EE 505

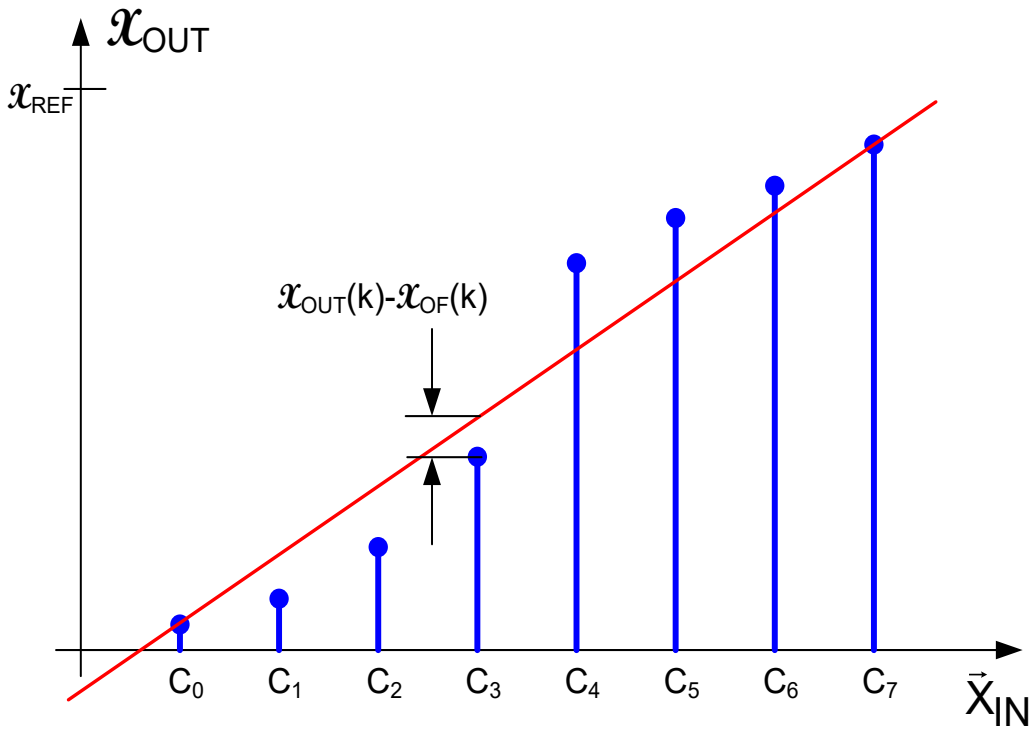
Lecture 3

Data Converter Operation and Characterization

-- Linearity Metrics

Integral Nonlinearity (DAC)

Nonideal DAC

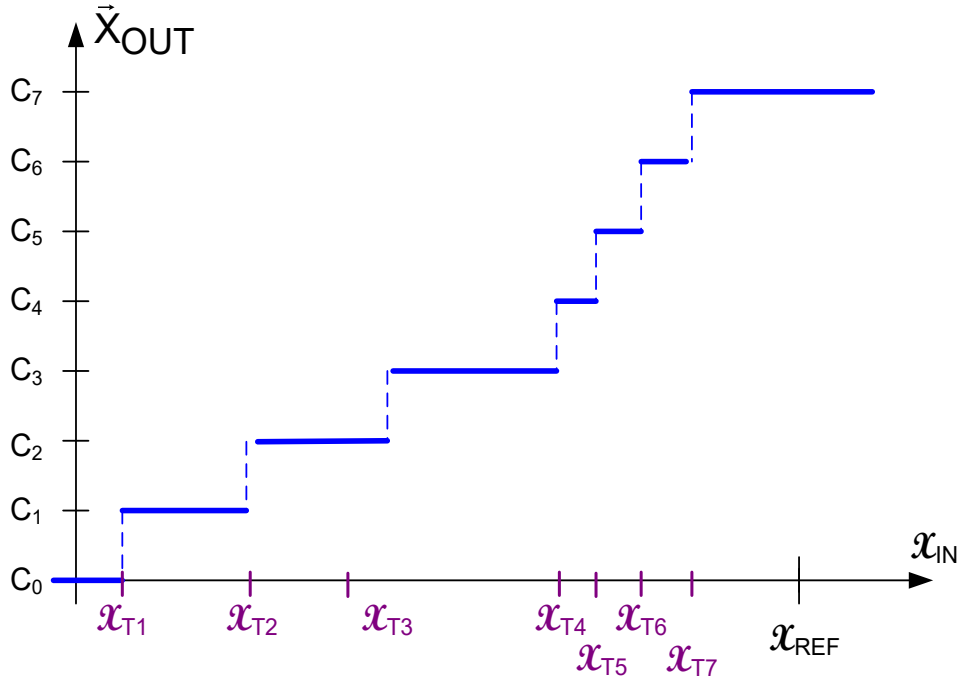


$$INL_k = x_{OUT}(k) - x_{OF}(k)$$

$$INL = \max_{0 \le k \le N-1} \{|INL_k|\}$$

Integral Nonlinearity (ADC)

Nonideal ADC



x_{Tk} is the transition input to code C_k

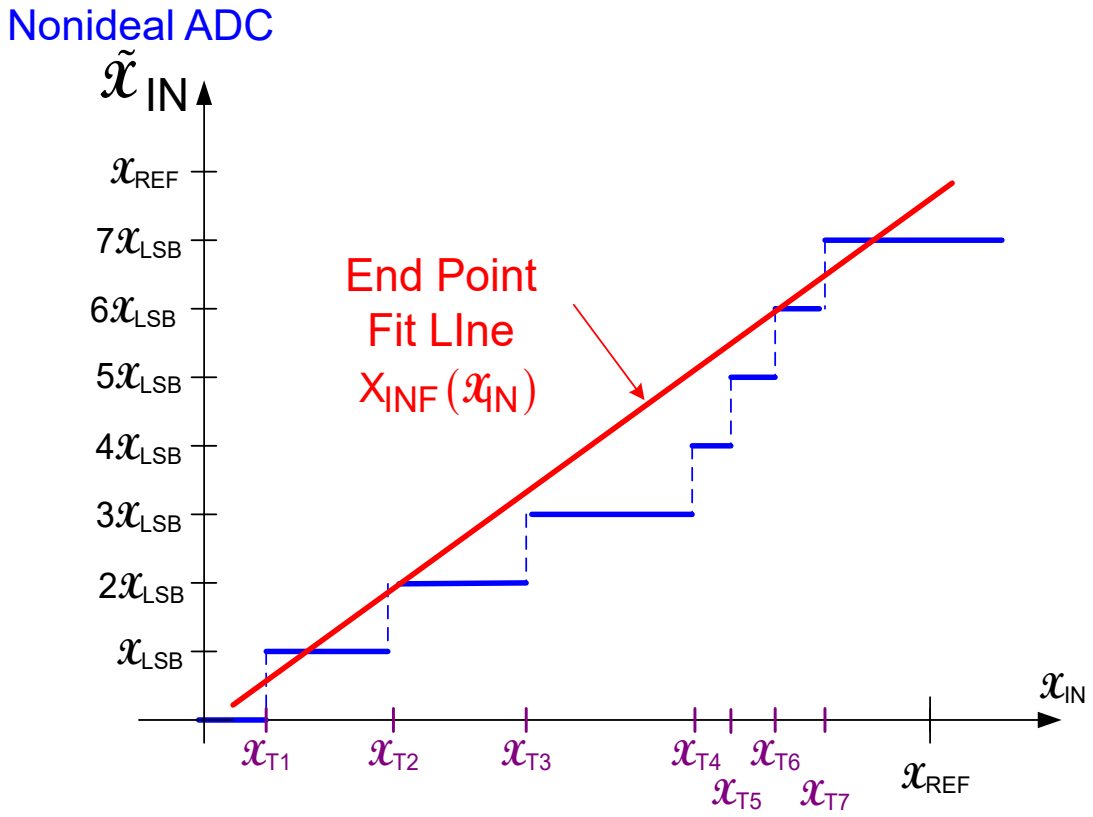
Transition points are not uniformly spaced !

More than one definition for INL exists !

Will give two definitions here (second almost always used)

Note: is some cases the sequence < x_{Tk} > may not be monotone

Integral Nonlinearity (ADC)



Consider end-point fit line with interpreted output axis

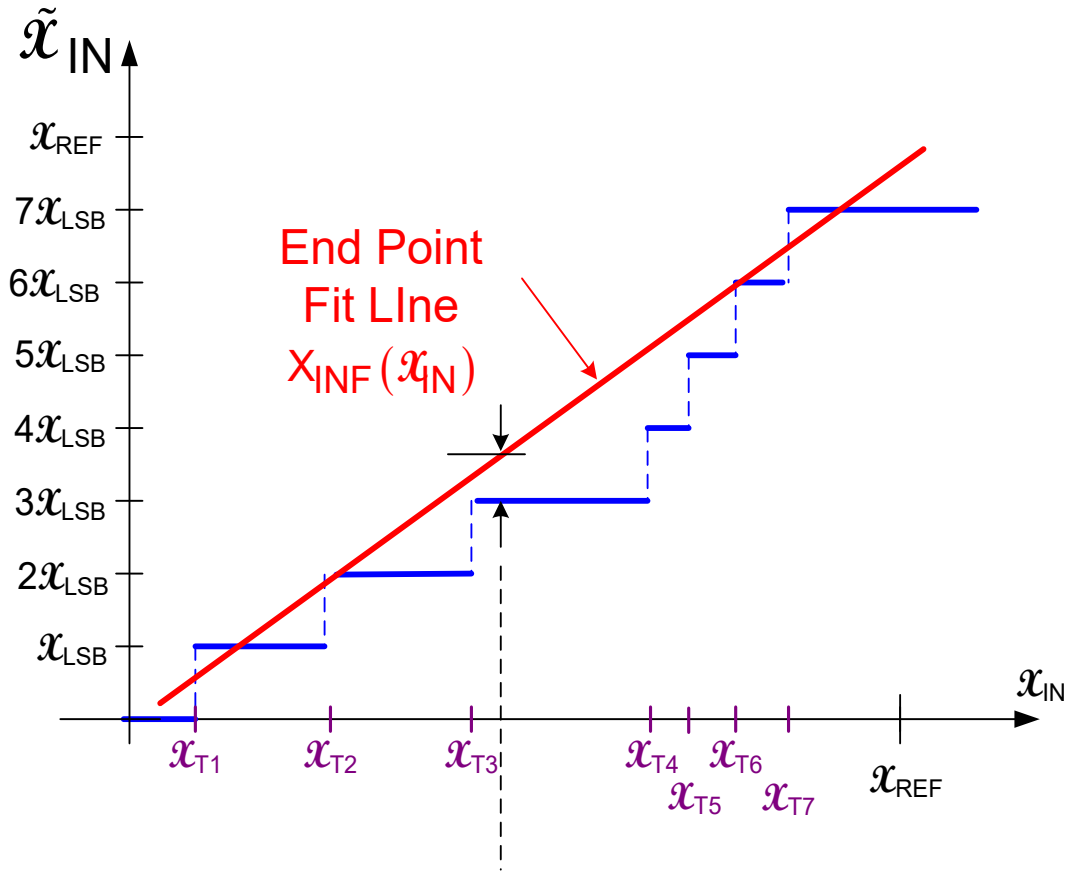
$$X_{INF}(x_{IN}) = m x_{IN} + \left(\frac{x_{LSB}}{2} - m x_{T1} \right)$$

$$m = \frac{(N-2)\, x_{LSB}}{x_{T7} - x_{T1}}$$

Integral Nonlinearity (ADC)

Nonideal ADC

Continuous-input based INL definition



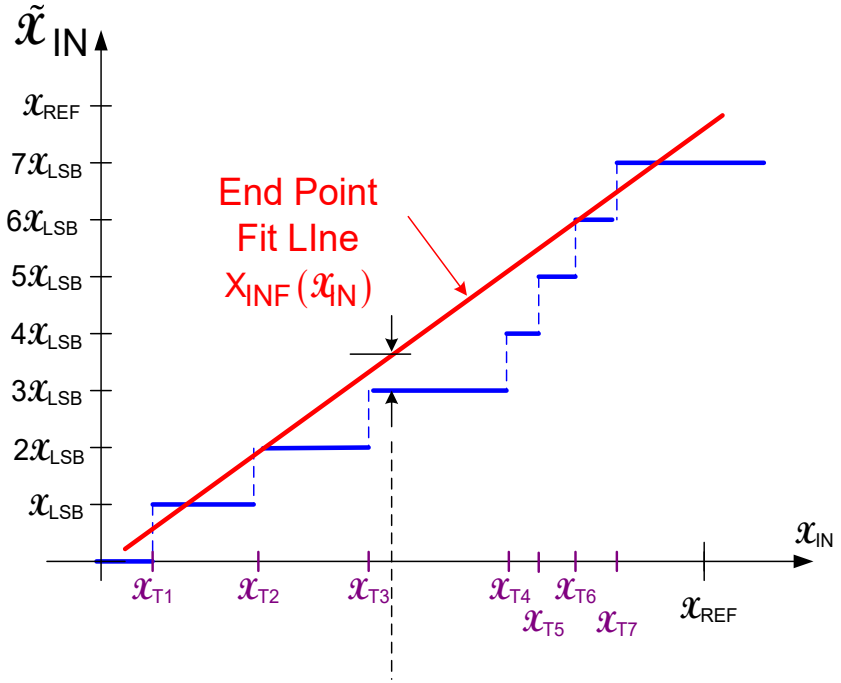
$$INL(x_{IN}) = \tilde{x}_{IN}(x_{IN}) - X_{INF}(x_{IN})$$

$$INL = \max_{0 \le x_{IN} \le x_{REF}} \left\{ \left| INL(x_{IN}) \right| \right\}$$

Integral Nonlinearity (ADC)

