EE 505

Lecture 4

Data Converter Operation and Characterization

-- Linearity Metrics
Integral Nonlinearity (DAC)

Nonideal DAC

\[ \text{INL}_k = x_{\text{OUT}(k)} - x_{\text{OF}(k)} \]

\[ \text{INL} = \max_{0 \leq k \leq N-1} \{|\text{INL}_k|\} \]
Integral Nonlinearity (DAC)

Nonideal DAC

INL often expressed in LSB

$$\text{INL}_k = \frac{\chi_{\text{OUT}}(k) - \chi_{\text{OF}}(k)}{\chi_{\text{LSB}}}$$

$$\text{INL} = \max_{0 \leq k \leq N-1} \left\{ |\text{INL}_k| \right\}$$

- INL is often the most important parameter of a DAC
- $\text{INL}_0$ and $\text{INL}_{N-1}$ are 0 (by definition)
- There are $N-2$ elements in the set of $\text{INL}_k$ that are of concern
- INL is almost always nominally 0 (i.e. designers try to make it 0)
- INL is a random variable at the design stage
- $\text{INL}_k$ is a random variable for $0 < k < N-1$
- $\text{INL}_k$ and $\text{INL}_{k+j}$ are almost always correlated for all $k,j$ (not incl 0, N-1)
- Fit Line is a random variable
- INL is the $N-2$ order statistic of a set of $N-2$ correlated random variables
Integral Nonlinearity (DAC)

Nonideal DAC

- At design stage, INL characterized by standard deviation of the random variable
- Closed-form expressions for INL almost never exist because PDF of order statistics of correlated random variables is extremely complicated
- Simulation of INL very time consuming if n is very large (large sample size required to establish reasonable level of confidence)
  - Model parameters become random variables
  - Process parameters affect multiple model parameters causing model parameter correlation
  - Simulation times can become very large
- INL can be readily measured in laboratory but often dominates test costs because of number of measurements needed when n is large
- Expected of INL_k at k=(N-1)/2 is largest for many architectures
- Major effort in DAC design is in obtaining acceptable yield!
Integral Nonlinearity (ADC)

Consider end-point fit line with interpreted output axis

\[ x_{\text{INF}}(x_{\text{IN}}) = m x_{\text{IN}} + \left( \frac{x_{\text{LSB}}}{2} - m x_{T1} \right) \]

\[ m = \frac{(N-2)x_{\text{LSB}}}{x_{T7} - x_{T1}} \]
Integral Nonlinearity (ADC)

Nonideal ADC

Continuous-input based INL definition

\[
\text{INL}(x_{IN}) = \tilde{x}_{IN}(x_{IN}) - x_{INF}(x_{IN})
\]

\[
\text{INL} = \max_{0 \leq x_{IN} \leq x_{REF}} \{ \text{INL}(x_{IN}) \}
\]
Integral Nonlinearity (ADC)

Nonideal ADC

Break-point INL definition

INL \_k = X_{T_k} - X_{FT_k} \quad 1 \leq k \leq N-2

INL = \max_{2 \leq k \leq N-2} \{ \text{INL}_k \}
**Integral Nonlinearity (ADC)**

**Nonideal ADC**

Break-point INL definition

\[
\text{INL}_k = \frac{x_{Tk} - x_{FTk}}{x_{LSB}} \quad 1 \leq k \leq N-2
\]

\[
\text{INL} = \max_{2 \leq k \leq N-2} \{|\text{INL}_k|\}
\]

- INL is often the most important parameter of an ADC
- \(\text{INL}_1\) and \(\text{INL}_{N-1}\) are 0 (by definition)
- There are \(N-3\) elements in the set of \(\text{INL}_k\) that are of concern
- INL is a random variable at the design stage
- \(\text{INL}_k\) is a random variable for \(0 < k < N-1\)
- \(\text{INL}_k\) and \(\text{INL}_{k+j}\) are correlated for all \(k,j\) (not incl 0, N-1) for most architectures
- Fit Line (for cont INL) and uniformly spaced break pts (breakpoint INL) are random variables
- INL is the N-3 order statistic of a set of N-3 correlated random variables (breakpoint INL)
**Integral Nonlinearity (ADC)**

