X_1(z)$	ROC_1
	$x_2(n)$	$X_2(z)$	ROC_2
Linearity:	$a_1x_1(n) + a_2x_2(n)$	$a_1X_1(z) + a_2X_2(z)$	At least $ROC_1 \cap ROC_2$
Time shifting:	$x(n-k)$	$z^{-k}X(z)$	At least ROC, except $z = 0$ (if $k > 0$) and $z = \infty$ (if $k < 0$)
z-Scaling:	$a^n x(n)$	$X(a^{-1}z)$	$\|a\|r_2 < \|z\| < \|a\|r_1$
Time reversal	$x(-n)$	$X(z^{-1})$	$\frac{1}{r_1} < \|z\| < \frac{1}{r_2}$
Conjugation:	$x^*(n)$	$X^*(z^*)$	ROC
z-Differentiation:	$n\, x(n)$	$-z\frac{dX(z)}{dz}$	$r_2 < \|z\| < r_1$
Convolution:	$x_1(n) * x_2(n)$	$X_1(z)X_2(z)$	At least $ROC_1 \cap ROC_2$

Discrete Time Integrator



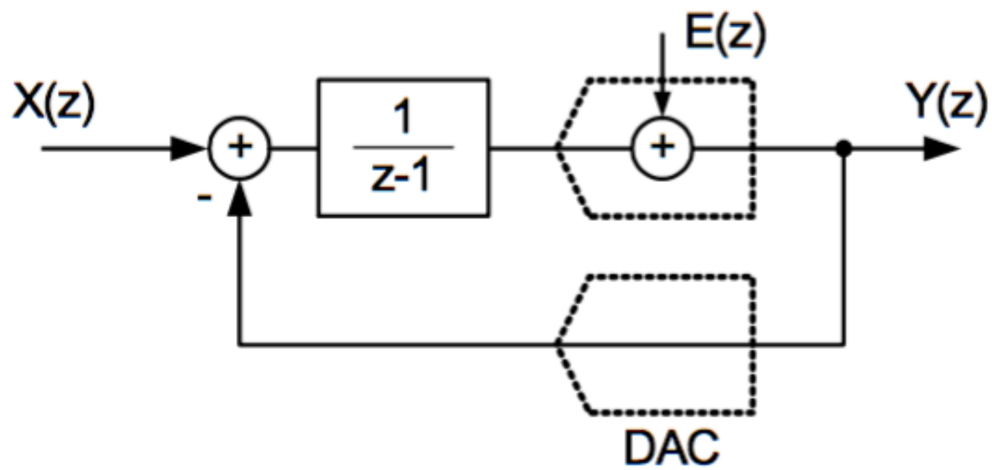
$$v(k) = u(k-1) + v(k-1)$$

$$V(z) = z^{-1}U(z) + z^{-1}V(z)$$

$$\frac{V(z)}{U(z)} = \frac{z^{-1}}{1-z^{-1}} = \frac{1}{z-1}$$

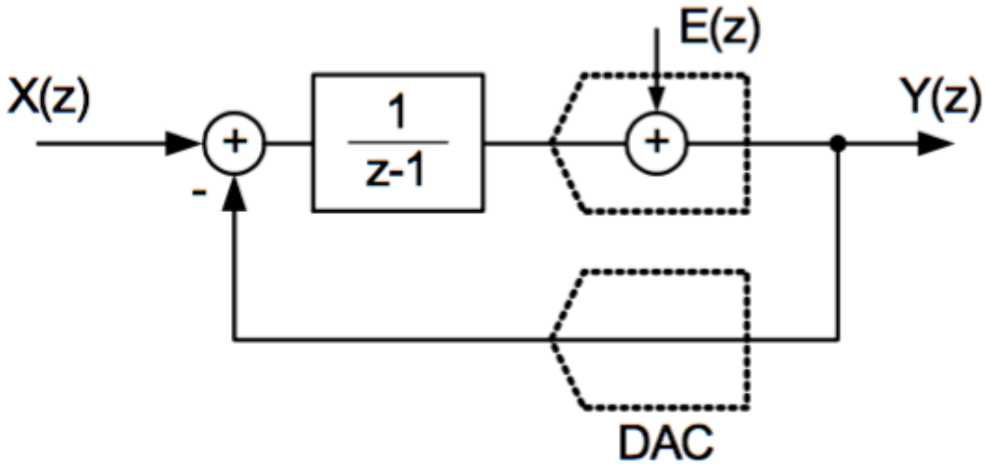
- "Infinite gain" at DC ($\omega=0$, $z=1$)