Nonideal ADC

Continuous-input based INL definition



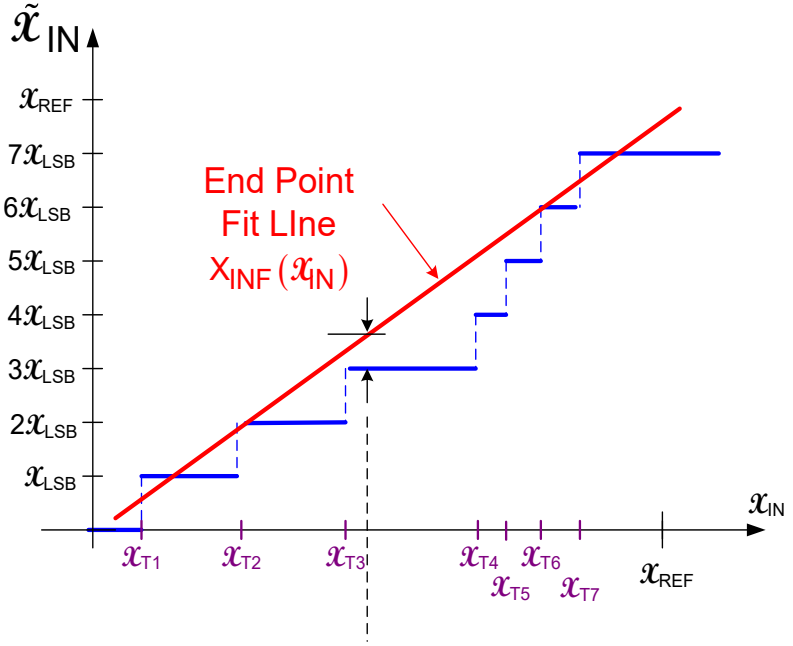
Often expressed in LSB

$$INL(x_{IN}) = \frac{\tilde{x}_{IN}(x_{IN}) - X_{INF}(x_{IN})}{x_{LSB}}$$

$$INL = \max_{0 \le x_{IN} \le x_{REF}} \left\{ \left| INL(x_{IN}) \right| \right\}$$

Integral Nonlinearity (ADC)

Nonideal ADC



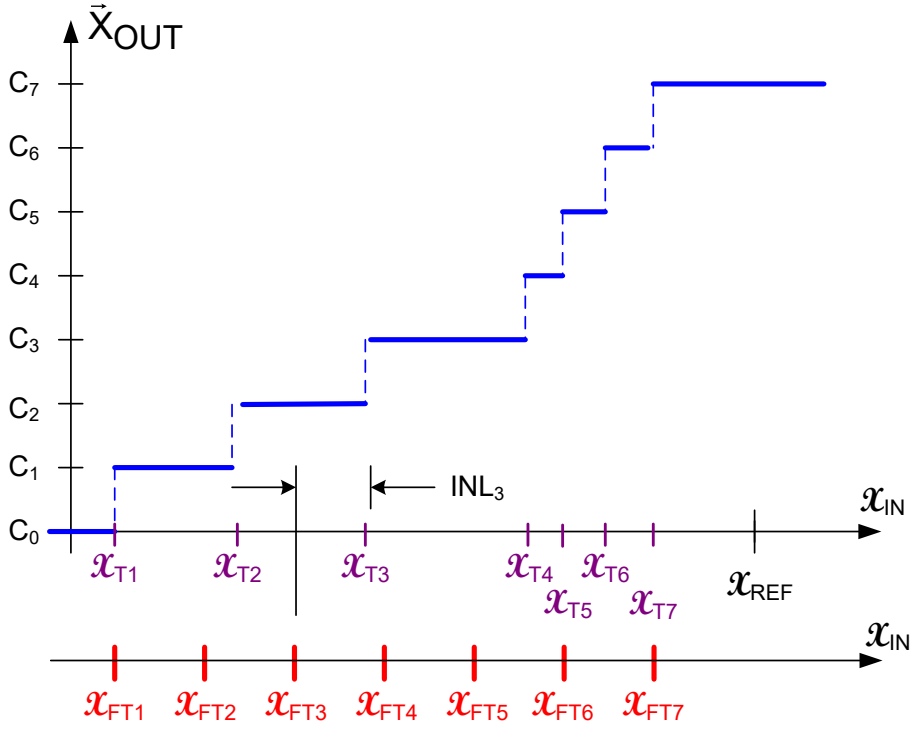
With this definition of INL, the INL of an ideal ADC is $x_{LSB}/2$ (for $x_{T1}=x_{LSB}$)

This is effective at characterizing the overall nonlinearity of the ADC but does not vanish when the ADC is ideal and the effects of the breakpoints is not explicit

Integral Nonlinearity (ADC)

Nonideal ADC

Break-point INL definition (assuming N-3 internal transitions)



Place N-3 uniformly spaced points between X_{T1} and $X_{T(N-1)}$ designated x_{FTk}

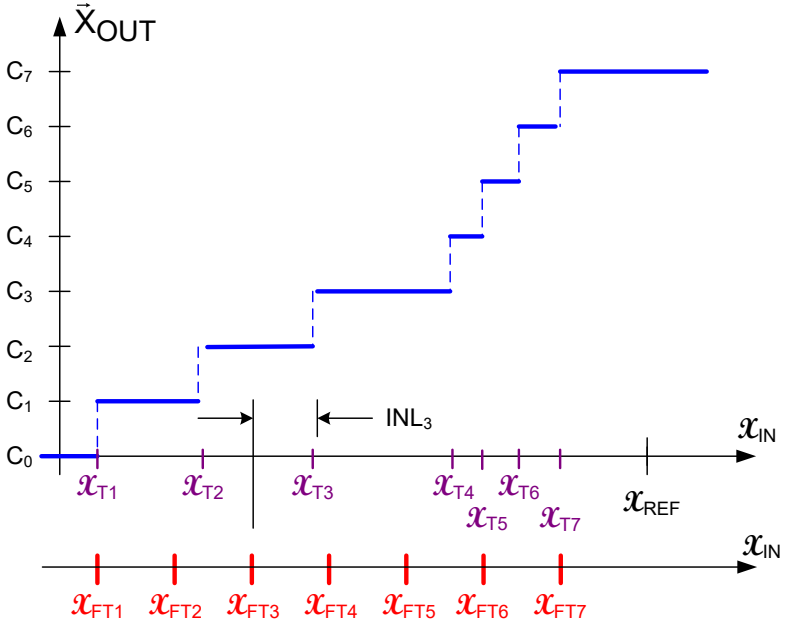
$$INL_k = x_{Tk} - x_{FTk} \qquad 1 \le k \le N\text{-}2$$

$$INL = \max_{2 \le k \le N-2} \{|INL_k|\}$$

Integral Nonlinearity (ADC)

Nonideal ADC

Break-point INL definition (assuming N-3 internal transitions)



Often expressed in LSB

$$INL_k = \frac{x_{Tk} - x_{FTk}}{x_{LSB}} \qquad 1 \le k \le N\text{-}2$$

$$INL = \max_{2 \le k \le N-2} \{|INL_k|\}$$

For an ideal ADC, INL is ideally 0

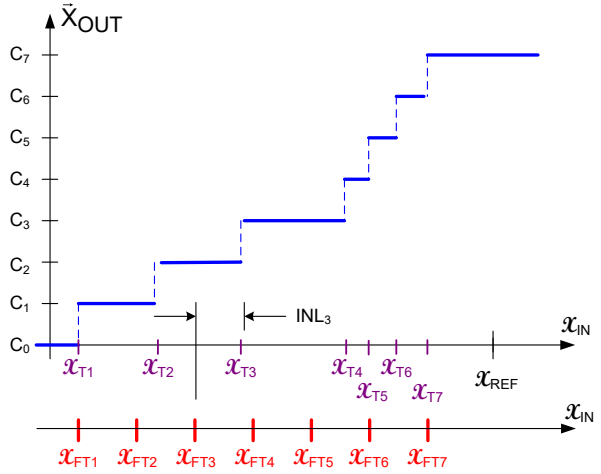
Integral Nonlinearity (ADC)

Nonideal ADC

Break-point INL definition (assuming N-3 internal transitions)

$$INL_k = \frac{x_{Tk} - x_{FTk}}{x_{LSB}} \qquad 1 \le k \le N\text{-}2$$

$$INL = \max_{2 \le k \le N-2} \{|INL_k|\}$$



- INL is often the most important parameter of an ADC
- INL_1 and INL_{N-1} are 0 (by definition)
- There are N-3 elements in the set of INL_k that are of concern
- INL is a random variable at the design stage
- INL_k is a random variable for 0<k<N-1
- INL_k and 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)
- **INL is a parameter that is attempting to characterize the linearity of an ADC !**

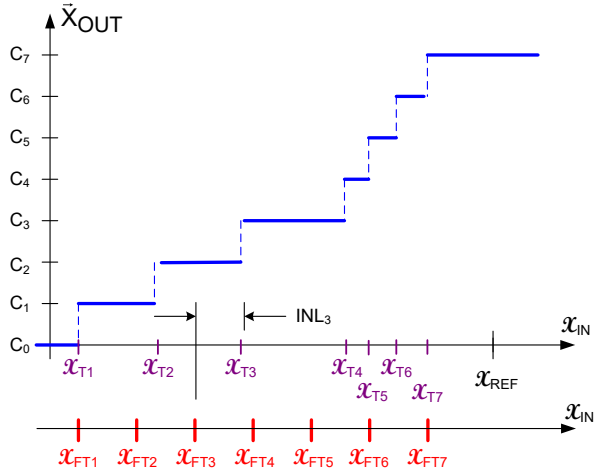
Integral Nonlinearity (ADC)

Nonideal ADC

Break-point INL definition (assuming N-3 internal transitions)

$$INL_k = \frac{x_{Tk} - x_{FTI}}{x_{LSB}} \qquad 1 \le k \le N\text{-}2$$

$$INL = \max_{2 \le k \le N-2} \{|INL_k|\}$$



What if there are less than N-3 internal transitions?

- Assume N-k internal transitions where k>3
- Data converter may still perform quite well !
- Insert N-k uniformly spaced values and use previous definition
- Unusual issues can crop up when testing data converters and it is important to have well-defined algorithms for handling these situations

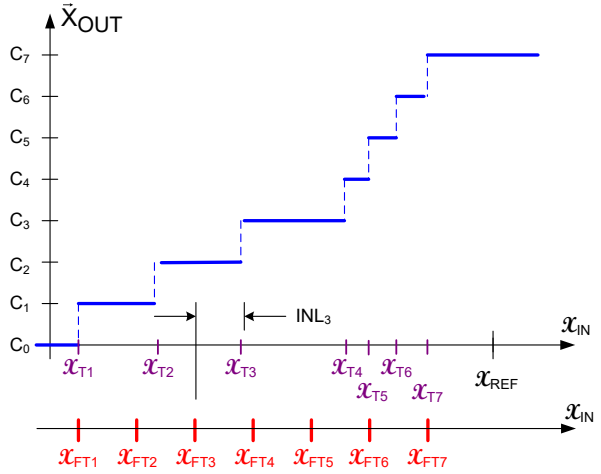
Integral Nonlinearity (ADC)

Nonideal ADC

Break-point INL definition

$$INL_k = \frac{x_{Tk} - x_{FTk}}{x_{LSB}} \qquad 1 \le k \le N-2$$

$$INL = \max_{2 \le k \le N-2} \{|INL_k|\}$$



- At design stage, INL characterized by standard deviation of many random variables
- 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

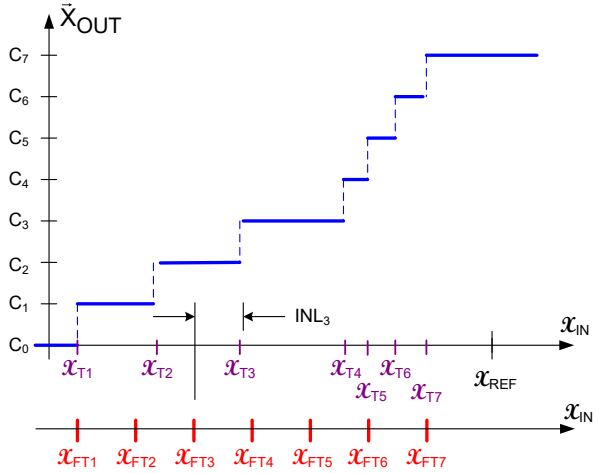
Integral Nonlinearity (ADC)

Nonideal ADC

Break-point INL definition

$$INL_k = \frac{x_{Tk} - x_{FTk}}{x_{LSB}} \qquad 1 \le k \le N-2$$

$$INL = \max_{2 \le k \le N-2} \{|INL_k|\}$$

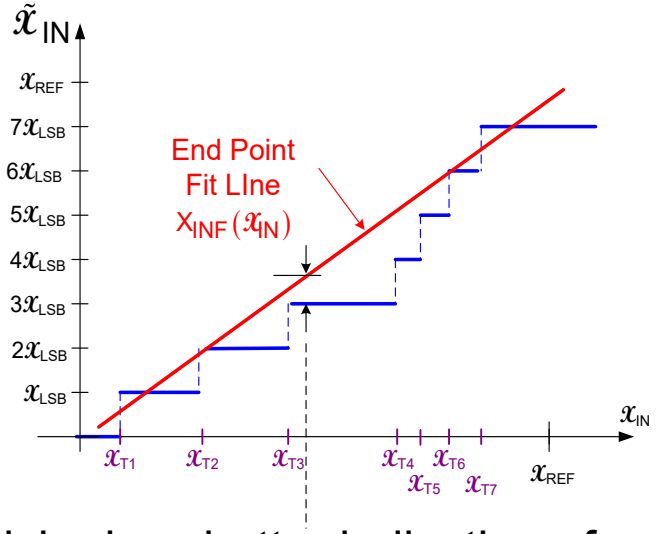


- INL (with some definition) can be readily measured in laboratory but often dominates test costs because of number of measurements needed when n is large
- Expected value of INL_k at k=(N-1)/2 is largest for many architectures
- INL of $\frac{x_{LSB}}{2}$ often considered acceptable (this is the ideal value of the continuous-input INL)

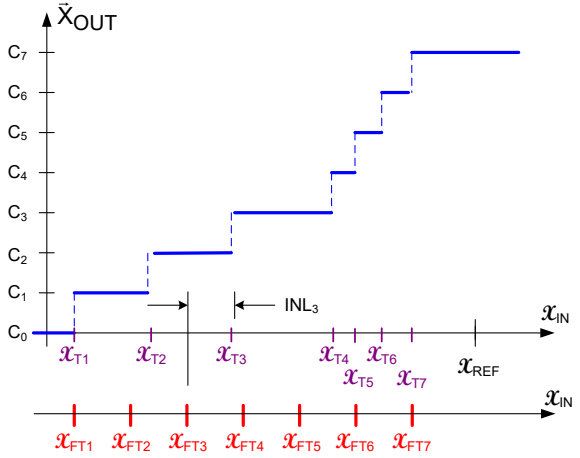
- Major effort in ADC design is invariably dominated by obtaining an acceptable INL yield !
- Yield often strongly dependent upon matching of random variables !

