EE 505

Lecture 5

Data Converter Operation and Characterization

-- Quantization Noise
INL-based ENOB

Since the break-point INL is ideally 0, it is not related to either $X_{\text{LSB}}$ or $X_{\text{REF}}$. As such, the magnitude of the break-point INL is independent of the resolution. It is thus difficult to naturally define the effective number of bits (ENOB) directly from the INL. However, since the gain (from input to interpreted output) of an ADC is ideally 1, the break-point INL is conveying about the same linearity information as the continuous-input INL. As such, the ENOB based upon the break-point INL is also defined by the same expression.

The ENOB based upon INL for both DACs and for ADCs is defined by the expression

$$\text{ENOB} = n_R - 1 - \log_2 (\nu)$$

where $n_R$ is the specified resolution and $\nu$ is the resolution in LSB at the $n_R$ bit level.
INL-based ENOB

Theorem: The INL ENOB is an inherent property of a data converter independent of the number of bits of resolution specified for a data converter.

Proof: Assume a data converter has $n_{RA}$ bits of resolution and an INL of $\nu_A$ LSB and the same data converter was specified with $n_{RB}$ bits of resolution and an INL of $\nu_B$ LSB.

Based upon the first specification, the INL can be expressed as

$$\text{INL}_A = \frac{\nu_A X_{\text{LSB}}}{X_{\text{REF}}} \quad (1)$$

But since it is assumed to have $n_{RA}$ bits of resolution

$$\frac{X_{\text{LSB}}}{X_{\text{REF}}} = 2^{-n_{RA}} \quad (2)$$

Thus we obtain the expression

$$\text{INL}_A = \nu_A 2^{-n_{RA}} \quad (3)$$
INL-based ENOB

Since ENOB as given by \( \text{ENOB} = n_R - 1 - \log_2 (\nu) \)

has been shown to be independent of \( n_R \), one should be able to express the ENOB in terms of the INL without the explicit presence of \( n_R \).

Theorem: The INL-based ENOB can be equivalently expressed as

\[
\text{ENOB} = -\log_2 (\text{INL}_{\text{REF}}) - 1
\]

where \( \text{INL}_{\text{REF}} \) is the INL expressed relative to \( X_{\text{REF}} \).

Note: The subscript on the INL was added strictly to emphasize in which form the INL is expressed and is usually not included.

Proof: (actually follows directly from definition of \( \text{INL} = (1/(2^{\theta})) \) but will repeat details)

By definition, \( \text{INL}_{\text{REF}} = \theta X_{\text{REF}} \)

but \( \nu = \frac{\theta \cdot X_{\text{REF}}}{X_{\text{LSB}}} \)
Performance Characterization of Data Converters

• Static characteristics
  – Resolution
  – Least Significant Bit (LSB)
  – Offset and Gain Errors
  – Absolute Accuracy
  – Relative Accuracy
  – Integral Nonlinearity (INL)
  – Differential Nonlinearity (DNL)
  – Monotonicity (DAC)
  – Missing Codes (ADC)
  – Low-f Spurious Free Dynamic Range (SFDR)
  – Low-f Total Harmonic Distortion (THD)
  – Effective Number of Bits (ENOB)
  – Power Dissipation
Differential Nonlinearity (DAC)

Nonideal DAC

\[ \Delta X_{\text{OUT}}(k) = X_{\text{OUT}}(k) - X_{\text{OUT}}(k-1) \]

Increment at code 4

\[ \text{DNL}(k) = \frac{X_{\text{OUT}}(k) - X_{\text{OUT}}(k-1) - X_{\text{LSB}}}{X_{\text{LSB}}} \]

DNL(k) is the actual increment from code (k-1) to code k minus the ideal increment normalized to X_{LB}. 

\[ \text{DNL}(k) = \frac{X_{\text{OUT}}(k) - X_{\text{OUT}}(k-1) - X_{\text{LSB}}}{X_{\text{LSB}}} \]
Theorem: The INL<sub>k</sub> of a DAC can be obtained from the DNL by the expression

\[ \text{INL}_k = \sum_{i=1}^{k} \text{DNL}(i) \]

Caution: Be careful about using this theorem to measure the INL since errors in DNL measurement (or simulation) can accumulate.