First Order Sigma-Delta Modulator



- Output is equal to delayed input plus filtered quantization noise

First Order Sigma-Delta Modulator



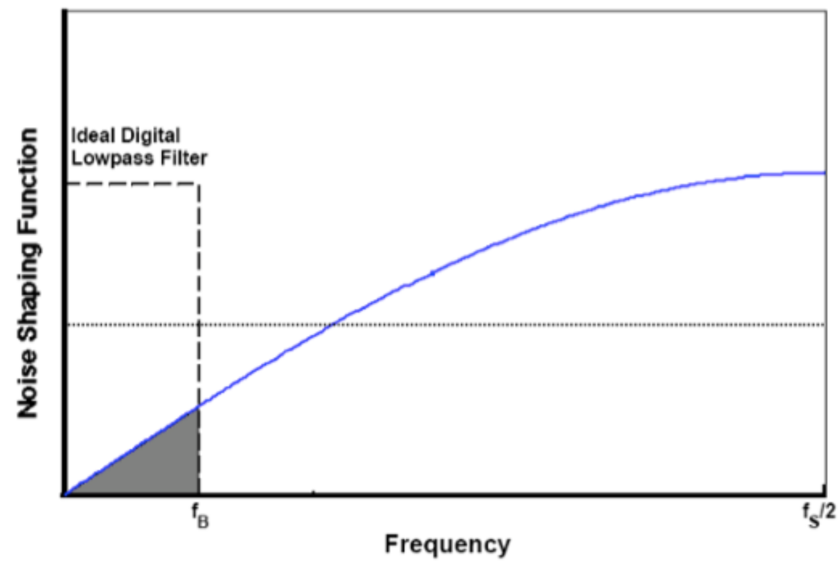
$$Y(z) = E(z)\frac{1}{1+\frac{1}{z-1}} + X(z)\frac{\frac{1}{z-1}}{1+\frac{1}{z-1}}$$

$$= E(z)\left(1 - z^{-1}\right) + X(z)z^{-1}$$

- Output is equal to delayed input plus filtered quantization noise

NTF Frequency Domain Analysis

$$H_e(z) = 1 - z^{-1}$$



- "First order noise Shaping"
  - Quantization noise is attenuated at low frequencies, amplified at high frequencies

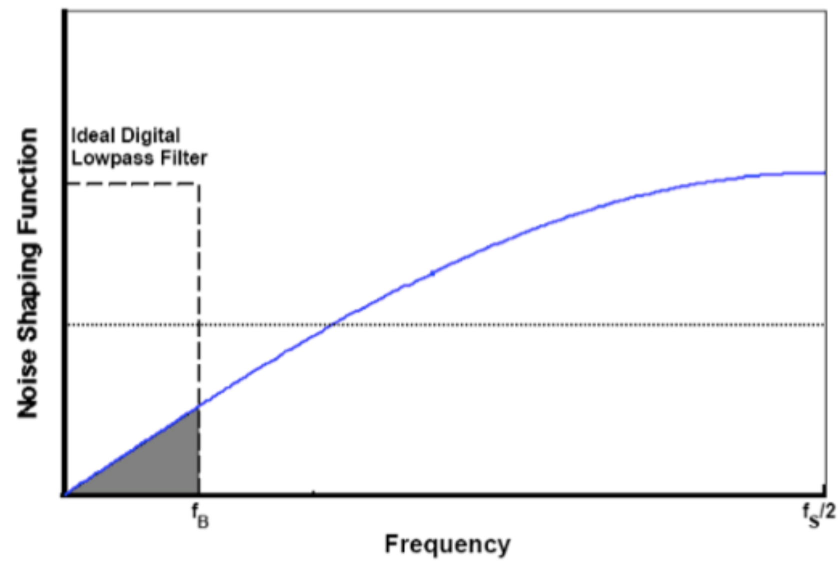
NTF Frequency Domain Analysis

$$H_e(z) = 1 - z^{-1}$$

$$H_e(j\omega) = \left(1 - e^{-j\omega T}\right) = 2e^{-j\omega T/2}\left(\frac{e^{j\omega T/2} - e^{-j\omega T/2}}{2}\right)$$