Integral Nonlinearity (ADC)

Continuous-input based INL definition



Break-point INL definition



Which gives better indication of overall system performance as affected by the ADC?

Continuous-input based INL

Which is less affected by quirks in an actual ADC such as missing break pointe or non-monotone break point sequencing?

Continuous-input based INL

Why is Break-point INL Definition Almost Always Specified?

Actual break points seldom determined in testing

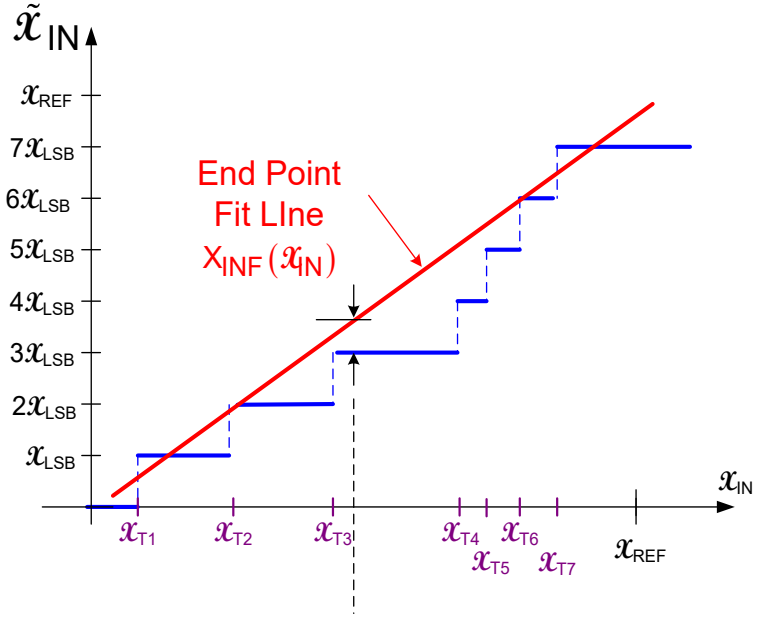
Which is actually specified in datasheets ?

Neither !!

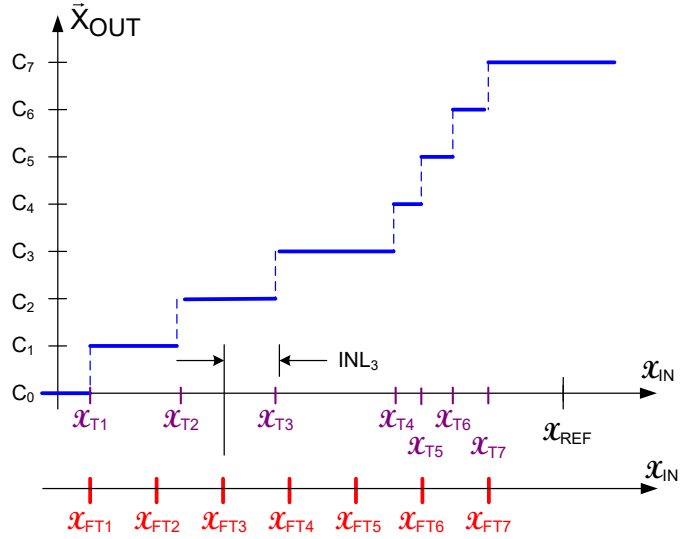
A third “definition” is used, and it is the “Code-Density INL”

Integral Nonlinearity (ADC)

Continuous-input based INL definition



Break-point INL definition



Code Density INL

Based upon number of times each code occurs usually based upon assumption that highly linear test signal is applied

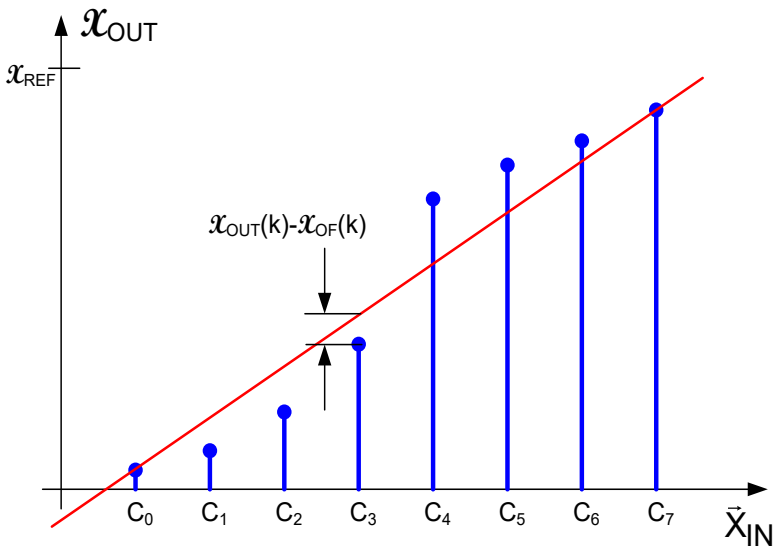
Significant reduction in test time with code density INL !

Can be thought of as a practical approximation of the Break-Point INL

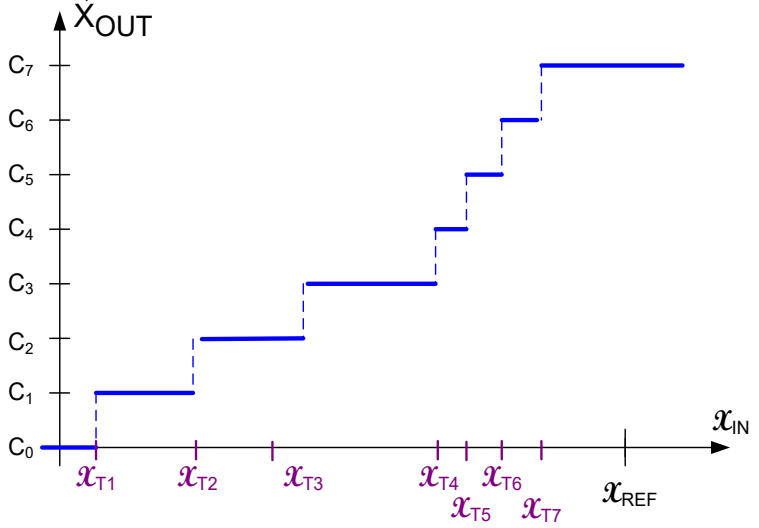
Difficult to assess system performance from Code Density INL if quirks exist in the ADC

Dominant Data Converter Static Nonlinearities Almost Always Determined by Inherent Missmatch in Components

Actual DAC Nonidealities

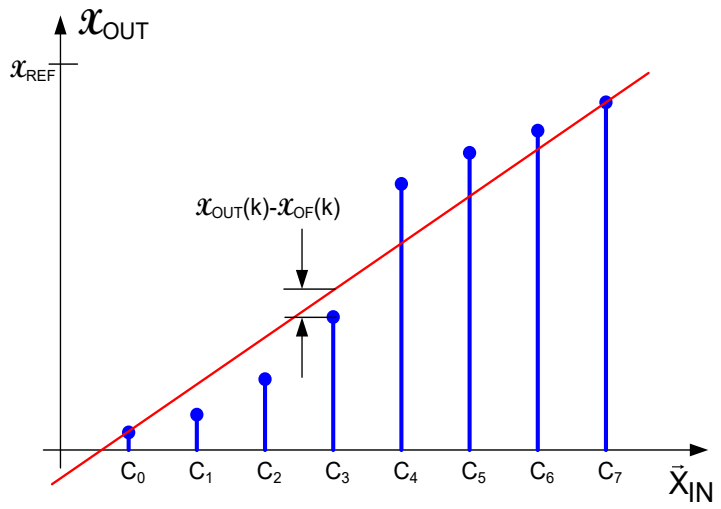


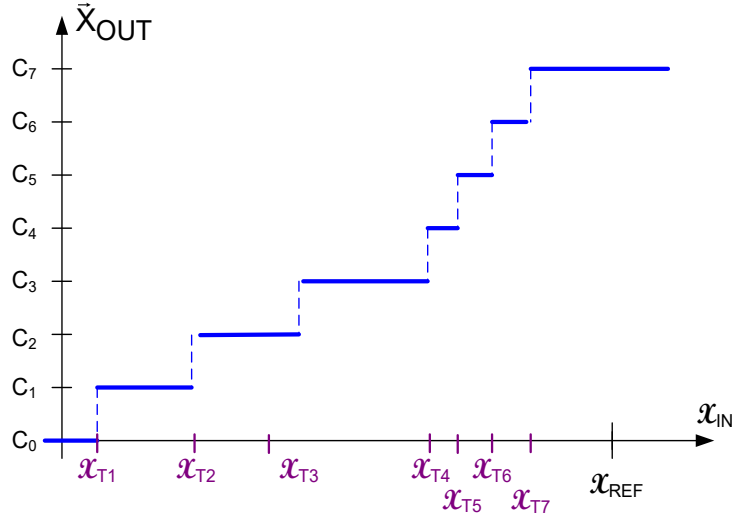
Actual ADC Nonidealities



- Mismatch a major contributor to yield degradation during volume production
- Effects of mismatch on yield requires a statistical analysis or statistical assessment

How important is statistical analysis or assessment to the successful design of data converters?





Characteristics Dominantly Depend Upon Random Variables

- 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

Characteristics Dominantly Depend Upon Random Variables

- Dynamic characteristics
  - Conversion Time or Conversion Rate (ADC)
  - Settling time or Clock Rate (DAC)
  - Sampling Time Uncertainty (aperture uncertainty or aperture jitter)
  - Dynamic Range
  - Spurious Free Dynamic Range (SFDR)
  - Total Harmonic Distortion (THD)
  - Signal to Noise Ratio (SNR)
  - Signal to Noise and Distortion Ratio (SNDR)
  - Sparkle Characteristics
  - Effective Number of Bits (ENOB)

Methods of Characterizing how Random Variables Affect Performance

- Analytical Statistical Formulation and Analysis
- MATLAB Simulations (often using Monte-Carlo Analysis)
- Spectre/Spice Monte-Carlo Simulations
- Ignore Effects of Random Effects

How important is statistical characterization of data converters?

How important is statistical analysis?

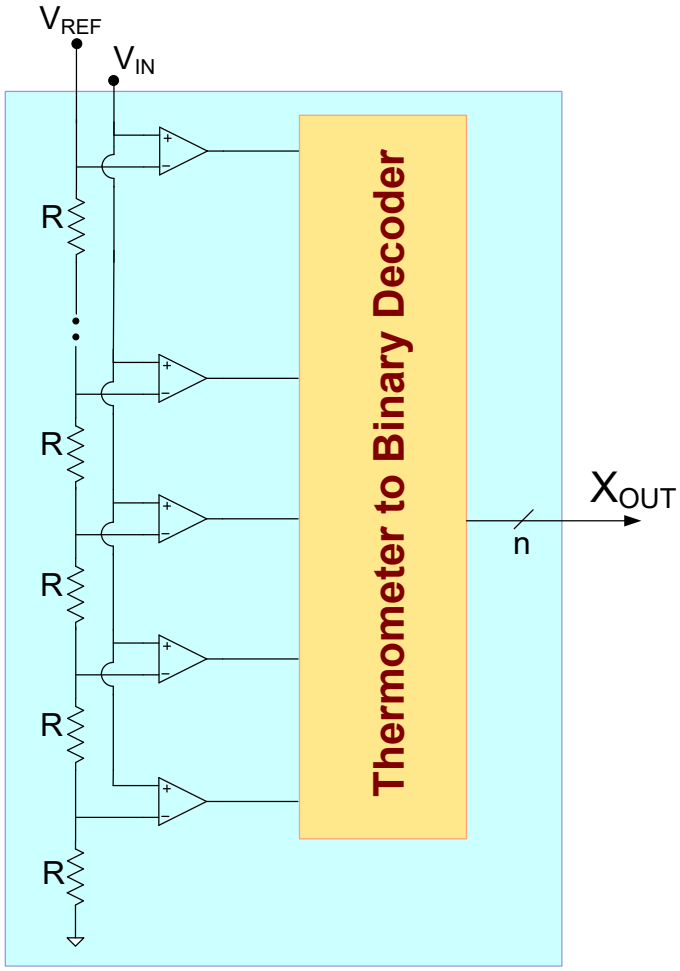
Example: 7-bit FLASH ADC with R-string DAC

Assume R-string is ideal, V_{REF}=1V and V_{OS} for each comparator must be at most +/- ½ LSB

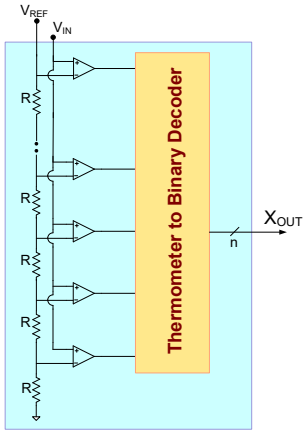
Why this assumption?

Case 1

Determine the yield if V_{OS} has a Gaussian distribution (Normal) with zero mean and a standard deviation of 5mV



How important is statistical analysis?



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

Assume R-string is ideal, V_{REF}=1V and V_{OS} for each comparator must be at most +/- ½ LSB

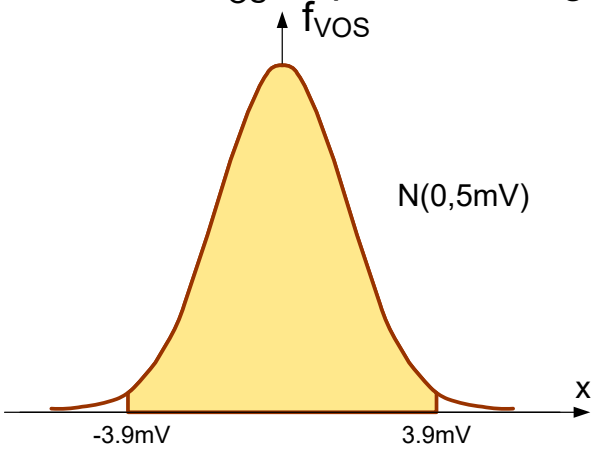
Case 1

Determine the yield if V_{OS} has a Gaussian distribution (Normal) with zero mean and a standard deviation of 5mV

½ LSB = $1V/(2^{(7+1)})=3.9mV$

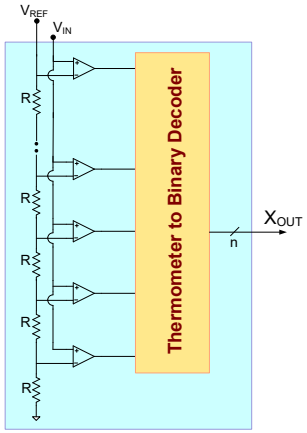
The probability that a single comparator meets the V_{OS} requirement is given by

$$P_{COMP} = \int_{-3.9mV}^{3.9mV} f_{VOS} dV$$



How important is statistical analysis?