Corollary: \( \text{DNL}(k) = \text{INL}_k - \text{INL}_{k-1} \)
Differential Nonlinearity (ADC)

Nonideal ADC

\[ \text{DNL}(k) = \frac{X_{T(k+1)} - X_{T_k} - X_{\text{LSB}}}{X_{\text{LSB}}} \]

\( X_{\text{LSB}} \) is the code width for code \( k \) – ideal code width normalized to \( X_{\text{LSB}} \).
Monotonicity in an ADC

Definition: An ADC is monotone if the

\[ \bar{X}_{OUT}(x_k) \geq \bar{X}_{OUT}(x_m) \quad \text{whenever} \quad x_k \geq x_m \]

Note: Have used \( x_{Bk} \) instead of \( x_{Tk} \) since more than one transition point to a given code

Note: Some authors do not define monotonicity in an ADC.
**Missing Codes (ADC)**

### Nonideal ADCs

- **No missing codes**
  - Definition: An ADC has no missing codes if there are N-1 transition points and a single LSB code increment occurs at each transition point. If these criteria are not satisfied, we say the ADC has missing code(s).
  - Note: With this definition, all codes can be present but we still say it has “missing codes”

- **One missing code**
  - Note: Some authors claim that missing codes in an ADC are the counterpart to nonmonotonicity in a DAC. This association is questionable.
Weird Things Can Happen

Nonideal ADCs

• Multiple outputs for given inputs
• All codes present but missing codes

Be careful on definition and measurement of linearity parameters to avoid having weird behavior convolute analysis, simulation or measurements

Most authors (including manufacturers) are sloppy with their definitions of data converter performance parameters and are not robust to some weird operation
ENOB based upon DNL

If it is assumed that an acceptable DNL for an n-bit data converter is $X_{\text{LSB}}/2$, then if the DNL is different from $X_{\text{LSB}}/2$, the effective number of bits essentially changes.

An ENOB based upon the DNL can be defined (homework problem)
ENOB relative to resolution

If an n-bit data converter has an INL of $\frac{1}{4}$ LSB, it is really performing from a linearity viewpoint at the n+1 bit level and if it has an INL of $\frac{1}{8}$ LSB it is really performing at the n+2 bit levels.

Correspondingly, if it has a DNL of $\frac{1}{4}$ LSB, it is also performing from a differential linearity viewpoint at the n+1 bit level.

Observation: The ENOB of a data converter can exceed the number of bits of resolution of the data converter.

Observations: Some applications benefit from an ENOB that exceeds the resolution of the data converter.
Limitations of INL & DNL in Characterizing Linearity

Consider the following 4 transfer characteristics, all of which have the same INL.
Limitations of INL & DNL in Characterizing Linearity
Limitations of INL & DNL in Characterizing Linearity

Although same INL, dramatic difference in performance particularly when Inputs are sinusoidal-type excitations

INL also gives little indication of how performance degrades at higher frequencies

Spectral Analysis often used as an alternative (and often more useful) linearity measure for data converters
Linearity Measurements (testing)

Consider ADC

Linearity testing often based upon code density testing

Code density testing:

- Ramp or multiple ramps often used for excitation
- Linearity of test signal is critical (typically 3 or 4 bits more linear than DUT)
Linearity Measurements (testing)

Code density testing:

- First and last bins generally have many extra counts (and thus no useful information)
- Typically average 16 or 32 hits per code
Linearity Measurements (testing)

Code density testing:

\[ \bar{C} = \frac{\sum_{i=1}^{N-2} \hat{C}_i}{N-2} \]

\[ \text{DNL}_i = \frac{\hat{C}_i - \bar{C}}{\bar{C}} \]

\[ \text{INL}_i = \begin{cases} 0 & i=0, N-2 \\ \left[ \sum_{k=1}^{i} \hat{C}_k \right] - i\bar{C} & 1 \leq i \leq N-3 \end{cases} \]

\[ \text{DNL} = \max_{1 \leq i \leq N-2} \{|\text{DNL}_i|\} \]

\[ \text{INL} = \max_{1 \leq i \leq N-3} \{|\text{INL}_i|\} \]
Linearity Measurements (testing)

