Data Converters [Ch. 11]

- Critical parts of many systems, the biggest reason why analog design will never disappear.
- Can be broadly divided into two categories:
  1. Nyquist rate converters: Sample incoming signal at or near Nyquist rate (twice the signal freq.), at desired # of bits of resolution.

  ![Nyquist rate converter diagram]

  - In practice, sampling rate will be 1.5-10 times the Nyquist rate to relax the design of the anti-aliasing filter.

  2. Oversampling Converters: Sample incoming signal much faster than Nyquist rate with fewer bits of accuracy, use filtering to improve Signal-to-Noise Ratio (SNR).

  ![Oversampling converter diagram]

- The concept of oversampling converters will be more clear next class after we discuss examples in more detail.
- For now, let's focus on Nyquist rate converters.

Ideal D/A Converter

- Take N-bit digital input word: \( b_n 2^n + b_{n-1} 2^{n-1} + \cdots + b_1 2^1 + b_0 2^0 \).
- Generate a quantized analog output: \( V_{out} = V_{ref} \cdot b_n \).

- Useful to define voltage corresponding to the LSB: \( V_{LSB} = \frac{V_{ref}}{2^n} \).
- Min. voltage that the output can change by.

\[
\begin{align*}
& B_n \quad \text{N} \\
& \xrightarrow{\text{D/A}} \quad V_{out}
\end{align*}
\]
Ideal A/D Converter

\[ V_{in} \rightarrow \text{A/D} \rightarrow B_{out} \]

(digital)

- To generate a binary representation of an incoming analog voltage:
  \[ V_{in} \pm V_A = A_{ref}, B_{out} \text{ where } B_{out} = b_1 2^{-1} + b_2 2^{-2} + \cdots + b_N 2^{-N} \]
  and \( \frac{1}{2} V_{USB} \leq V_A < \frac{1}{2} V_{USB} \).

- \( V_A \) is the quantization error, and results from the stepped nature of the output.

**Quantization Noise**

- Occurs in all A/D converters (even ideal). The "noise" designation is a bit misleading as it is not really random (like thermal noise).

Another view of \( V_A \):

\[ V_{in} \rightarrow \text{A/D} \rightarrow \text{D/A} \rightarrow V_A \]

- We can write \( V_A = V_I - V_{in} \).
- Rearrange to get \( V_I = V_{in} + V_A \).

- So, output of A/D is input signal \( V_{in} \) plus some added "noise" \( V_A \).

- Called noise because it changes as the input signal changes.

- Let's look at quantization noise for a ramp input:
verage level of $V_a(t)$ is 0, but we can find the rms level:

$$V_{a,\text{rms}} = \sqrt{\frac{1}{T} \int_{0}^{T} V_a^2(t) \, dt}$$

$$V_{\text{rms}} = \frac{V_{\text{ref}}}{\sqrt{2}}$$

So, increasing # of bits $(N)$ reduces $V_{\text{rms}}$ and reduces quantization noise.

Can define a SNR for output signal, and calculate an upper bound based on quantization noise.

For a full-scale (0 to $V_{\text{ref}}$) sinusoidal input, can show that $\text{SNR} = 6.02N + 17.6$ [dB]

**Data Converter Performance Metrics**

**Resolution**: The number of distinct output levels, given as number of bits.
- eg. 4 bits would be a 2-bit resolution converter.

**Offset & Gain Error**: Difference in zero crossing and slope of transfer function.

**Accuracy**: Difference between actual and expected transfer function.
- Also specified in # of bits.
- eg. an ADC with 4-bit accuracy has error $\lt \frac{V_{\text{ref}}}{2^4}$.

**Integral Non-Linearity (INL)**: After removal of gain & offset errors, deviation of transfer from straight line.

**Differential Non-Linearity (DNL)**: Deviation of successive steps from ideal value of $\frac{V_{\text{ref}}}{2^{N}}$.

**Sample Rate**: max. operating speed.
Nyquist Rate D/A Converters [Ch. 12]

- Three main types:
  1. Decoder-Based
  2. Binary-Weighted
  3. Thermometer-Coded

- We will discuss pros/cons of each type as well as give an example.

- **Decoder-Based**
  
  - Very simple: Create \(2^n\) different reference voltages and multiplex the right one to the output according to the incoming digital word.

  - Example:

  
  \[
  V_{out} = \sum_{i=0}^{n-1} b_i \cdot 2^i 
  \]

  - Example MUX implementation:

  ![MUX Diagram]

  - **Advantages:**
    - Guaranteed to have monotonic output.

  - **Disadvantages:**
    - Accuracy depends on resistor matching.
    - Limited speed since signal needs to propagate through MUX.

  - MUX above can be modeled as: \( \frac{1}{sC} \) \( \frac{1}{sC} \)

  - More bits adds more RC sections, slowing it down (ZVTC)

  - Many variations on this type of DAC, many bases on using different MUX architectures.

- **Binary-Weighted**

  - Use binary weighting instead of generating every required output.

  - Saw 2 examples of these in 3110: Binary-Weighted Resistor + R-2R Ladder.

  - Example: Switched-Capacitor Implementation based on Charge-redistribution.
\[ Q = b_1 \text{ } VC + b_2 2C \text{ } VC + b_3 C \text{ } VC \]

\[ = Cb \text{ } VC \left( b_1 2^{-1} + b_2 2^{-1} + b_3 2^{-1} \right) \]

- When \( \Phi_2 \) goes high, charge flows onto \( 8C \), generating an output voltage \( V_{out} \), such that:

\[ V_{out} = \frac{Q}{8C} = V_{ref} \left( b_1 2^{-1} + b_2 2^{-1} + b_3 2^{-1} \right) \]

Advantages: - Easier to build higher resolution DACs, complexity scales linearly with \( N \) instead of \( 2^N \).

Disadvantages: - Can be subject to glitches if bits change at different times.

- **Example**: Consider transition from \( 100 \) to \( 011 \) in a 3-bit counter.
  - If \( b_1 \) changes before \( b_2 \) or \( b_3 \), will have a glitch at output with output \( 0 \), temporarily \( 000 \).

**3. Thermometer-Code Converter**

- Thermometer code converters solve the glitch problem of the previous type at the expense of some added complexity.

- Thermometer code uses \( 2^N - 1 \) bits to represent \( 2^N \) levels, as opposed to normal binary representation, which only requires \( N \) levels.

- Example for 4 levels:

<table>
<thead>
<tr>
<th>Decimal</th>
<th>Binary</th>
<th>Thermometer</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>000</td>
<td>000</td>
</tr>
<tr>
<td>1</td>
<td>001</td>
<td>001</td>
</tr>
<tr>
<td>2</td>
<td>010</td>
<td>011</td>
</tr>
<tr>
<td>3</td>
<td>011</td>
<td>111</td>
</tr>
</tbody>
</table>

- Eliminates possibility of glitches in moving between adjacent values.
Example: Binary Weighted Charge Redistribution DAC can be changed for Thermometer Coding.

- not too much work in area due to passives. (7C for both).

- Hybrid Converters

- Some DACs use combinations of the aforementioned techniques.

- Example: Thermometer coding for MSBs with glitching will have the largest impact, binary weighting for LSBs to save on space & complexity.