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



Case 1 σ_{VOS}=5mV

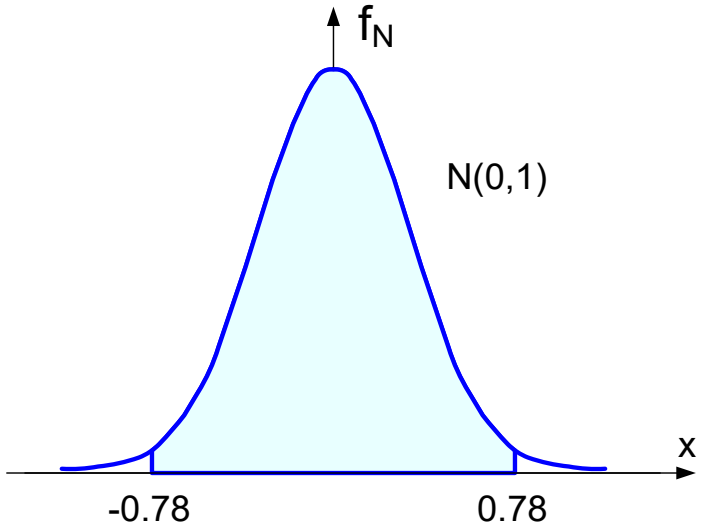
$$P_{COMP} = \int_{-3.9mV}^{3.9mV} f_{VOS} dV$$

Define $X_N = V_{OS}/\sigma$ $\quad X_N \sim N(0,1)$

$$P_{COMP} = \int_{-X_N}^{X_N} f_N dx$$

X_N=3.9mV/5mV=0.78

$$P_{COMP} = \int_{-0.78}^{0.78} f_N dx$$

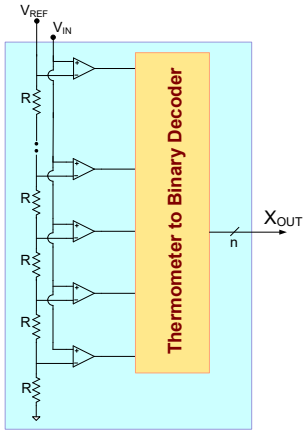


$P_{COMP} = 2 \bullet F_N(0.78) - 1 = 2 \bullet .7823 - 1 = 0.565$

Each comparator has 56.5% yield

How important is statistical analysis?

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



Case 1 $\sigma_{VOS}=5mV$

$P_{COMP} = 0.565$

Since all comparators must be good, the ADC yield is

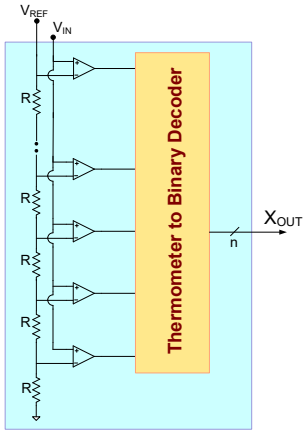
$$Y_{ADC}=(P_{COMP})^{127}=(0.565)^{127}$$

$$Y_{ADC}=3.2\bullet 10^{-32}$$

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?



Bubbles:

An ideal ADC has a monotone relationship between analog inputs and digital outpu

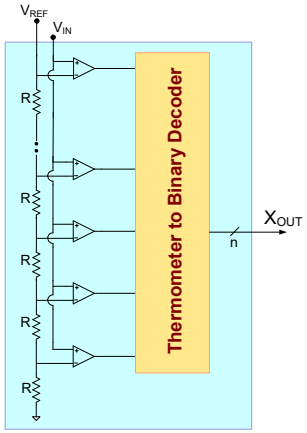
If expressed in thermometer code, the “thermometer” output never decreases with increasing inputs

Occasionally, some ADCs will exhibit one or more outputs where the Boolean output drops rather than increases

This flash structure naturally provides a thermometer code output. If the thermometer code output is not monotone, we say it has a bubble

Major errors can occur in the TBD if a bubble exists in the thermometer code outpu

How important is statistical analysis?



Sparkle:

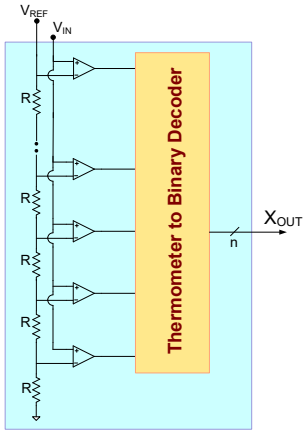
Occasionally, some ADCs will have one or more outputs that appear to be almost unrelated to the input (very large error in output, maybe only for small range of outputs or maybe only occur once in a while)

If this happens, the data converter is said to have “Sparkle”

This ADC architecture is vulnerable to sparkle if bubbles exist in the thermometer output code and the TBD, a logic circuit, does not appropriately handle them

How important is statistical analysis?

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



Case 1 σ_{VOS}=5mV

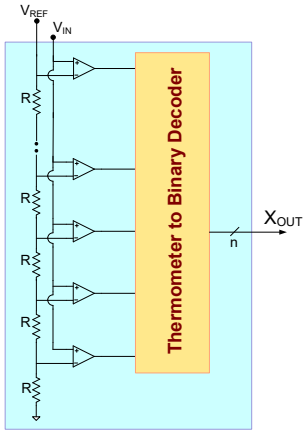
Since all comparators must be good, the ADC yield is

$$Y_{ADC}=3.2\bullet10^{-32}$$

Note: The specification in this example that requires no comparator has an offset voltage of larger than 0.5LSB may not be a good performance specification as the FLASH ADC may actually perform reasonably well even if some comparators have an offset that is larger than 0.5LSB. A more useful requirement might be that there be no bubbles in the thermometer code output. Certainly if all comparators have an offset that is at most 0.5LSB, there will be no bubbles in the output code attributable to comparator offset but a modestly weaker constraint can also guarantee there are no bubbles. With the 0.5LSB assumption, a specification that was dependent upon 127 uncorrelated random variables was obtained which made the analysis quite easy. A “no bubble” specification could be approximated by stating that the maximum of the 127

How important is statistical analysis?

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



Case 2 Repeat the previous example if σ_{VOS}=1mV

$$P_{COMP} = \int_{-3.9mV}^{3.9mV} f_{VOS}\,dV$$

$\rightarrow$ X_N=3.9mV/1mV=3.9

$$P_{COMP} = \int_{-3.9}^{3.9} f_N\,dx \qquad P_{COMP} = 2 \bullet F_N(3.9) - 1 = 2 \bullet 0.999952 - 1 = 0.999904$$

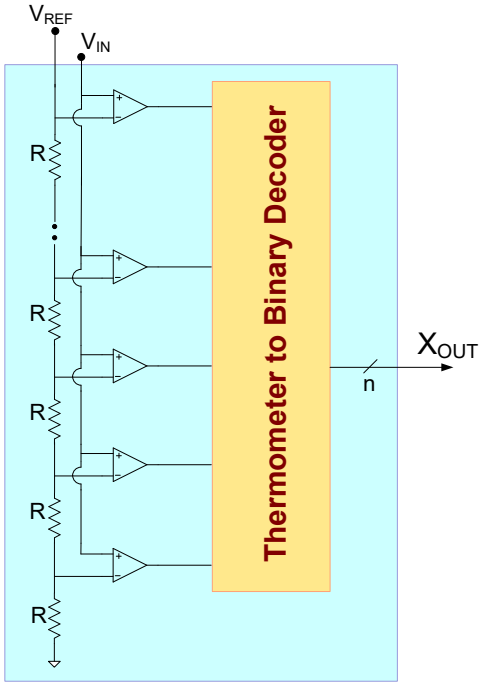
$$Y_{ADC} = (P_{COMP})^{127} = (0.999904)^{127}$$

$$Y_{ADC} = 0.988$$

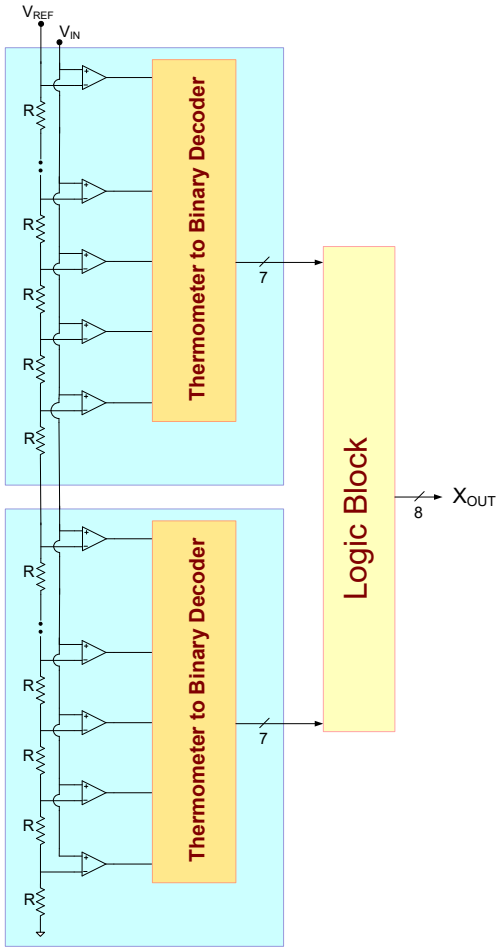
This modest change in the offset voltage has increased the yield to 98.8%

How important is statistical analysis?

Example: What will be the yield if two of the 7-bit FLASH ADCs with yields of 98.8% are combined to obtain an 8-bit ADC?



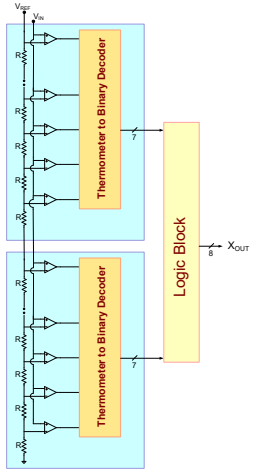
Y_{ADC}=98.8%



Y_{ADC}=?

How important is statistical analysis?

Example: What will be the yield if two of the 7-bit FLASH ADCs with yields of 98.8% are combined to obtain an 8-bit ADC?



Since one additional bit has been added, V_{LSB} will decrease From 7.8mV to 3.9mV. Thus ½ LSB will be reduced to 1.95mV

$$P_{COMP} = \int_{-1.95mV}^{1.95mV} f_{VOS}\,dV$$

With the same σ_{VOS}=1mV, X_N=1.95mV/1mV=1.95

$$P_{COMP} = \int_{-1.95}^{1.95} f_N\,dx \qquad P_{COMP} = 2 \bullet F_N(1.95) - 1 = 2 \bullet 0.97441 - 1 = 0.9488$$

$$Y_{ADC} = (P_{COMP})^{255} = (0.9488)^{255}$$

$$Y_{ADC} = 1.52 \bullet 10^{-6}$$

This seemingly simple extension of a circuit with a very high yield has essentially no yield !

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

  Recall examples where σ_{VOS}=5mV compared with σ_{VOS}=1mV

- 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

$$\sigma_X \propto \frac{A_0}{\sqrt{A_C}}$$

where A_C is the area of the matching critical components and A_0 is a process parameter

Importance of statistical analysis – example

What changes in area would be needed to decrease σ_{VOS} from 5mV to 1mV?

$$\sigma_X \propto \frac{A_0}{\sqrt{A_C}}$$

$$\left.\begin{aligned} \sigma_{X_5} &= \theta \frac{A_0}{\sqrt{A_{C_5}}} \\ \sigma_{X_1} &= \theta \frac{A_0}{\sqrt{A_{C_1}}} \end{aligned}\right\} \quad \Rightarrow \quad \frac{\sigma_{X_5}}{\sigma_{X_1}} = \frac{\sqrt{A_{C_1}}}{\sqrt{A_{C_5}}} = 5$$

$$A_{C_1} = 25 A_{C_5}$$

Equivalent Number of Bits (ENOB)

- Often the performance of an n-bit data converter is not commensurate with that of an ideal n-bit data converter but more like that of an n-k bit data converter

- The equivalent number of bits (ENOB) is often used to characterize the actual level of performance

- Different ENOB definitions depending upon which characterization parameter is of interest
  (e.g. INL, SFDR, SNR, …)

- ENOB is just one real number that attempts to assess the performance of a data converter that may have 2^n different outputs so it dramatically discards a lot of information about a data converter!