Code density testing:

\[
\text{DNL}_i = \frac{\hat{C}_i - \overline{C}}{\overline{C}} \\
\text{INL}_i = \left\{ \begin{array} {l}
0 \\
\sum_{k=1}^{i} \hat{C}_k - i\overline{C}
\end{array} \right. \\
\text{DNL} = \max_{1 \leq i \leq N-2} \{|\text{DNL}_i|\} \\
\text{INL} = \max_{1 \leq i \leq N-3} \{|\text{INL}_i|\}
\]

- Code Density Measurements are Indirect Measurements of the INL and DNL
- Can give very wrong information under some nonmonotone missing code scenarios
- Often use an average of 16 or 32 samples per code
- Measurement noise often 1 lsb or larger but averages out
- Sometimes use good sinusoidal waveform but must correct code density for this distinction
- Full code-density testing is costly for high-resolution low-speed data converters because of data acquisition costs
- Reduced code testing using servo methods is often a less costly alternative but may miss some errors
Performance Characterization of Data Converters

- Static characteristics
  - Resolution
  - Least Significant Bit (LSB)
  - Offset and Gain Errors
  - Absolute Accuracy
  - Relative Accuracy
  - Integral Nonlinearity (INL)
  - Differential Nonlinearity (DNL)
  - Monotonicity (DAC)
  - Missing Codes (ADC)
  - Quantization Noise
  - Low-f Spurious Free Dynamic Range (SFDR)
  - Low-f Total Harmonic Distortion (THD)
  - Effective Number of Bits (ENOB)
  - Power Dissipation
Quantization Noise

• DACs and ADCs generally quantize both amplitude and time
• If converting a continuous-time signal (ADC) or generating a desired continuous-time signal (DAC) these quantizations cause a difference in time and amplitude from the desired signal
• First a few comments about Noise
We will define “Noise” to be the difference between the actual output and the desired output of a system.

Types of noise:

• Random noise due to movement of electrons in electronic circuits
• Interfering signals generated by other systems
• Interfering signals generated by a circuit or system itself
• Error signals associated with imperfect signal processing algorithms or circuits
Noise

We will define “Noise” to be the difference between the actual output and the desired output of a system.

All of these types of noise are present in data converters and are of concern when designing most data converters.

Can not eliminate any of these noise types but with careful design can manage their effects to certain levels.

Noise (in particular the random noise) is often the major factor limiting the ultimate performance potential of many if not most data converters.
Noise

We will define “Noise” to be the difference between the actual output and the desired output of a system.

Types of noise:

- Random noise due to movement of electrons in electronic circuits
- Interfering signals generated by other systems
- Interfering signals generated by a circuit or system itself
- Error signals associated with imperfect signal processing algorithms or circuits

Quantization noise is a significant component of this noise in ADCs and DACs and is present even if the ADC or DAC is ideal.
Quantization Noise in ADC
(same concepts apply to DACs)

Consider an Ideal ADC with first transition point at 0.5$X_{LSB}$

If the input is a low frequency sawtooth waveform of period $T$ that goes from 0 to $X_{REF}$, the error signal in the time domain will be:

This time-domain waveform is termed the Quantization Noise for the ADC with a saw-tooth (or triangular) input
Quantization Noise in ADC

For large $n$, this periodic waveform behaves much like a random noise source that is uncorrelated with the input and can be characterized by its RMS value which can be obtained by integrating over any interval of length $T_1$. For notational convenience, shift the waveform by $T_1/2$ units

\[
E_{\text{RMS}} = \sqrt{\frac{1}{T_1/2} \int_{T_1 - T_1/2}^{T_1/2} \varepsilon_Q^2(t) dt}
\]
Quantization Noise in ADC