**Nonideal ADC**

Break-point INL definition

\[
\text{INL}_k = \frac{x_{Tk} - x_{FT1}}{x_{LSB}} \quad 1 \leq k \leq N-2
\]

\[
\text{INL} = \max_{2 \leq k \leq N-2} \{ |\text{INL}_k| \}
\]

• At design stage, INL characterized by standard deviation of the random variable
• Closed-form expressions for INL almost never exist because PDF of order statistics of correlated random variables is extremely complicated
• Simulation of INL very time consuming if n is very large (large sample size required to establish reasonable level of confidence)
  - Model parameters become random variables
  - Process parameters affect multiple model parameters causing model parameter correlation
  - Simulation times can become very large
Integral Nonlinearity (ADC)

Nonideal ADC

Break-point INL definition

\[
\text{INL}_k = \frac{x_{Tk} - x_{FTk}}{x_{LSB}} \quad 1 \leq k \leq N-2
\]

\[
\text{INL} = \max_{2 \leq k \leq N-2} \{|\text{INL}_k|\}
\]

- INL can be readily measured in laboratory but often dominates test costs because of number of measurements needed when \( n \) is large
- Expected of INL\(_k\) at \( k=(N-1)/2 \) is largest for many architectures
- Major effort in ADC design is in obtaining an acceptable yield
How important is statistical analysis?

Example: 7-bit FLASH ADC with R-string DAC

Case 1  \( \sigma_{VOS} = 5\text{mV} \)

\[ P_{COMP} = 0.565 \]

Since all comparators must be good, the ADC yield is

\[ Y_{ADC} = (P_{COMP})^{127} = (0.565)^{127} \]

\[ Y_{ADC} = 3.2 \cdot 10^{-32} \]

This yield is essentially 0 and a standard deviation of 5mV is even not trivial to obtain with MOS comparators!

The effects of statistical variation can have dramatic effects on yield of data converters!
How important is statistical analysis?

Statistical analysis of data converters is critical

Some architectures are more sensitive than others to statistical variations in components

The onset of yield loss due to statistical limitations is generally quite abrupt and can have disastrous effects if not considered as part of the design process

Substantially over-designing to avoid concerns about statistical yield loss is not a practical solution since the area penalty, the speed penalty, and the power penalty are generally quite severe

For the effects of local random variations of a parameter $X$, generally

$$\sigma_X \sim \frac{1}{\sqrt{A_C}}$$

where $A_C$ is the area of the matching critical components
INL-based ENOB

Consider initially the continuous INL definition for an ADC where the INL of an ideal ADC is $X_{\text{LSB}}/2$

Assume

$$\text{INL} = \theta X_{\text{REF}} = \nu X_{\text{LSBR}}$$

where $X_{\text{LSBR}}$ is the LSB based upon the defined resolution

Define the LSB by

$$X_{\text{LSB}} = \frac{X_{\text{REF}}}{2^{n_{\text{EQ}}}}$$

Thus

$$\text{INL} = \theta 2^{n_{\text{EQ}}} X_{\text{LSB}}$$

Since an ideal ADC has an INL of $X_{\text{LSB}}/2$, express INL in terms of ideal ADC

$$\text{INL} = \left[ \theta 2^{(n_{\text{EQ}}+1)} \right] \left( \frac{X_{\text{LSB}}}{2} \right)$$

Setting term in [ ] to 1, can solve for $n_{\text{EQ}}$ to obtain

$$\text{ENOB} = n_{\text{EQ}} = \log_2 \left( \frac{1}{2\theta} \right) = n_{\text{R}} - 1 - \log_2 (\nu)$$

where $n_{\text{R}}$ is the defined resolution
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.

