Lecture Outline

- Data Converters
  - Anti-aliasing
  - ADC
  - Quantization
  - Practical DAC
- Noise Shaping

ADC

Anti-Aliasing Filter with ADC

Oversampled ADC

Oversampled ADC
Effect of Quantization Error on Signal

- Quantization error is a deterministic function of the signal
  - Consequently, the effect of quantization strongly depends on the signal itself

- Unless, we consider fairly trivial signals, a deterministic analysis is usually impractical
  - More common to look at errors from a statistical perspective
    - "Quantization noise"

- Two aspects
  - How much noise power (variance) does quantization add to our samples?
  - How is this noise distributed in frequency?

Quantization Error

- Model quantization error as noise
  \[ x[n] \xrightarrow{\text{Quantizer}} \hat{x}[n] = x[n] + e[n] \]

- In that case:
  \[-\Delta/2 \leq e[n] < \Delta/2\]
  \[ -X_m - \Delta/2 \leq x[n] \leq X_m - \Delta/2 \]
Quantization step $\Delta$
- Quantization error has sawtooth shape.
- Bounded by $-\Delta/2, +\Delta/2$
- Ideally infinite input range and infinite number of quantization levels.

Ideal Quantizer

Quantization Error Statistics
- Crude assumption: $e_q(x)$ has uniform probability density.
- This approximation holds reasonably well in practice when
  - Signal spans large number of quantization steps
  - Signal is "sufficiently active"
  - Quantizer does not overload

Signal-to-Quantization-Noise Ratio
- For uniform B+1 bits quantizer
  \[ SNR_Q = 10 \log_{10} \left( \frac{\sigma_x^2}{\sigma_e^2} \right) = 10 \log_{10} \left( \frac{12 \cdot 2^{2B} \sigma_e^2}{X_m^2} \right) \]
- \[ SNR_Q = 6.02B + 10.8 - 20 \log_{10} \left( \frac{X_m}{\sigma_e} \right) \text{ Quantizer range /rms of amp} \]

Noise Model for Quantization Error
- Assumptions:
  - $e[n]$ as a sample sequence of a stationary random process
  - $e[n]$ is not correlated with $x[n]$
  - $e[n]$ not correlated with $e[m]$ where $m \neq n$ (white noise)
  - $e[n] \sim U[-\Delta/2, \Delta/2]$ (uniform pdf)
- 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 $\Delta$

Signal-to-Quantization-Noise Ratio
- 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!
  - Often use pre-amp
  - Sometimes use analog auto gain controller (AGC)
Assuming full-scale sinusoidal input, we have

\[
\text{SQNR} = \frac{P_{\text{in}}}{P_{\text{noise}}} = \frac{1}{2} \left( \frac{\sigma_x^2}{\Delta x^2} \right)^2 = 1.5 \times 2^{10} = 6.02 \text{dB} + 1.76 \text{dB}
\]

<table>
<thead>
<tr>
<th>B (Number of Bits)</th>
<th>SQNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>50dB</td>
</tr>
<tr>
<td>12</td>
<td>74dB</td>
</tr>
<tr>
<td>16</td>
<td>96dB</td>
</tr>
<tr>
<td>20</td>
<td>122dB</td>
</tr>
</tbody>
</table>

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

References

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

Quantization Noise with Oversampling
- Energy of \( x[n] \) equals energy of \( x[n] \)
  - No filtering of signal!
- Noise variance is reduced by factor of \( M \)

\[
\text{SNR}_Q = 6.02B + 10.8 \times 20 \log_{10} \left( \frac{X_m}{\sigma_q} \right) - 10 \log_{10} M
\]

For doubling of \( M \) we get 3dB improvement, which is the same as 1/2 a bit of accuracy
With oversampling of 16 with 8bit ADC we get the same quantization noise as 10bit ADC!
Practical DAC

\[ x[n] = x(t)|_{t=nT} \rightarrow \text{sinc pulse generator} \rightarrow x_s(t) = \sum_{n=-\infty}^{\infty} x[n] \sin(\frac{t-nT}{T}) \]

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

\[ h_0(t) \text{ is finite length pulse} \rightarrow \text{easy to implement} \]
- For example: zero-order hold

\[ H_0(j\Omega) = Te^{-\Omega^2T} \sin(\frac{\Omega}{\Omega_c}) \]

Practical DAC

**Zero-Order-Hold Interpolation**

\[ x_0(t) = \sum_{n=-\infty}^{\infty} x[n]h_0(t-nT) = h_0(t) \ast x_s(t) \]

Taking a FT:

\[ X_0(j\Omega) = H_0(j\Omega)X(j\Omega) = \frac{H_0(j\Omega)}{T} \sum_{k=-\infty}^{\infty} X(j(\Omega - k\Omega_c)) \]

Practical DAC

**Output of the reconstruction filter**

\[ X_s(j\Omega) = H_s(j\Omega) \ast X_0(j\Omega) \]

\[ = \frac{H_s(j\Omega)}{T} \sum_{k=-\infty}^{\infty} X(j(\Omega - k\Omega_c)) \]

Practically:

\[ X_s(j\Omega) / H_s(j\Omega) \]

Ideally:

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

Practically:
Practical DAC

\[ X_d(j\Omega) \]