$$= 2e^{-j\frac{\omega T}{2}}\left(j\sin\left(\frac{\omega T}{2}\right)\right) = 2\sin\left(\frac{\omega T}{2}\right)e^{-j\frac{\omega T - \pi}{2}}$$

$$\left|H_e(f)\right| = 2\left|\sin(\pi f T)\right| = 2\left|\sin\left(\pi\frac{f}{f_s}\right)\right|$$

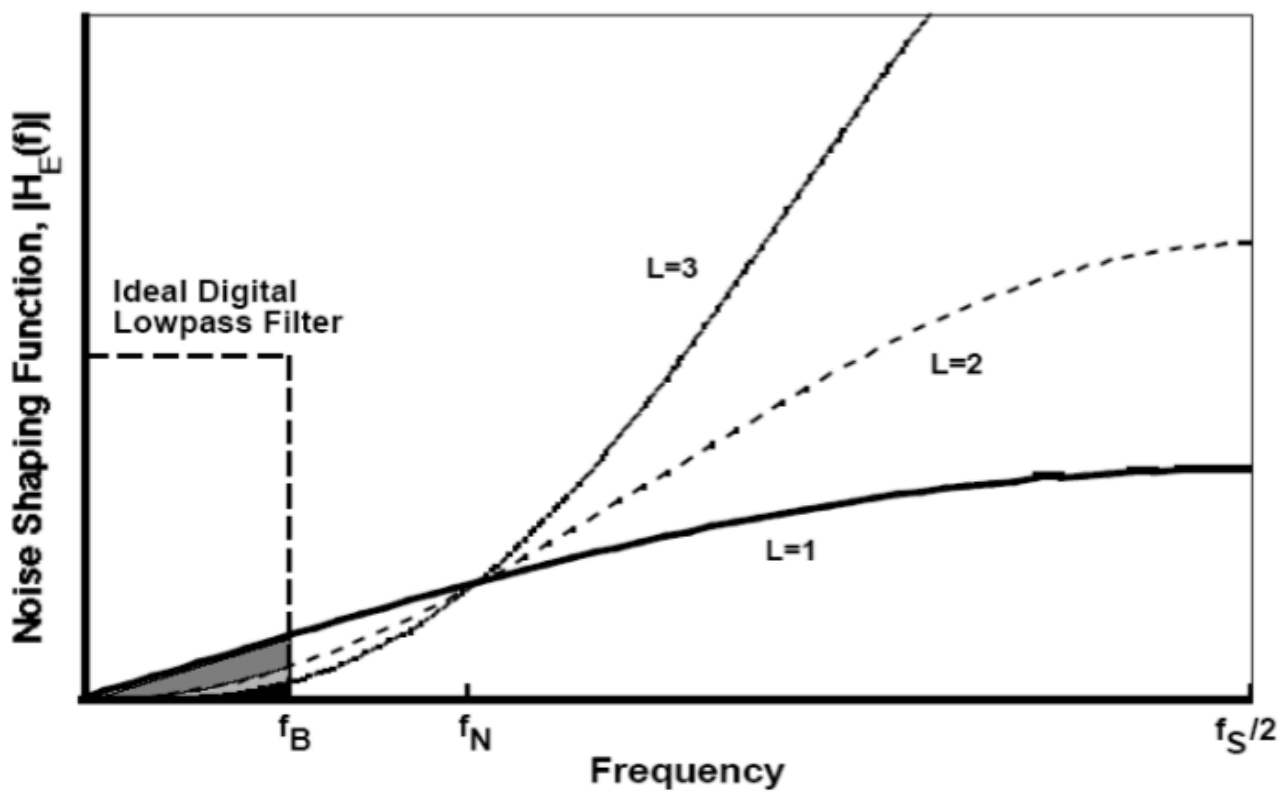


- "First order noise Shaping"
  - Quantization noise is attenuated at low frequencies, amplified at high frequencies