In this interval, $\varepsilon_Q$ can be expressed as

$$\varepsilon_Q(t) = -\left(\frac{X_{\text{LSB}}}{T_1}\right)t$$
Quantization Noise in ADC

\[
E_{\text{RMS}} = \sqrt{\frac{1}{T_1} \int_{-T_1/2}^{T_1/2} \varepsilon_Q^2(t) \, dt}
\]

\[
E_{\text{RMS}} = \sqrt{\frac{1}{T_1} \int_{-T_1/2}^{T_1/2} \left(-\frac{\Delta_{\text{LSB}}}{T_1}\right)^2 t^2 \, dt}
\]

\[
E_{\text{RMS}} = \Delta_{\text{LSB}} \sqrt{\frac{1}{T_1^3} \int_{-T_1/2}^{T_1/2} t^3 \, dt}
\]

\[
E_{\text{RMS}} = \frac{\Delta_{\text{LSB}}}{\sqrt{12}}
\]
Quantization Noise in ADC

\[ E_{\text{RMS}} = \frac{X_{\text{LSB}}}{\sqrt{12}} \]

The signal to quantization noise ratio (SNR) can now be determined. Since the input signal is a saw-tooth waveform of period T and amplitude \( X_{\text{REF}} \), it follows by the same analysis that it has an RMS value of

\[ X_{\text{RMS}} = \frac{X_{\text{REF}}}{\sqrt{12}} \]

Thus the SNR is given by

\[ \text{SNR} = \frac{X_{\text{RMS}}}{E_{\text{RMS}}} = \frac{X_{\text{RMS}}}{X_{\text{LSB}}} = 2^n \]

or, in dB,

\[ \text{SNR}_{\text{dB}} = 20(n \cdot \log_2) = 6.02n \]

Note: dB subscript often neglected when not concerned about confusion.
Quantization Noise in ADC

How does the SNR change if the input is a sinusoid that goes from 0 to $X_{\text{REF}}$ centered at $X_{\text{REF}}/2$?
Quantization Noise in ADC

How does the SNR change if the input is a sinusoid that goes from 0 to $x_{\text{REF}}$ centered at $x_{\text{REF}}/2$?

Time and amplitude quantization points
Quantization Noise in ADC

How does the SNR change if the input is a sinusoid that goes from 0 to $X_{\text{REF}}$ centered at $X_{\text{REF}}/2$?

Time and Amplitude Quantized Waveform
Quantization Noise in ADC

How does the SNR change if the input is a sinusoid that goes from 0 to $X_{\text{REF}}$ centered at $X_{\text{REF}}/2$?

![Error waveform](image-url)
Quantization Noise in ADC

How does the SNR change if the input is a sinusoid that goes from 0 to $\mathcal{X}_{\text{REF}}$ centered at $\mathcal{X}_{\text{REF}}/2$?

- Appears to be highly uncorrelated with input even though deterministic
- Mathematical expression for $\varepsilon_Q$ very messy
- Excursions exceed $X_{\text{LSB}}$ (but will be smaller and bounded by $\pm X_{\text{LSB}}/2$ for lower frequency signal/frequency clock ratios)
- For lower frequency inputs and higher resolution, at any time, errors are approximately uniformly distributed between $-X_{\text{LSB}}/2$ and $X_{\text{LSB}}/2$
- Analytical form for $\varepsilon_{\text{QRMS}}$ essentially impossible to obtain from $\varepsilon_Q(t)$
Quantization Noise in ADC

How does the SNR change if the input is a sinusoid that goes from 0 to $X_{\text{REF}}$ centered at $X_{\text{REF}}/2$?

