Non Ideal Anti-Aliasing

\[ X_c(j\Omega)H_{LP}(j\Omega) \]

- Problem: Hard to implement sharp analog filter
- Tradeoff:
  - Crop part of the signal
  - Suffer from noise and interference (See lab II !)

Ideal Anti-Aliasing

\[ x_c(t) \rightarrow \text{Analog Anti-Aliasing Filter } H_{LP}(j\Omega) \rightarrow \text{_sampler } x[n] = x_c(nT) \rightarrow \text{Quantizer} \]

\[ X_c(j\Omega) \quad \text{and} \quad \Omega_s < 2\Omega_N \]

\[ X_s(j\Omega) \]

Oversampled ADC

\[ x_c(t) \rightarrow \text{Sharp Analog Anti-Aliasing Filter } H_{LP}(j\Omega) \rightarrow \text{C/D } T \rightarrow x[n] = x_c(nT) \rightarrow \text{Quantizer} \]

\[ x_c(t) \rightarrow \text{Simple Analog Anti-Aliasing Filter} \rightarrow \text{C/D } T = \frac{1}{M} \frac{\pi}{\Omega_N} \rightarrow \text{Sharp Digital Anti-aliasing Filter } \frac{\pi}{MT} \rightarrow \downarrow M \rightarrow \text{Quantizer} \]
Oversampled ADC

\[ X_c(j\Omega)H_{LP}(j\Omega) \]

after oversampling x2

\[ \hat{X}(e^{j\omega}) \]

after digital LP and decimation

\[ \hat{X}_d(e^{j\omega}) \]

\[ T_d = MT \]

Sampling and Quantization

\[ x_c(t) \]

\[ x[n] = x_c(nT) \]

Quantizer

\[ \hat{x}[n] \]

\[ 2X_m \]

\[ \Delta \]
Sampling and Quantization

• for 2’s complement with $B+1$ bits $-1 \leq \hat{x}_B[n] < 1$

\[
\Delta = \frac{2X_m}{2^{B+1}} = \frac{X_m}{2^B}
\]

\[
\hat{x}[n] = X_m \hat{x}_B[n]
\]

Quantization Error

• Model quantization error as noise

\[
x[n] \xrightarrow{\text{Quantizer}} \hat{x}[n] = x[n] + e[n]
\]

• In that case:

\[
-\Delta/2 \leq e[n] < \Delta/2
\]

\[
(-X_m - \Delta/2) < x[n] \leq (X_m - \Delta/2)
\]

Noise Model for Quantization Error

• Assumptions:
  – Model $e[n]$ as a sample sequence of a stationary random process
  – $e[n]$ is not correlated with $x[n]$
  – $e[n]$ not correlated with $e[m]$, e.g., white noise
  – $e[n] \sim U[-\Delta/2, \Delta/2]$

• Result:
  – Variance is: $\sigma_e^2 = \frac{\Delta^2}{12}$, or $\sigma_e^2 = \frac{2^{-2B}X_m^2}{12}$ since $\Delta = 2^{-B}X_m$
  – Assumptions work well for signals that change rapidly, are not clipped and for small $\Delta$

Quantization Noise

Figure 4.57 (continued) (b) Quantized samples of the cosine waveform in part (a) with a 3-bit quantizer. (c) Quantization error sequence for 3-bit quantization of the signal in (a). (d) Quantization error sequence for 8-bit quantization of the signal in (a).
SNR of Quantization Noise

- For uniform B+1 bits quantizer:

\[ \sigma_e^2 = \frac{2^{-2B}X_m^2}{12} \]

\[ SNR_Q = 10 \log_{10} \left( \frac{\sigma_e^2}{\sigma_x^2} \right) \]

\[ = 10 \log_{10} \left( \frac{12 \cdot 2^B \sigma_x^2}{X_m^2} \right) \]

\[ SNR_Q = 6.02B + 10.8 - 20 \log_{10} \left( \frac{X_m}{\sigma_x} \right) \]

- Improvement of 6dB with every bit
- The range of the quantization must be adapted to the rms amplitude of the signal
  - Tradeoff between clipping and noise!
  - If \( \sigma_x = \frac{X_m}{4} \) then \( SNR_Q \approx 6B - 1.25dB \)
  - So SNR of 90-96 dB requires 16-bits (audio)

Practical ADC (Ch. 4.8.4)

- Scaled train of sinc pulses
- Difficult to generate sinc \( \Rightarrow \) Too long!

Practical ADC

- \( h_0 \) is finite length pulse \( \Rightarrow \) easy to implement
Practical ADC

Output: zero-order hold.

$x_0(t) = \sum_{k=-\infty}^{\infty} x(t-kT) h_0(t-nT) = h_0(t) * x_s(t)$

taking FT:

$X_r(j\omega) = H_r(j\omega) \cdot X_s(j\omega)$

$= H_r(j\omega) \cdot \frac{1}{T_0} \sum_{k=-\infty}^{\infty} X(j(\omega - k\Omega_s))$

Practical ADC

Ideally:

$X_s(j\omega) H_{LP}(j\omega)$

Practically:

$X_s(j\omega) H_0(j\omega)$
Practical ADC

Practically:

Easier Implementation with Digital upsampling

Practically:

Easier Implementation with Digital upsampling

Practically:

Harder