Higher Order Noise Shaping

- L^{th} order noise transfer function

$$H_E(z) = \left(1 - z^{-1}\right)^L$$



SQNR Improvement

- Example Revisited
  - Want 16-bit ADC, f_B=1MHz
  - Use oversampled 8-bit ADC, first order noise shaping and (ideal) digital lowpass filter

SQNR Improvement

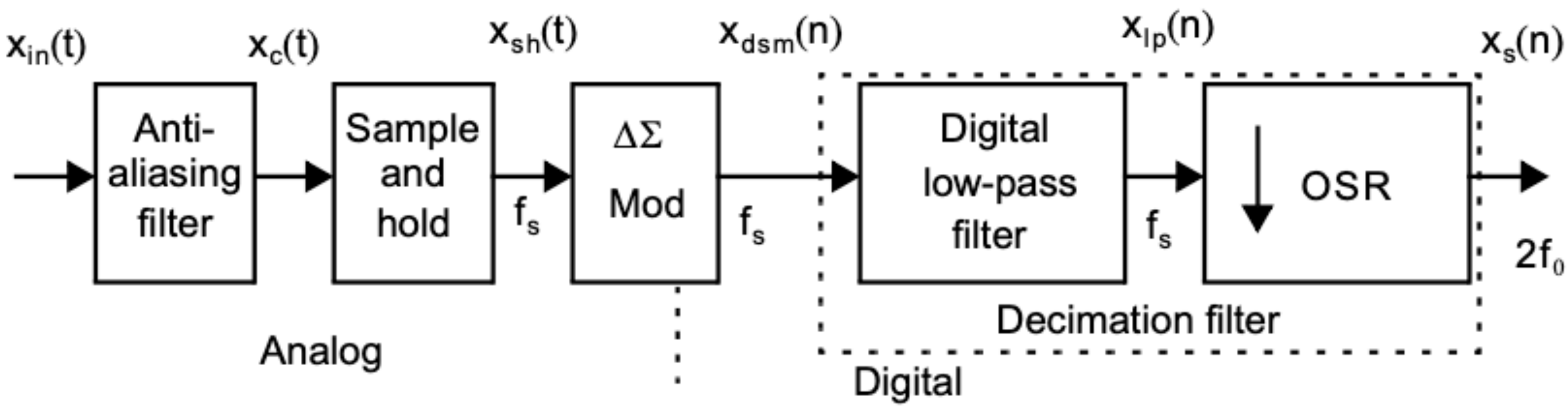
- Example Revisited
  - Want 16-bit ADC, f_B=1MHz
  - Use oversampled 8-bit ADC, first order noise shaping and (ideal) digital lowpass filter
    - SQNR improvement compared to case without oversampling is -5.2dB+30log(M)
  - 8-bit increase in resolution (48 dB SQNR improvement) would necessitate M≅60 → f_S=120MHz
- Not all that bad!

SQNR Improvement

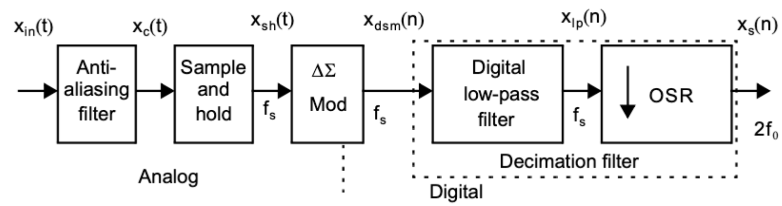
- Example Revisited
  - Want 16-bit ADC, f_B=1MHz
  - Use oversampled 8-bit ADC, first order noise shaping and (ideal) digital lowpass filter
  - 8-bit increase in resolution (48 dB SQNR improvement) would necessitate M≅60 → f_S=120MHz
- Not all that bad!