INL-based ENOB

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

Assume $$\text{INL} = \upsilon X_{LSBR} = \upsilon \frac{X_{REF}}{2^{n_R}}$$

where X_{LSBR} is the LSB based upon the defined resolution , n_R

Define the equivalent LSB by $$X_{LSBE} = \frac{X_{REF}}{2^{n_{EQ}}}$$

Thus (substituting for X_{REF} into INL expression):

$$\text{INL} = \upsilon \frac{2^{n_{EQ}}}{2^{n_R}} X_{LSBE} = \left[\upsilon 2^{n_{EQ}+1-n_R} \right] \frac{X_{LSBE}}{2}$$

Since an ideal ADC has an INL of $X_{LSB}/2$, Setting term in [ ] to 1, can solve for n_{EQ} to obtain

$$\text{ENOB} = n_{EQ} = n_R - 1 - \log_2(\upsilon)$$

where n_R is the defined resolution

Observe: the ENOB based upon the INL has been defined as the maximum deviation from the end-point fit line

INL-based ENOB

$$ENOB = n_R - 1 - \log_2(\upsilon)$$

Consider an ADC with specified resolution of n_R and INL of v LSB

υ	ENOB
½	n_R
1	n_R-1
2	n_R-2
4	n_R-3
8	n_R-4
16	n_R-5

Though based upon the continuous-INL definition, often used to define ENOB from INL viewpoint

ANALOG DEVICES

16-Bit, 200 MSPS/250 MSPS Analog-to-Digital Converter

Data Sheet

$120 in 1000's

AD9467

FEATURES

- 75.5 dBFS SNR to 210 MHz at 250 MSPS
- 90 dBFS SFDR to 300 MHz at 250 MSPS
- SFDR at 170 MHz at 250 MSPS
  - 92 dBFS at −1 dBFS
  - 100 dBFS at −2 dBFS
- 60 fs rms jitter
- Excellent linearity at 250 MSPS
  - ~~DNL = ±0.5 LSB typical~~
  - INL = ±3.5 LSB typical
- ~~2 V p-p to 2.5 V p-p (default)~~ differential full-scale input (programmable)
- Integrated input buffer
- External reference support option
- Clock duty cycle stabilizer
- Output clock available
- Serial port control
  - Built-in selectable digital test pattern generation
  - Selectable output data format
- LVDS outputs (ANSI-644 compatible)
- 1.8 V and 3.3 V supply operation

APPLICATIONS

- Multicarrier, multimode cellular receivers
- Antenna array positioning
- Power amplifier linearization
- Broadband wireless
- Radar
- Infrared imaging
- Communications instrumentation

FUNCTIONAL BLOCK DIAGRAM

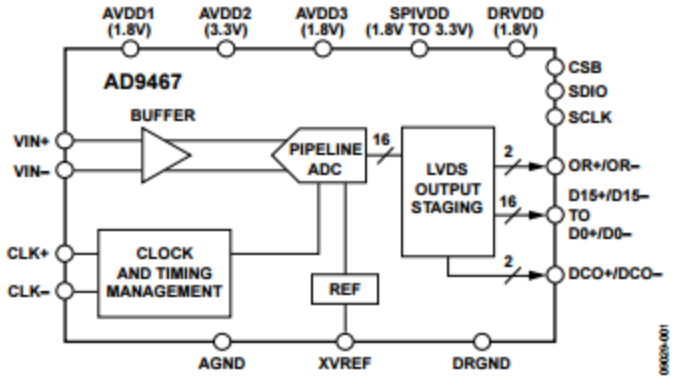


Figure 1.

ENOB $= n_R\text{-}1\text{-}\log_2(\upsilon) = 16 - 1 - 1.85 \cong 13.15$

Is this close to 16-bit performance?

A data clock output (DCO) for capturing data on the output is provided for signaling a new output bit.

The internal power-down feature supported via the SPI typically consumes less than 5 mW when disabled.

Optional features allow users to implement various selectable operating conditions, including input range, data format select, and output data test patterns.

The AD9467 is available in a Pb-free, 72-lead, LFCSP specified over the −40°C to +85°C industrial temperature range.

Test Setup Quite Sophisticated

From ADI AN-835

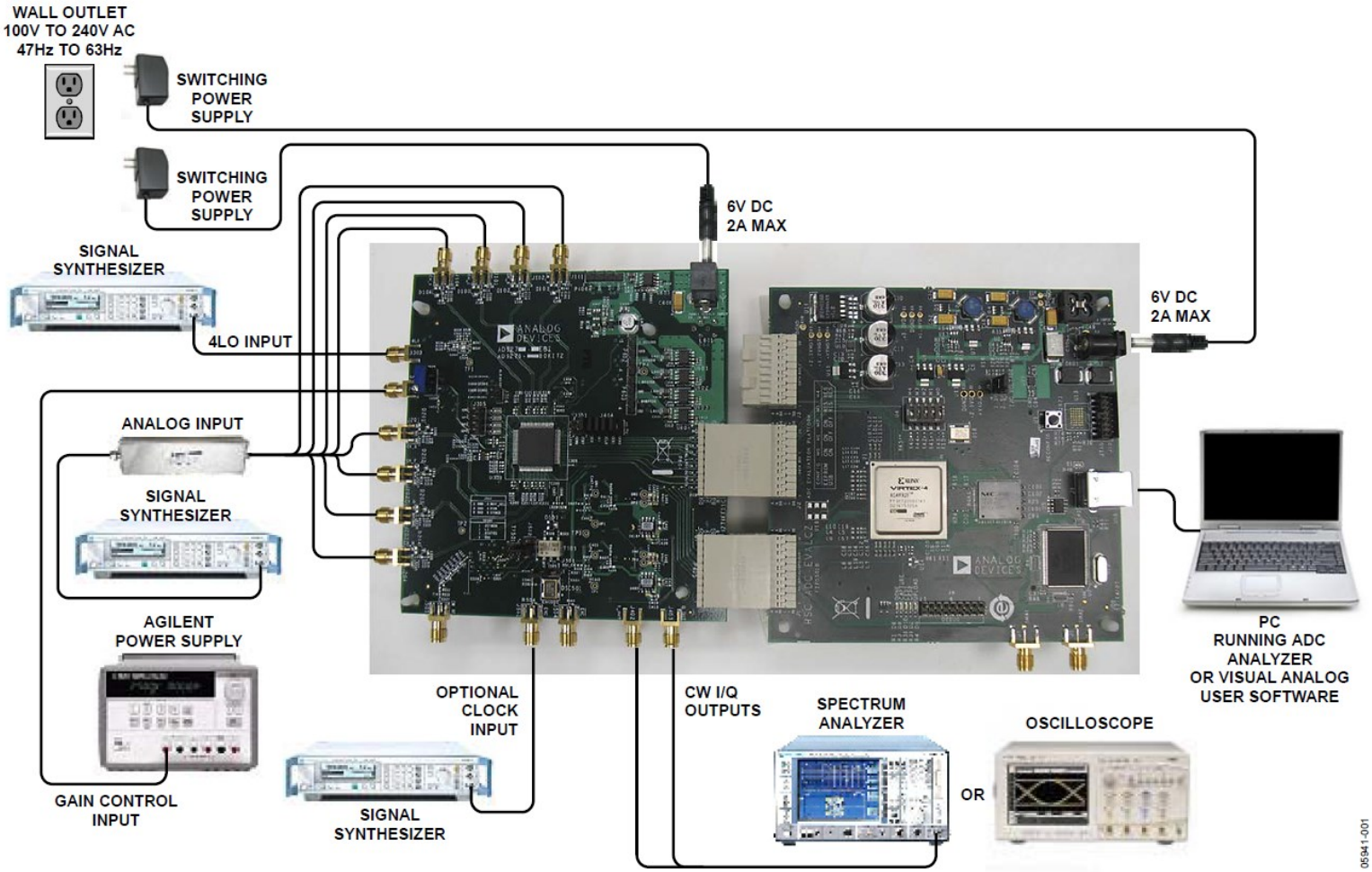


Figure 1. Typical Characterization Test Setup

Test Setup Quite Sophisticated

From ADI AN-835

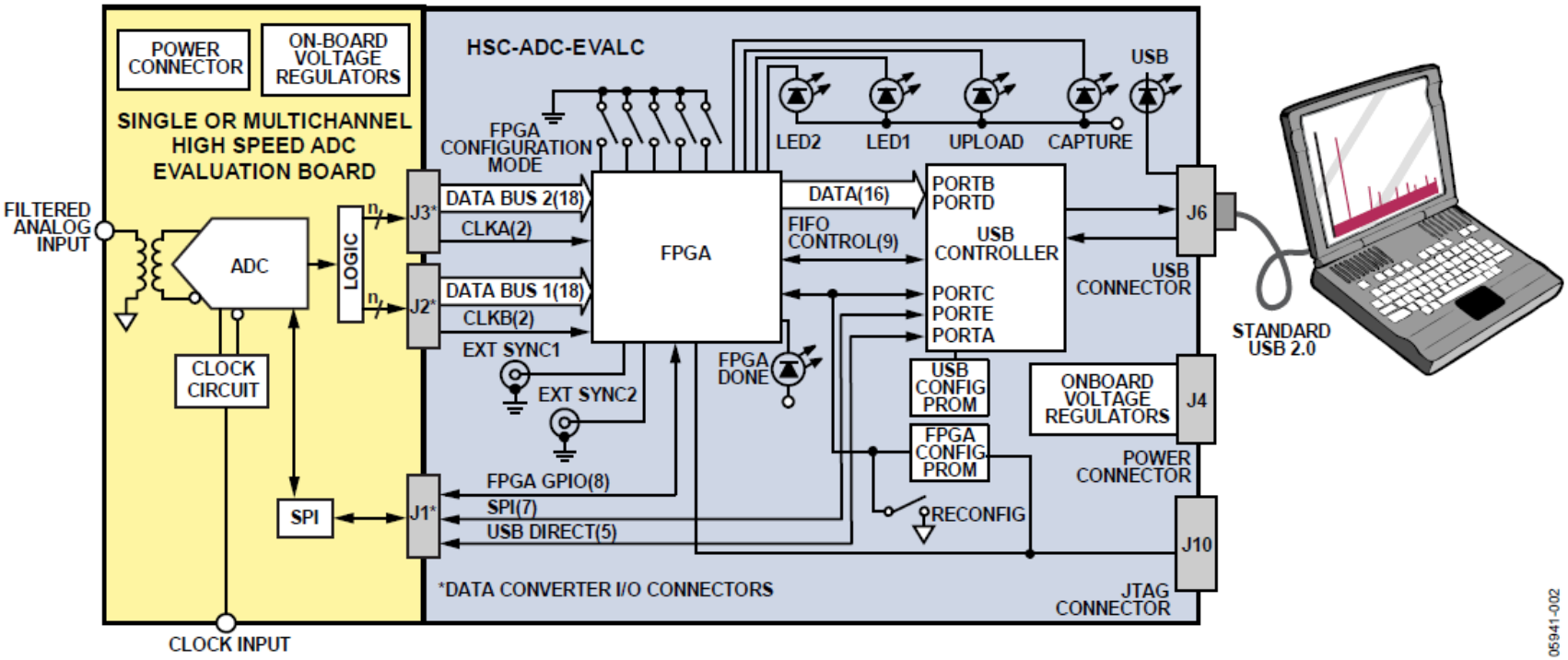


Figure 2. Typical HSC-ADC-EVALC Evaluation Platform

Test Setup Quite Sophisticated

From ADI AN-835

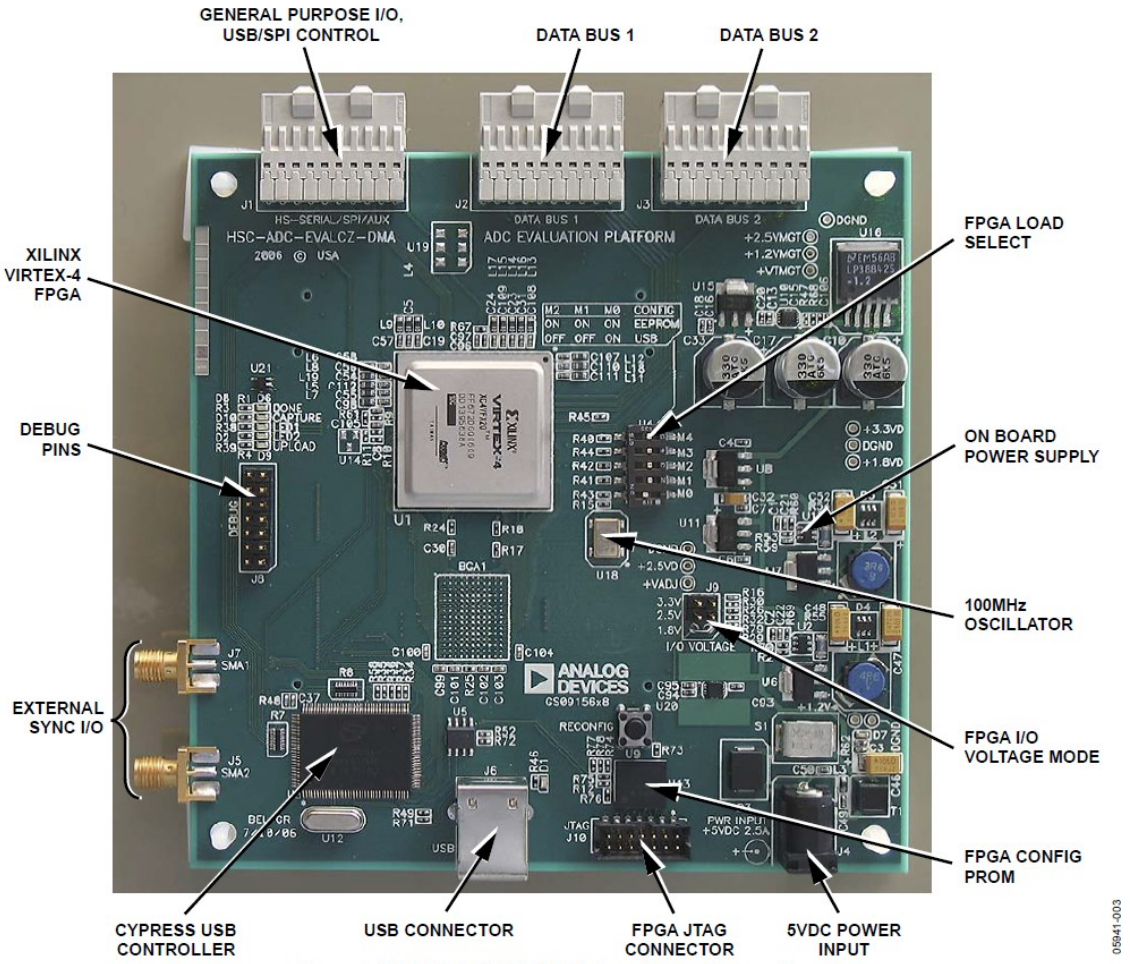


Figure 3. HSC-ADC-EVALC: FPGA-Based Data Capture Board

Can we depend on this “13-bit” INL performance?

SPECIFICATIONS

AVDD1 = 1.8 V, AVDD2 = 3.3 V, AVDD3 = 1.8 V, DRVDD = 1.8 V, specified maximum sampling rate, 2.5 V p-p differential input, 1.25 V internal reference, AIN = −1.0 dBFS, DCS on, default SPI settings, unless otherwise noted.

Table 1.