For low $f_{\text{SIG}}/f_{\text{CL}}$ ratios, bounded by ±XLB and at any point in time, behaves almost as if a uniformly distributed random variable

$$\varepsilon_Q \sim U[-0.5X_{\text{LSB}}, 0.5X_{\text{LSB}}]$$
Quantization Noise in ADC

Recall:

If the random variable \( f \) is uniformly distributed in the interval \([A,B]\)
\[ f : U[A,B] \]
then the mean and standard deviation of \( f \) are given by

\[
\mu_f = \frac{A+B}{2} \quad \sigma_f = \frac{B-A}{\sqrt{12}}
\]

If \( n(t) \) is a random process, then for large \( T \),

\[
V_{RMS} = \sqrt{\int_{t_1}^{t_1+T} n^2(t) dt} = \sqrt{\sigma_n^2 + \mu_n^2}
\]

Theorem:
Quantization Noise in ADC

How does the SNR change if the input is a sinusoid that goes from 0 to $X_{REF}$ centered at $X_{REF}/2$?

$$\varepsilon_Q \sim U[-0.5X_{LSB}, 0.5X_{LSB}]$$

$$\mu_{\varepsilon_Q} = \frac{A+B}{2} = 0 \quad \sigma_f = \frac{B-A}{\sqrt{12}} = \frac{X_{LSB}}{\sqrt{12}}$$

$$V_{RMS} = \sqrt{\frac{1}{T} \int_{t_1}^{t_1+T} n^2(t) \, dt} = \sqrt{\sigma_n^2 + \mu_n^2}$$

$$V_{RMS} = \sigma_{\varepsilon_Q} = \frac{X_{LSB}}{\sqrt{12}}$$

Note this is the same RMS noise that was present with a triangular input
Quantization Noise in ADC

How does the SNR change if the input is a sinusoid that goes from 0 to $x_{\text{REF}}$ centered at $x_{\text{REF}}/2$?

$$V_{\text{RMS}} = \frac{x_{\text{LSB}}}{\sqrt{12}}$$

But

$$V_{\text{INRMS}} = \left(\frac{x_{\text{REF}}}{2}\right) \frac{1}{\sqrt{2}}$$

Thus obtain

$$\text{SNR} = \frac{x_{\text{REF}}}{2\sqrt{2}} = 2^n \frac{3}{2}$$

Finally, in db,

$$\text{SNR}_{\text{dB}} = 20\log \left(2^n \frac{3}{2}\right) = 6.02n + 1.76$$
ENOB based upon Quantization Noise

\[ \text{SNR} = 6.02 \, n - 1.76 \]

Solving for \( n \), obtain

\[ \text{ENOB} = \frac{\text{SNR}_{\text{dB}} - 1.76}{6.02} \]

Note: could have used the \( \text{SNR}_{\text{dB}} \) for a triangle input and would have obtained the expression

\[ \text{ENOB} = \frac{\text{SNR}_{\text{dB}}}{6.02} \]

But the earlier expression is more widely used when specifying the ENOB based upon the noise level present in a data converter.
ENOB based upon Quantization Noise

For very low resolution levels, the assumption that the quantization noise is uncorrelated with the signal is not valid and the ENOB expression will cause a modest error

$$\text{SNR}_{\text{corr}} = \left(2^{n-2} + \frac{4}{\pi}\right)\sqrt{\frac{3}{2}}$$

from van de Plassche (p13)

<table>
<thead>
<tr>
<th>Res (n)</th>
<th>SNR corr</th>
<th>SNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.86</td>
<td>7.78</td>
</tr>
<tr>
<td>2</td>
<td>12.06</td>
<td>13.8</td>
</tr>
<tr>
<td>3</td>
<td>19.0</td>
<td>19.82</td>
</tr>
<tr>
<td>4</td>
<td>25.44</td>
<td>25.84</td>
</tr>
<tr>
<td>5</td>
<td>31.66</td>
<td>31.86</td>
</tr>
<tr>
<td>6</td>
<td>37.79</td>
<td>37.88</td>
</tr>
<tr>
<td>8</td>
<td>49.90</td>
<td>49.92</td>
</tr>
<tr>
<td>10</td>
<td>61.95</td>
<td>61.96</td>
</tr>
</tbody>
</table>

Table values in dB

Almost no difference for \( n \geq 3 \)

$$\text{SNR} = 6.02 \times n + 1.76$$
Quantization Noise

Effects of quantization noise can be very significant, even at high resolution, when signals are not of maximum magnitude.