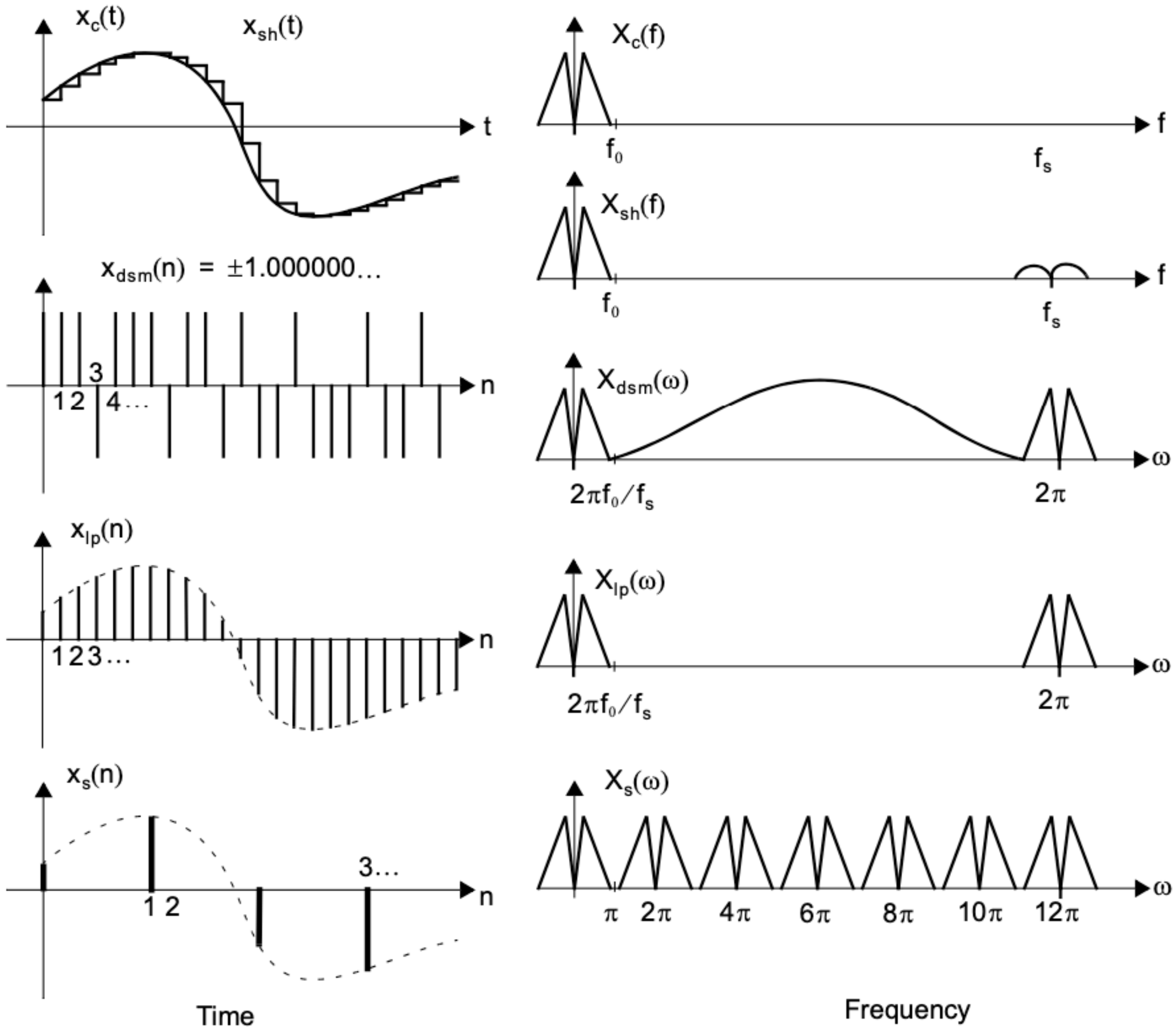
M	SQNR improvement
16	31dB (~5 bits)
256	67dB (~11 bits)
1024	85dB (~14 bits)

Decimation Filter



Decimation Filter





Big Ideas

- Oversampling
  - Enables reduction in quantization noise with digital filter
- Noise Shaping
  - Allows for further quantization noise reduction for a more practical oversampling factor
- Sigma-Delta ADCs
  - Use integrator in feedback to shape noise and achieve high resolution
  - Usually for low speed, low power applications
    - Suited for medical devices

Admin

- Lab tomorrow
  - DSP in Python