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

where \( n_R \) is the specified resolution and \( \nu \) is the resolution in LSB at the \( n_R \) bit level.
A DAC with $n_{\text{EFF}}$ bits (ENOB) of resolution should have all outputs bounded by $\pm X_{\text{LSB}}/2$ from the fit line so distance between fit line and upper/lower bounding lines determines the 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

\[ \text{INL}_A = \frac{\nu_A X_{\text{LSB}}}{X_{\text{REF}}} \]  \hspace{1cm} (1)

But since it is assumed to have \( n_{RA} \) bits of resolution

\[ \frac{X_{\text{LSB}}}{X_{\text{REF}}} = 2^{-n_{RA}} \]  \hspace{1cm} (2)

Thus we obtain the expression

\[ \text{INL}_A = \nu_A 2^{-n_{RA}} \]  \hspace{1cm} (3)
INL-based ENOB

Proof (cont) \[ \text{INL}_A = \nu_A 2^{-n_{RA}} \]

and the ENOB is given by

\[ \text{ENOB}_A = n_{RA} - 1 - \log_2(\nu_A) \quad (4) \]

By a similar argument we obtain

\[ \text{INL}_B = \nu_B 2^{-n_{RB}} \quad (5) \]

\[ \text{ENOB}_B = n_{RB} - 1 - \log_2(\nu_B) \quad (6) \]

Since there are simply two representations of the same nonlinearity, the absolute INL will be the same for both representations. That is, \[ \text{INL}_A = \text{INL}_B \quad (7) \]

Substituting from (3) into (4) we obtain

\[ \text{ENOB}_A = n_{RA} - 1 - (n_{RA} + \log_2(\text{INL}_A)) \quad (8) \]
INL-based ENOB

Proof (cont)  

\[ \text{ENOBA} = n_{RA} - 1 - (n_{RA} + \log_2 (\text{INLA})) \]  \hspace{1cm} (8)

\[ \text{ENOBA} = -1 - \log_2 (\text{INLA}) \]  \hspace{1cm} (9)

from (7) this can be expressed as

\[ \text{ENOBA} = -1 - \log_2 (\text{INLB}) \]  \hspace{1cm} (10)

\[ \text{ENOBA} = n_{RB} - 1 - (n_{RB} + \log_2 (\text{INLB})) \]  \hspace{1cm} (11)

Substituting from (5) into (6) we obtain

\[ \text{ENOBB} = n_{RB} - 1 - (n_{RB} + \log_2 (\text{INLB})) \]  \hspace{1cm} (12)

Finally, from (11) and (12) we thus conclude that

\[ \text{ENOBA} = \text{ENOBB} \]
INL-based ENOB

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

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

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

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

\[
\nu = \frac{\theta \cdot X_{\text{REF}}}{X_{\text{LSB}}}
\]
Proof (cont):

\[ \nu = \frac{\theta \cdot X_{\text{REF}}}{X_{\text{LSB}}} \]

but

\[ X_{\text{LSB}} = \frac{X_{\text{REF}}}{2^{n_R}} \]

Substituting into the previous equation we obtain

\[ \theta = \nu 2^{-n_R} \]

thus

\[ -\log_2 (\text{INL}_{\text{REF}}) - 1 = -\log_2 (\theta) - 1 = -\log_2 (\nu 2^{-n_R}) - 1 \]

\[ = n_R - \log_2 (\nu) - 1 \]

\[ = \text{ENOB} \]
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

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

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

DNL(k) is the actual increment from code (k-1) to code k minus the ideal increment normalized to \( x_{\text{LSB}} \).
Differential Nonlinearity (DAC)

Nonideal DAC

Increment at code $k$ is a signed quantity and will be negative if $X_{\text{OUT}(k)} < X_{\text{OUT}(k-1)}$

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

$$DNL = \max_{1 \leq k \leq N-1} \{|DNL(k)|\}$$

$DNL = 0$ for an ideal DAC
Monotonicity (DAC)

A DAC is monotone if $X_{OUT}(k) > X_{OUT}(k-1)$ for all $k$

Theorem:
A DAC is monotone if $DNL(k) > -1$ for all $k$
Differential Nonlinearity (DAC)