Quantization noise remains constant but signal level is reduced.

The desire to use a data converter at a small fraction of full range is one of the major reasons high resolution is required.
Quantization Noise

Effects of quantization noise can be very significant, even at high resolution, when signals are not of maximum magnitude.

\[ x_{\text{REF}} \]

\[ x_{\text{IN}} \]

\[ t \]
Quantization Noise

Example: If a 14-bit audio output is derived from a DAC designed for providing an output of 100W but the normal listening level is at 50mW, what is the SNR due to quantization noise at maximum output and at the normal listening level? What is the ENOB of the audio system when operating at 50mW?

At 100W output, SNR = 6.02n + 1.76 = 90.6dB

\[
\frac{V^2}{R_L} = 100W \quad \frac{V_1^2}{R_L} = 50mW \quad V_1 = \frac{V}{44.7}
\]

\[
20\log_{10} V_1 = 20\log_{10} V - 20\log_{10} 44.7 = 20\log_{10} V - 33dB
\]

At 50mW output, SNR reduced by 33dB to 57.6dB

\[
\text{ENOB} = \frac{\text{SNR}_{\text{dB}} - 1.76}{6.02} = \frac{57.6 - 1.76}{6.02} = -9.3
\]

Note the dramatic reduction in the effective resolution of the DAC when operated at only a small fraction of full-scale.
ENOB Summary

Resolution:

\[
\text{ENOB} = \frac{\log_{10} N_{\text{ACT}}}{\log_{10} 2} = \log_2 N_{\text{ACT}}
\]

INL:

\[
\text{ENOB} = n_R - \log_2 (\nu) - 1\quad n_R \text{ specified res, } \nu \text{ INL in LSB}
\]

\[
\text{ENOB} = -\log_2 (\text{INL}_{\text{REF}}) - 1\quad \text{INL}_{\text{REF}} \text{ INL rel to } X_{\text{REF}}
\]

DNL:

HW problem

Quantization noise:

\[
\text{ENOB} = \frac{\text{SNR}_{\text{dB}}}{6.02}\quad \text{rel to triangle/sawtooth}
\]

\[
\text{ENOB} = \frac{\text{SNR}_{\text{dB}} - 1.76}{6.02}\quad \text{rel to sinusoid}
\]
Performance Characterization of Data Converters

• Static characteristics
  – Resolution
  – Least Significant Bit (LSB)
  – Offset and Gain Errors
  – Absolute Accuracy
  – Relative Accuracy
  – Integral Nonlinearity (INL)
  – Differential Nonlinearity (DNL)
  – Monotonicity (DAC)
  – Missing Codes (ADC)
  – Quantization Noise
  – Low-f Spurious Free Dynamic Range (SFDR)
  – Low-f Total Harmonic Distortion (THD)
  – Effective Number of Bits (ENOB)
  – Power Dissipation
Absolute Accuracy

Absolute Accuracy is the difference between the actual output and the ideal or desired output of a data converter.

The ideal or desired output is in reference to an absolute standard (often maintained by the National Bureau of Standards) and could be volts, amps, time, weight, distance, or one of a large number of other physical quantities.

Absolute accuracy provides no tolerance to offset errors, gain errors, nonlinearity errors, quantization errors, or noise.

In many applications, absolute accuracy is not of a major concern.

but … scales, meters, etc. may be more concerned about Absolute accuracy than any other parameter.
Relative Accuracy

In the context of data converters, pseudo-static Relative Accuracy is the difference between the actual output and an appropriate fit-line to overall output of the data converter.

INL is often used as a measure of the relative accuracy.

In many, if not most, applications, relative accuracy is of much more concern than absolute accuracy.

Some architectures with good relative accuracy will have very small deviations in the outputs for closely-spaced inputs whereas others may have relatively large deviations in outputs for closely-spaced inputs.

DNL provides some measure of how outputs for closely-spaced inputs compare.