Practically:

\[ X_d(j\Omega)H_0(j\Omega)H_1(j\Omega) \]

Practical DAC with Upsampling

\[ x[n] \rightarrow \frac{1}{L} x_c[n] \rightarrow \text{LPF} \]

\[ X_d(j\Omega) \]

Practically:

\[ X_d(j\Omega)H_0(j\Omega)H_1(j\Omega) \]

Quantization Noise with Oversampling

\[ T = \frac{\Omega_N M}{\sigma_e^2} \]

\[ X_c(j\Omega) \]

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

\[ X_d(e^{j\omega}) \]

Quantization Noise with Oversampling

- Energy of \( x_c[n] \) equals energy of \( x[n] \)
- No filtering of signal!
- Noise variance is reduced by factor of \( M \)

\[ \text{SNR}_Q = 6.02B + 10.8 - 20 \log_{10} \left( \frac{X_m}{\sigma_e} \right) - 10 \log_{10} M \]

- For doubling of \( M \) we get 3dB improvement, which is the same as 1/2 a bit of accuracy
- With oversampling of 16 with 8bit ADC we get the same quantization noise as 10bit ADC!

Noise Shaping

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

\[ Y(z) = E(z) + \frac{1}{1 + A(z)} \cdot X(z) \cdot A(z) \]

\[ Y(z) = E(z) + \frac{1}{1 + A(z)} + X(z) \cdot \frac{A(z)}{1 + A(z)} \]

\[ Y(z) = E(z)H(z) + X(z)H(z) \]

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...\( f_B \)) we achieve this by making |\( A(z) \)| \( \gg 1 \) at low frequencies
- Means that NTF \( \ll 1 \)
- Means that STF \( \approx 1 \)

Discrete Time Integrator

\[ v(k) = u(k) - u(k-1) \]

\[ V(z) = z^{-1}U(z) + z^{-2}V(z) \]

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

First Order Sigma-Delta Modulator

\[ Y(z) = E(z) + \frac{1}{1 + A(z)} + X(z) \cdot \frac{A(z)}{1 + A(z)} \]

- Output is equal to delayed input plus filtered quantization noise

NTF Frequency Domain Analysis

\[ H_N(f) = 1 - z^{-1} \]

\[ |H_N(f)|^2 = 2 \sin \left( \frac{\pi f}{f_s} \right)^2 \]

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

In-Band Quantization Noise

- Question: If we had an ideal digital lowpass, what is the achieved SQNR as a function of oversampling ratio?
- Can integrate shaped quantization noise spectrum up to \( f_B \) and compare to full-scale signal

\[ P_{\text{noise}} = \int \frac{2 + 2}{12 \cdot f_s} \left( 2 \sin \left( \frac{\pi f}{f_s} \right) \right)^2 df \]

\[ \approx \frac{\pi^2}{12} \cdot \frac{2}{f_s} \]

\[ \approx \frac{\pi^2}{3} \cdot \frac{1}{M^3} \]
**In-Band Quantization Noise**

- Assuming a full-scale sinusoidal signal, we have
  \[
  \text{SQNR} = 10 \log \left( \frac{P_{\text{max}}}{P_{\text{noise}}} \right) = 1.5 \times (2^8 - 1) \times \frac{3}{12} \times M^3
  \]
  \[
  \approx 1.76 \times 6.02B - 5.2 + 30 \log(M) \quad \text{[dB]} \quad \text{(for large B)}
  \]
- Each 2x increase in M results in 8x SQNR improvement
- Also added $\frac{1}{2}$ bit resolution

**Digital Noise Filter**

- Increasing M by 2x, means 3-dB reduction in quantization noise power, and thus $\frac{1}{2}$ bit increase in resolution
  - "$\frac{1}{2}$ bit per octave"
- Is this useful?
- Reality check
  - Want 16-bit ADC, $f_S = 1$MHz
  - Use oversampled 8-bit ADC with digital lowpass filter
  - 8-bit increase in resolution necessitates oversampling by 16 octaves
    \[
    f_s \geq 2 \cdot f_S \cdot M = 2 \cdot 1 \text{MHz} \cdot 2^{16}
    \geq 131 \text{GHz}
    \]

**SQNR Improvement**

- Example Revisited
  - Want 16-bit ADC, $f_S = 1$MHz
  - 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 $M \approx 60$ at $f_S = 120$MHz
- Not all that bad!

<table>
<thead>
<tr>
<th>M</th>
<th>SQNR improvement</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>31dB (~6 bits)</td>
</tr>
<tr>
<td>256</td>
<td>67dB (~11 bits)</td>
</tr>
<tr>
<td>1024</td>
<td>85dB (~14 bits)</td>
</tr>
</tbody>
</table>

**Higher Order Noise Shaping**

- $L$th order noise transfer function
  \[
  H_f(z) = \left(1 - z^{-1}\right)^L
  \]

**Big Ideas**

- Data Converters
  - Oversampling to reduce interference and quantization noise → increase ENOB (effective number of bits)
  - Practical DACs use practical interpolation and reconstruction filters with oversampling
- Noise Shaping
  - Use feedback to reduce oversampling factor

**Admin**

- HW 4 due tonight at midnight
  - Typo in code in MATLAB problem, corrected handout
  - See Piazza for more information
- HW 5 posted after class
  - Due in 1.5 weeks 3/3