Nonideal DAC

Theorem: The $\text{INL}_k$ 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: $\text{DNL}(k)=\text{INL}_k-\text{INL}_{k-1}$
Theorem: If the INL of a DAC satisfies the relationship

\[ \text{INL} < \frac{1}{2} X_{\text{LSB}} \]

then the DAC is monotone

Note: This is a necessary but not sufficient condition for monotonicity
Differential Nonlinearity (ADC)

Nonideal ADC

DNL(k) is the code width for code k – ideal code width normalized to $X_{\text{LSB}}$

$$DNL(k) = \frac{X_{T(k+1)} - X_{Tk} - X_{\text{LSB}}}{X_{\text{LSB}}}$$
Differential Nonlinearity (ADC)

Nonideal ADC

\[ DNL(k) = \frac{x_{T(k+1)} - x_{Tk} - x_{LSB}}{x_{LSB}} \]

\[ DNL = \max_{2 \leq k \leq N-1} \{ |DNL(k)| \} \]

DNL=0 for an ideal ADC

Note: In some nonideal ADCs, two or more break points could cause transitions to the same code \( C_k \) making the definition of DNL ambiguous.
Monotonicity in an ADC

Definition: An ADC is monotone if the

\[ \bar{X}_{\text{OUT}}(\mathcal{X}_k) \geq \bar{X}_{\text{OUT}}(\mathcal{X}_m) \quad \text{whenever} \quad \mathcal{X}_k \geq \mathcal{X}_m \]

Note: Have used \( \mathcal{X}_{Bk} \) instead of \( \mathcal{X}_{Tk} \) 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”

Note: Some authors claim that missing codes in an ADC are the counterpart to nonmonotonicity in a DAC. This association is questionable.
Missing Codes (ADC)

Nonideal ADCs

Missing codes

Missing code with all codes present
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
LSB Definition

$X_{\text{LSB}}$ appears in many performance specifications but the definition of $X_{\text{LSB}}$ is generally not given. This can cause modest inconsistencies in the definition of some performance specifications.

What is $X_{\text{LSB}}$?
LSB Definition

$X_{\text{LSB}}$ appears in many performance specifications but the definition of $X_{\text{LSB}}$ is generally not given. This can cause modest inconsistencies in the definition of some performance specifications.

What is $X_{\text{LSB}}$?

Conventional Wisdom $X_{\text{LSB}}$

$$X_{\text{LSB}} = \frac{X_{\text{REF}}}{2^{n_R}}$$

($X_{\text{LSB}}$ determined by specified resolution and can not be measured)
LSB Definition

\( X_{\text{LSB}} \) appears in many performance specifications but the definition of \( X_{\text{LSB}} \) is generally not given. This can cause modest inconsistencies in the definition of some performance specifications.

Alternate definitions of \( X_{\text{LSB}} \)

\[ X_{\text{LSB}} = \frac{\text{REF}}{N} \]

where \( N \) is the measured number of DAC output levels

\[ X_{\text{LSB}} = \frac{X_{0(N-1)} - X_{0(0)}}{N-1} \]

where \( N \) is the measured number of DAC output levels and \( X_{0(N-1)} \) and \( X_{0(0)} \) are last and first outputs respectively

useful when extreme values do not occur at minimum and maximum input codes

\[ X_{\text{LSB}} = \frac{\max\{X_{0(k)}\} - \min\{X_{0(k)}\}}{N-1} \]

useful for determining worst-case resolution of a DAC

\[ X_{\text{LSB}} = \max_{k}\{X_{0(k)} - X_{0(k-1)}\} \]

Similar definitions can be made for \( X_{\text{LSB}} \) of an ADC based upon the breakpoints
End of Lecture 4
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:

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

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

\[
INL_i = \begin{cases} 
0 & i=0, N-2 \\
\frac{\sum_{k=1}^{i} \hat{C}_k}{\bar{C}} - i\bar{C} & 1 \leq i \leq N-3
\end{cases}
\]

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

\[
INL = \max_{1 \leq i \leq N-3} \{|DNL_i|\}
\]