Parameter[1]	Temp	Min	Typ	Max	Unit
RESOLUTION		16			Bits
ACCURACY					
No Missing Codes	Full		Guaranteed		
Offset Error	Full	−200	0	+200	LSB
Gain Error	Full	−3.9	−0.1	+2.6	%FSR
Differential Nonlinearity (DNL)[2]	Full	−0.9	±0.5	+1.5	LSB
Integral Nonlinearity (INL)[2]	Full	−12	±3.5	+12	LSB
TEMPERATURE DRIFT					
Offset Error	Full		±0.023		%FSR/°C
Gain Error	Full		±0.036		%FSR/°C
ANALOG INPUTS					
Differential Input Voltage Range (Internal VREF = 1 V to 1.25 V)	Full	2	2.5	2.5	V p-p
Common-Mode Voltage	25°C		2.15		V
Differential Input Resistance	25°C		530		Ω
Differential Input Capacitance	25°C		3.5		pF
Full Power Bandwidth	25°C		900		MHz
XVREF INPUT					
Input Voltage	Full	1		1.25	V
Input Capacitance	Full		3		pF
POWER SUPPLY					
AVDD1	Full	1.75	1.8	1.85	V
AVDD2	Full	3.0	3.3	3.6	V
AVDD3	Full	1.7	1.8	1.9	V
DRVDD	Full	1.7	1.8	1.9	V
I_{AVDD1}	Full		567	620	mA
I_{AVDD2}	Full		55	61	mA
I_{AVDD3}	Full		31	35	mA
I_{DRVDD}	Full		40	43	mA
Total Power Dissipation (Including Output Drivers)	Full		1.33	1.5	W
Power-Down Dissipation	Full		4.4	90	mW

[1] See the AN-835 Application Note, *Understanding High Speed ADC Testing and Evaluation*, for a complete set of definitions and how these tests were completed.
[2] Measured with a low input frequency, full-scale sine wave, with approximately 5 pF loading on each output bit.

$$\text{ENOB} = n_R\text{-}1\text{-}\log_2(\upsilon) = 16 - 1 - 3.58 \cong 11.42$$

From INL viewpoint, performance is about 4.5 bits less than physical resolution but does have other attractive properties

AC SPECIFICATIONS

AVDD1 = 1.8 V, AVDD2 = 3.3 V, AVDD3 = 1.8 V, DRVDD = 1.8 V, specified maximum sampling rate, 2.5 V p-p differential input, 1.25 V internal reference, AIN = −1.0 dBFS, DCS on, default SPI settings, unless otherwise noted.

Table 2.

Parameter[1]	Temp	Min	Typ	Max	Unit
ANALOG INPUT FULL SCALE		2.5	2/2.5		V p-p
SIGNAL-TO-NOISE RATIO (SNR)					
f_{IN} = 5 MHz	25°C		74.7/76.4		dBFS
f_{IN} = 97 MHz	25°C		74.5/76.1		dBFS
f_{IN} = 140 MHz	25°C		74.4/76.0		dBFS
f_{IN} = 170 MHz	25°C	73.7	74.3/75.8		dBFS
	Full	71.5			dBFS
f_{IN} = 210 MHz	25°C		74.0/75.5		dBFS
f_{IN} = 300 MHz	25°C		73.3/74.6		dBFS
SIGNAL-TO-NOISE AND DISTORTION RATIO (SINAD)					
f_{IN} = 5 MHz	25°C		74.6/76.3		dBFS
f_{IN} = 97 MHz	25°C		74.4/76.0		dBFS
f_{IN} = 140 MHz	25°C		74.4/76.0		dBFS
f_{IN} = 170 MHz	25°C	72.4	74.2/75.8		dBFS
	Full	71.0			dBFS
f_{IN} = 210 MHz	25°C		73.9/75.4		dBFS
f_{IN} = 300 MHz	25°C		73.1/74.4		dBFS
EFFECTIVE NUMBER OF BITS (ENOB)					
f_{IN} = 5 MHz	25°C		12.1/12.4		Bits
f_{IN} = 97 MHz	25°C		12.1/12.3		Bits
f_{IN} = 140 MHz	25°C		12.1/12.3		Bits
f_{IN} = 170 MHz	25°C		12.0/12.3		Bits
	Full	11.5			Bits
f_{IN} = 210 MHz	25°C		12.0/12.2		Bits
f_{IN} = 300 MHz	25°C		11.9/12.1		Bits
SPURIOUS-FREE DYNAMIC RANGE (SFDR) (INCLUDING SECOND AND THIRD HARMONIC DISTORTION)					
f_{IN} = 5 MHz	25°C		98/97		dBFS
f_{IN} = 97 MHz	25°C		95/93		dBFS
f_{IN} = 140 MHz	25°C		94/95		dBFS
f_{IN} = 170 MHz	25°C	82	93/92		dBFS
	Full	82			dBFS
f_{IN} = 210 MHz	25°C		93/92		dBFS
f_{IN} = 300 MHz	25°C		93/90		dBFS
SFDR (INCLUDING SECOND AND THIRD HARMONIC DISTORTION)					
f_{IN} = 5 MHz at −2 dB Full Scale	25°C		100/100		dBFS
f_{IN} = 97 MHz at −2 dB Full Scale	25°C		97/97		dBFS
f_{IN} = 140 MHz at−2 dB Full Scale	25°C		100/95		dBFS
f_{IN} = 170 MHz at −2 dB Full Scale	25°C		100/100		dBFS
f_{IN} = 210 MHz at −2 dB Full Scale	25°C		93/93		dBFS
f_{IN} = 300 MHz at −2 dB Full Scale	25°C		90/90		dBFS
WORST OTHER (EXCLUDING SECOND AND THIRD HARMONIC DISTORTION)					
f_{IN} = 5 MHz	25°C		98/97		dBFS
f_{IN} = 97 MHz	25°C		97/93		dBFS
f_{IN} = 140 MHz	25°C		97/95		dBFS
f_{IN} = 170 MHz	25°C	88	97/93		dBFS
	Full	82			dBFS
f_{IN} = 210 MHz	25°C		97/95		dBFS
f_{IN} = 300 MHz	25°C		97/95		dBFS

- Can be defined different ways
- Only given as typical

INL-based ENOB

Since the break-point INL is ideally 0, it is not related to either X_{LSB} or X_{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 given by the expression

$$ENOB = n_R - 1 - \log_2(\upsilon)$$

where n_R is the specified resolution and v is the INL in LSB at the n_R bit level.

Observe: the ENOB based upon the INL has been defined as the maximum deviation from the end-point fit line

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

$$ENOB = n_R - 1 - \log_2(\upsilon)$$

where n_R is the specified resolution and v is the INL in LSB at the n_R bit level.

Observe: the ENOB based upon the INL has been defined as the maximum deviation from the end-point fit line

Question: With this definition, is it possible for a data converter to have an ENOB that is actually larger than n_R? YES !

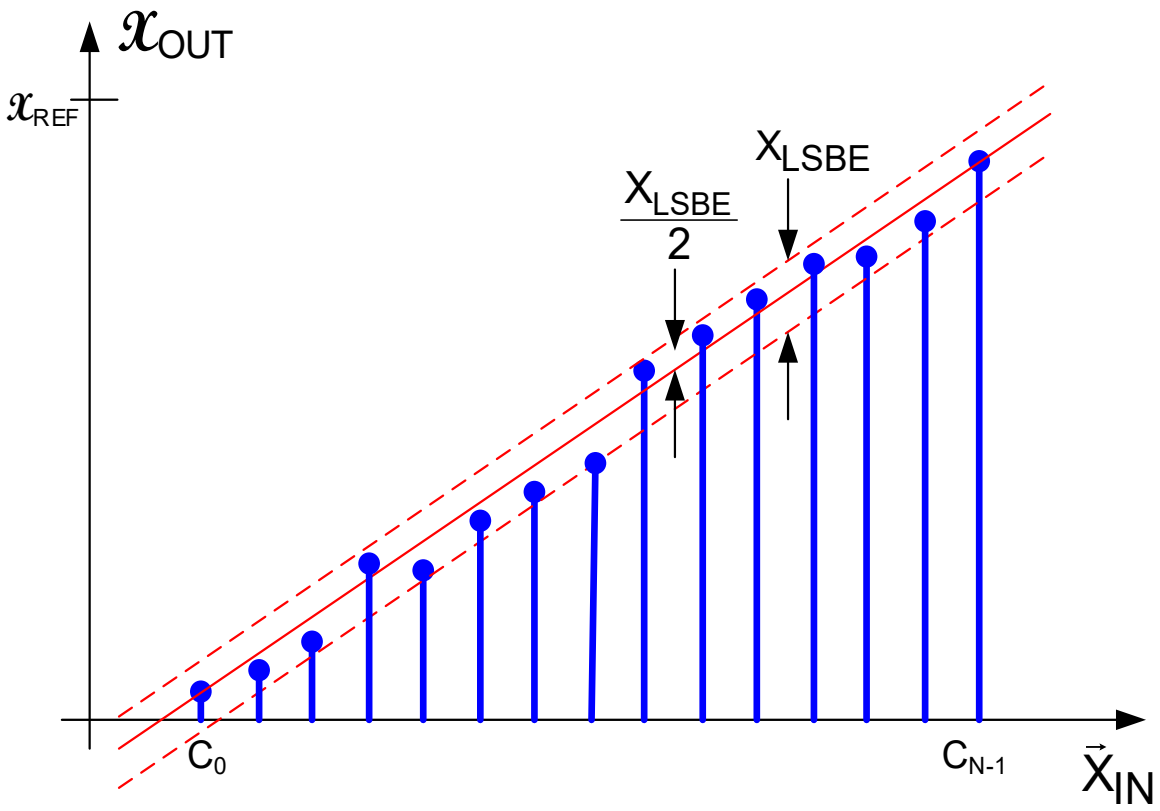
Question: What is the ENOB of any 1-bit ADC? ∞ !

Question: Is it easy to design a 4-bit ADC with an ENOB of 7 bits? YES !

Question: Is it easy to design a 14-bit ADC with an ENOB of 16 bits? No !

Question: Is ENOB (based on INL) a systematic metric?

INL-based ENOB



Interpretation of ENOB definition for a DAC:

A DAC with n_{EFF} bits (ENOB) of resolution should have all outputs bounded by +/- $X_{LSBE}/2$ from the fit line so distance between fit line and upper/lower bounding lines determines the ENOB (X_{LSBE} is relative to n_{EFF} bits)

INL-based ENOB

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

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

where n_R is the specified resolution and v is the resolution in LSB at the n_R bit level.

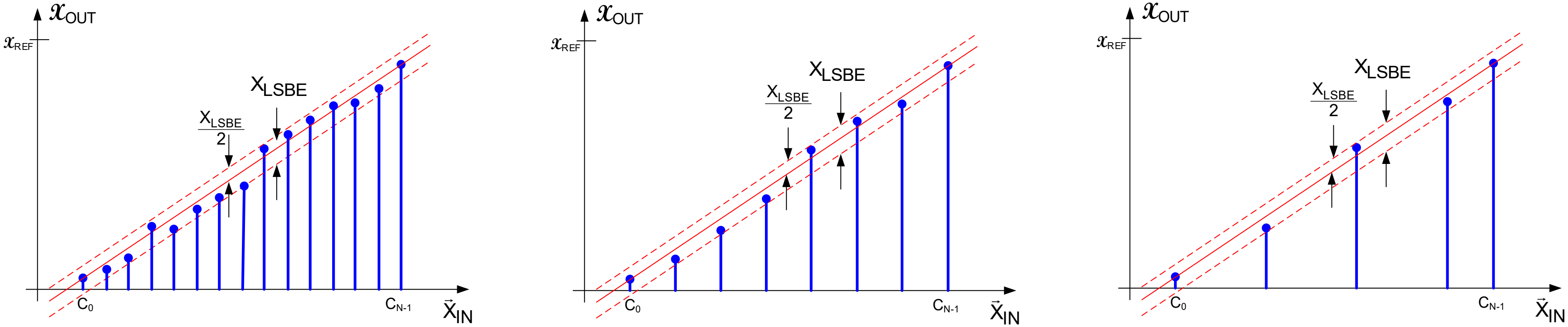
Observation: The ENOB was defined relative to a fit line and was not dependent upon the number of DAC levels or the number of break points in the ADC

Question: Then, why does n_R appear in the ENOB expression?

INL-based ENOB

Question: Then, why does n_R appear in the ENOB expression?

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



Normalization was with respect to the LSB which is dependent upon n_R

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 v_A LSB and a converter with the same linearity was specified with n_{RB} bits of resolution and an INL of v_B LSB.

Since there are simply two representations of the same nonlinearity, the absolute INL will be the same for both representations. That is,

$$INL_A = INL_B \quad (1)$$

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

$$INL_A = \upsilon_A X_{LSBA} \quad (2)$$

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

$$\frac{X_{LSBA}}{X_{REF}} = 2^{-n_{RA}} \quad (3)$$

INL-based ENOB

Proof (cont)

Thus we obtain the expression

$$INL_A = \upsilon_A \, 2^{-n_{RA}} \, X_{REF} \qquad (4)$$

and the ENOB is given by

$$ENOB_A = n_{RA} - 1 - \log_2(\upsilon_A) \qquad (5)$$

Substituting from (4) into (5) we obtain

$$ENOB_A = \log_2(X_{REF}) - 1 - \log_2(INL_A) \qquad (6)$$

By a similar argument we obtain

$$ENOB_B = n_{RB} - 1 - \log_2(\upsilon_B) \qquad (7)$$

and

$$ENOB_B = \log_2(X_{REF}) - 1 - \log_2(INL_B) \qquad (8)$$

Now,since $INL_A = INL_B$, it follows that

$$ENOB_A = ENOB_B$$

INL-based ENOB

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

$$ENOB = \log_2(X_{REF}) - \log_2(INL_{REF}) - 1$$

where INL_{REF} is the INL expressed relative to X_{REF}.

Proof: follows directly from proof of previous theorem

$$ENOB = ENOB_A = \log_2(X_{REF}) - 1 - \log_2(INL_A) = \log_2(X_{REF}) - 1 - \log_2(INL_{REF})$$

To avoid possible misinterpretation, INL_{REF} defined below

$$INL_{REF} = \frac{INL_V}{V_{REF}}$$

where INL_V is the deviation in volts from the end-point fit line and $X_{REF}=V_{REF}$

INL-based ENOB

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

$$ENOB=\log_2(X_{REF})-\log_2(INL_{REF})-1$$

where INL_{REF} is the INL expressed relative to X_{REF}.

Again, observe the INL-based ENOB does not depend upon the number of bits of resolution !

Recall previous question: can the INL-based ENOB on an n-bit ADC or DAC exceed n?

The answer was YES but in most such data converters it would probably be relatively easy to increase the number of bits of resolution while maintaining the ENOB

If the INL-based ENOB of a data converter exceeds n, (with a rather degenerate exception) it is probably over-designed

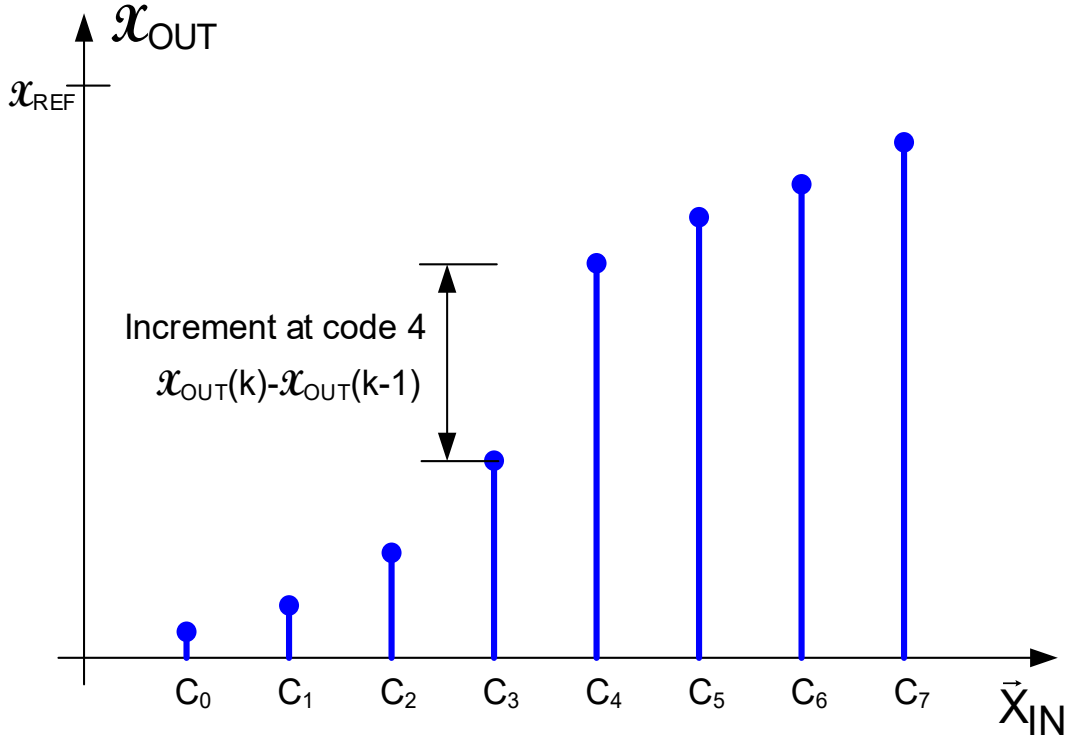
Designing a data converter of more than 1 bit that has a high number of bits of linearity (as measured by ENOB) is challenging

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

Differential Nonlinearity (DAC)

Nonideal DAC

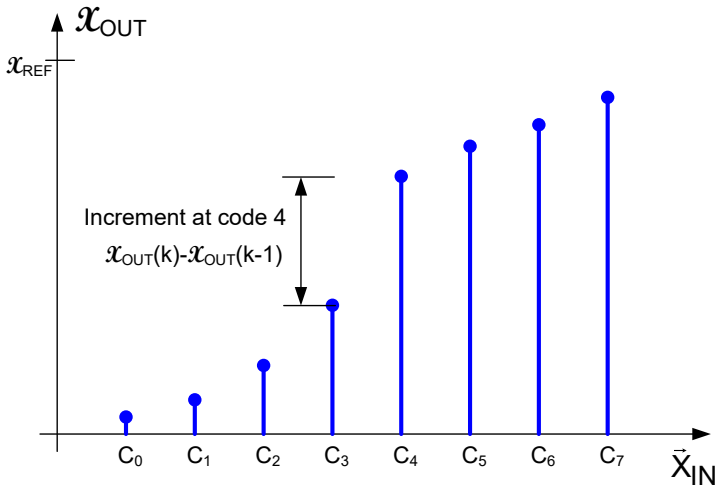


DNL(k) is the actual increment from code (k-1) to code k minus the ideal increment normalized to X_{LSB}

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

Differential Nonlinearity (DAC)

Nonideal DAC



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

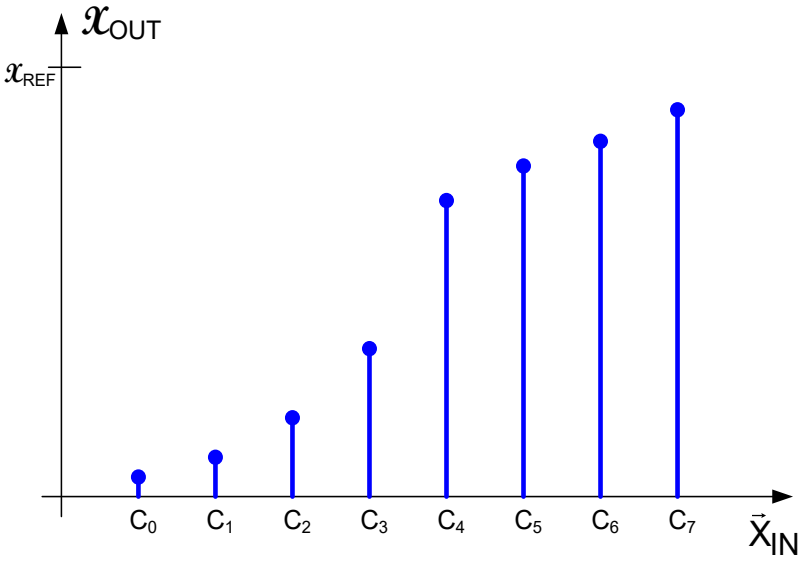
$$DNL(k)=\frac{X_{OUT}(k)-X_{OUT}(k-1)-X_{LSB}}{X_{LSB}}$$

$$DNL=\max_{1\le k\le N-1}\left\{\left|DNL(k)\right|\right\}$$

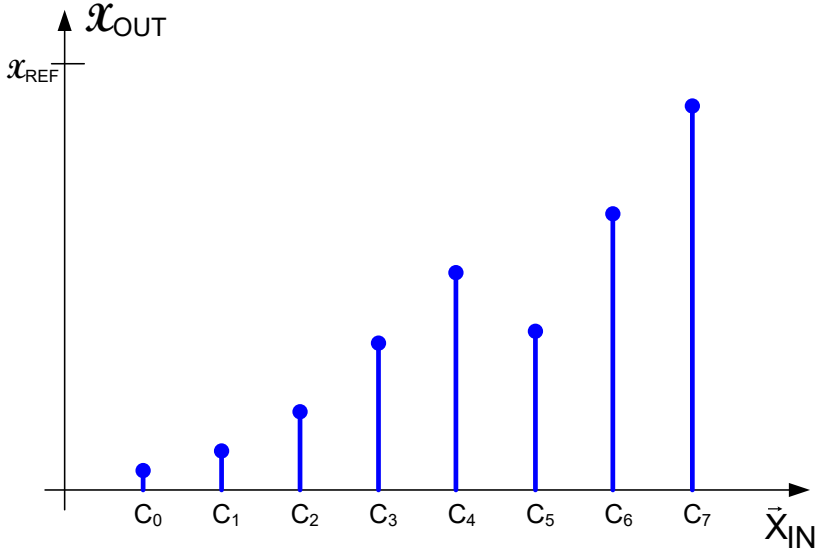
DNL=0 for an ideal DAC

Monotonicity (DAC)

Nonideal DAC



Monotone DAC



Non-monotone DAC

Definition:

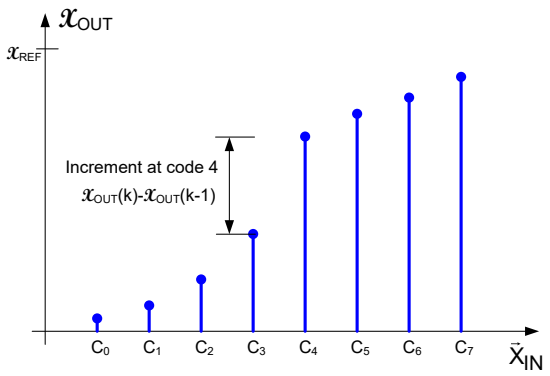
A DAC is monotone if $x_{OUT}(k) > x_{OUT}(k-1)$ for all k

Theorem:

A DAC is monotone if DNL(k)> -1 for all k

Differential Nonlinearity (DAC)

Nonideal DAC



Theorem: The INL_k of a DAC (when corrected for gain error and offset) can be obtained from the DNL by the expression

$$INL_k = \sum_{i=1}^{k} DNL(i)$$

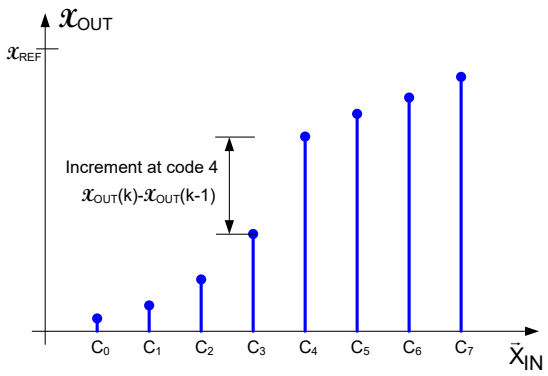
Caution: Be careful about using this theorem to measure the INL since errors in DNL measurement (or simulation) can accumulate

Corollary: The DNL of a DAC (when corrected for gain error and offset) can be expressed as

$$DNL(k) = INL_k - INL_{k-1}$$

Differential Nonlinearity (DAC)

Nonideal DAC



Theorem: If the INL of a DAC satisfies the relationship

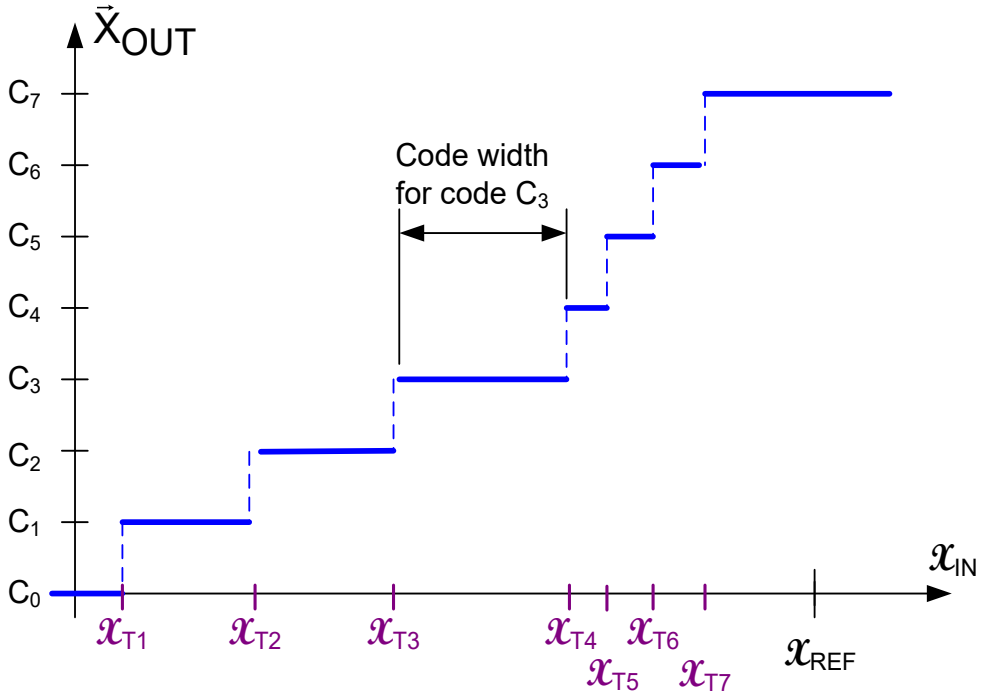
$$INL_k < \frac{1}{2} X_{LSB}$$

for all k, then the DAC is monotone

Note: This is a sufficient but not necessary condition for monotonicity

Differential Nonlinearity (ADC)

Nonideal ADC

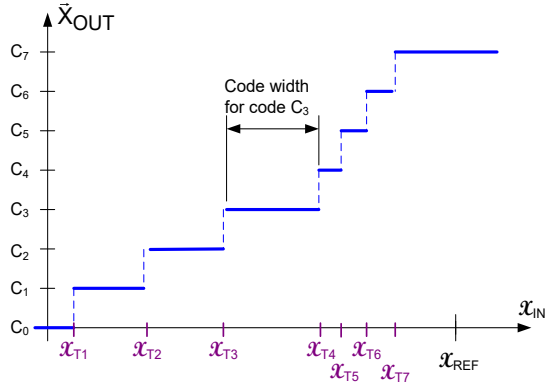


DNL(k) is the code width for code k – ideal code width normalized to X_{LSB}

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

Differential Nonlinearity (ADC)

Nonideal ADC



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

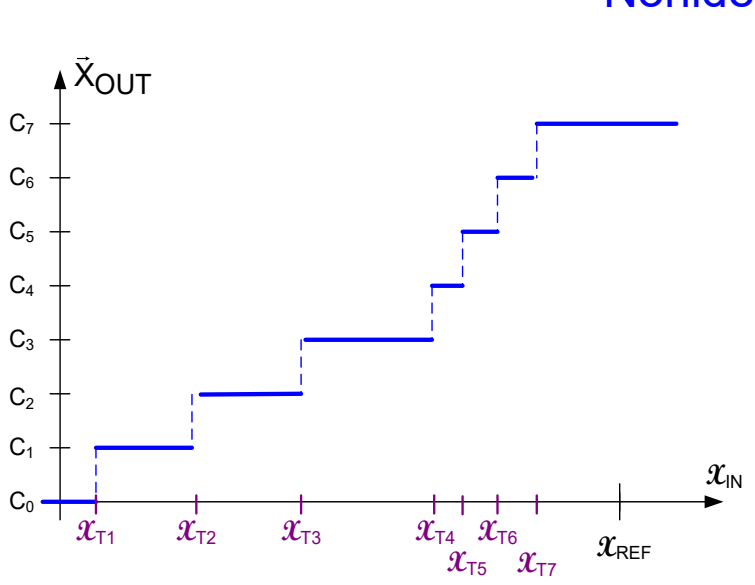
$$DNL=\max_{2\le k\le N-1}\left\{\left|DNL(k)\right|\right\}$$

DNL=0 for an ideal ADC

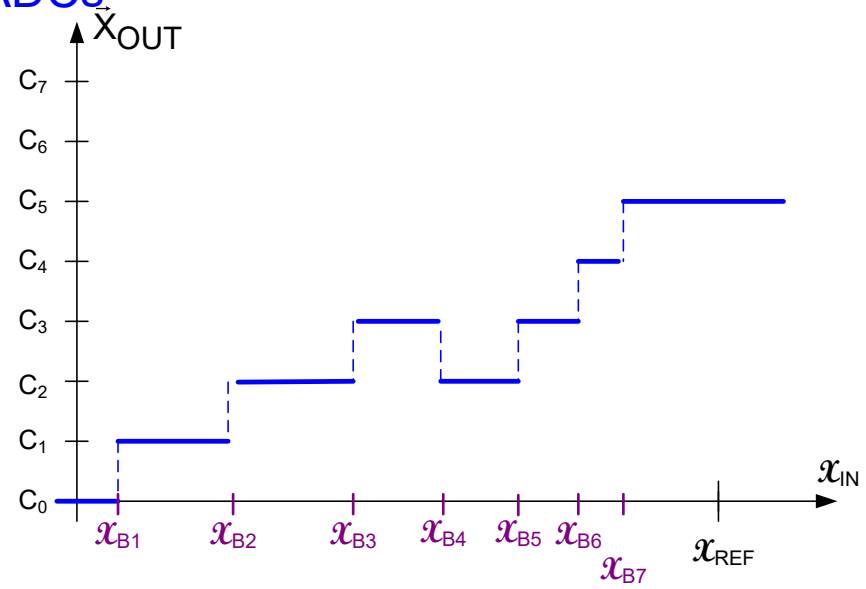
Note: In some nonideal ADCs, two or more break points could cause transitions to the same code C_k making the definition of DNL ambiguous

Monotonicity in an ADC

Nonideal ADCs



Monotone ADC



Nonmonotone ADC

Definition: An ADC is monotone if the

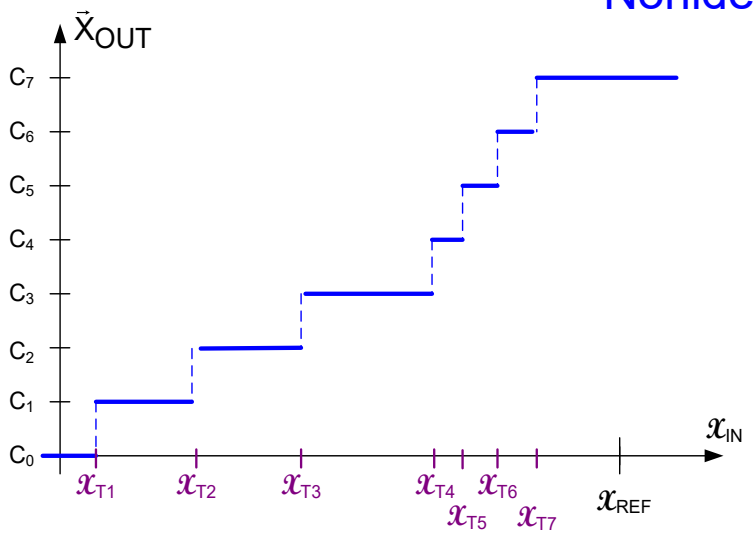
$$\vec{X}_{OUT}(x_k) \geq \vec{X}_{OUT}(x_m) \quad \text{whenever} \quad x_k \geq x_m$$

Note: Have used x_{Bk} instead of x_{Tk} in figure on right since more than one transition point corresponds to a given code

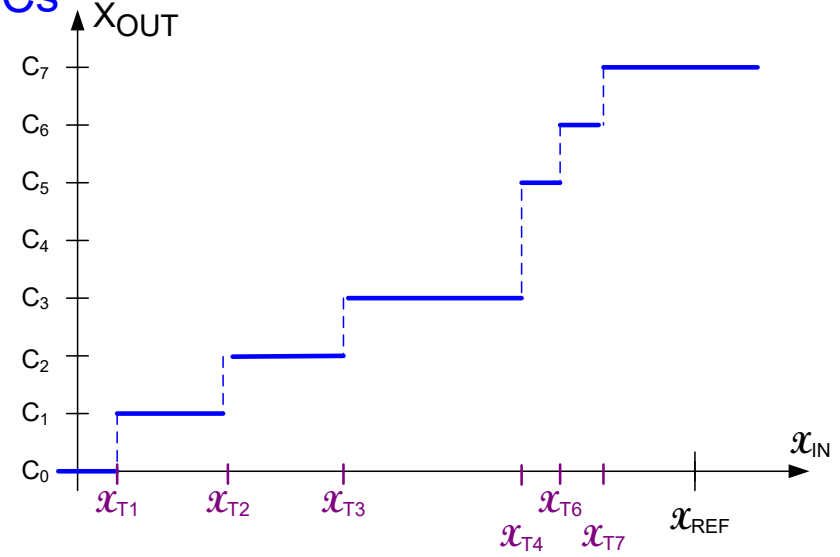
Note: Some authors do not define monotonicity in an ADC.

Missing Codes (ADC)

Nonideal ADCs



No missing codes



One missing code

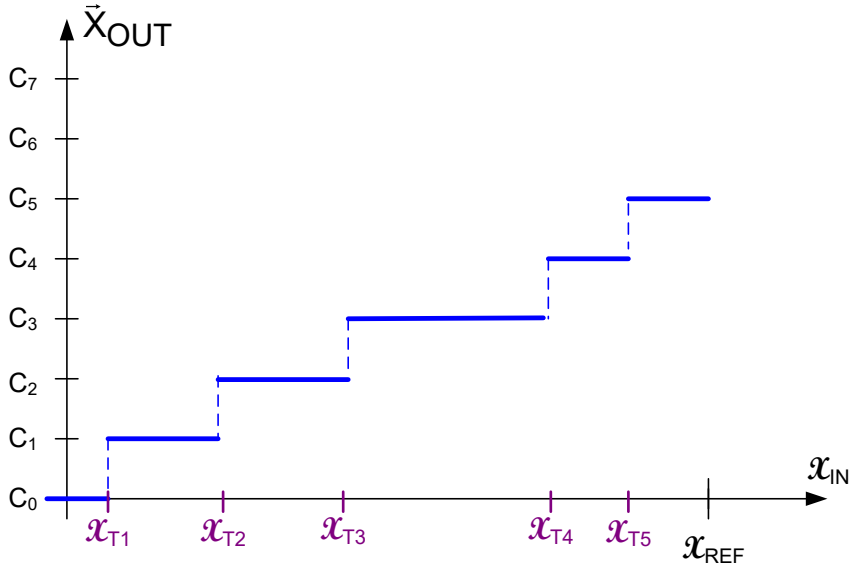
Definition: An ADC has no missing codes if there are N-1 transition points and a single LSB code increment occurs at each transition point. If these criteria are <u>not</u> satisfied, we say the ADC has missing code(s).

Note: With this definition, all codes can be present but we still say it has "missing codes"

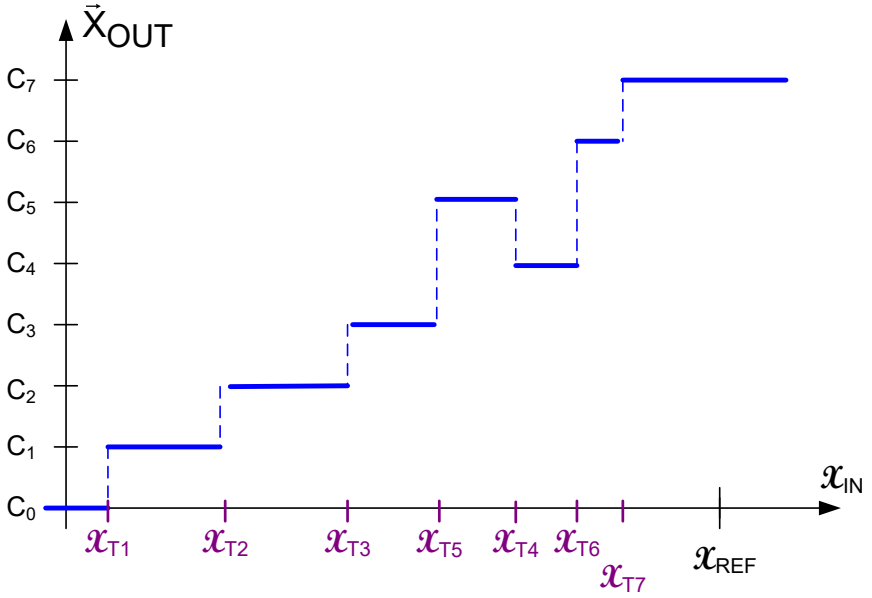
Note: Some authors claim that missing codes in an ADC are the counterpart to nonmonotonicity in a DAC. This association is questionable.

Missing Codes (ADC)

Nonideal ADCs



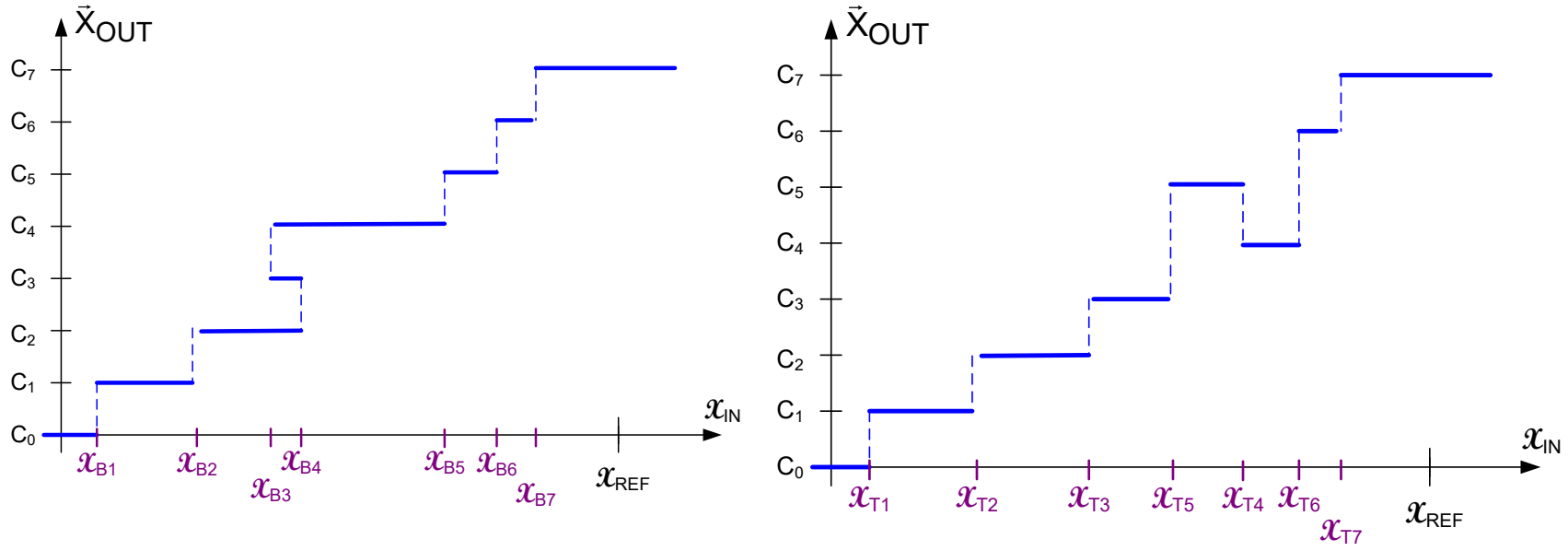
Missing codes



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

What is X_{LSB}?

LSB Definition

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

What is X_{LSB}?

Conventional Wisdom X_{LSB}

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

(X_{LSB} determined by specified resolution and can not be measured)

Alternate LSB Definition

X_{LSB} appears in many performance specifications but a distinction in X_{LSB} that differs from that obtained from specified values for X_{REF} and n_R is generally not given. This can cause modest inconsistencies in the definition of some performance specifications.

DAC

Alternate definitions of X_{LSB}

where N is the measured number of DAC output levels

$$X_{LSB} = \frac{X_{REF}}{N}$$

---

where N is the measured number of DAC output levels and $X_0(N\text{-}1)$ and $X_0(0)$ are last and first outputs respectively

$$X_{LSB} = \frac{X_0(N\text{-}1) - X_0(0)}{N\text{-}1}$$

---

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

$$X_{LSB} = \frac{\max_k\{X_0(k)\} - \min_k\{X_0(k)\}}{N\text{-}1}$$

---

useful for determining worst-case resolution of a DAC

$$X_{LSB} = \max_k\{X_0(k) - X_0(k\text{-}1)\}$$

ADC

Similar definitions can be made for X_{LSB} of an ADC based upon the breakpoints

Alternate LSB Definition

Is the concept of an LSB that is based upon measurements useful?

In many control applications, the largest gap between outputs of a DAC is often of interest and though that is ideally V_{LSB}, it may differ significantly

ENOB based upon DNL

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

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

ENOB relative to resolution

Summary of previous observations relating to ENOB (based upon INL) :

If an n-bit data converter has an INL of ¼ LSB, it is really performing from a linearity viewpoint at the n+1 bit level and if it has an INL of 1/8 LSB it is really performing at the n+2 bit level

Correspondingly, if it has a DNL of ¼ LSB, it is also performing from a differential linearity viewpoint at the n+1 bit level

The ENOB (based upon INL) of a data converter can exceed the number of bits of resolution of the data converter

Some applications benefit from an ENOB that exceeds the resolution of the data converter

Limitations of INL & DNL in Characterizing Linearity

See Lecture 4

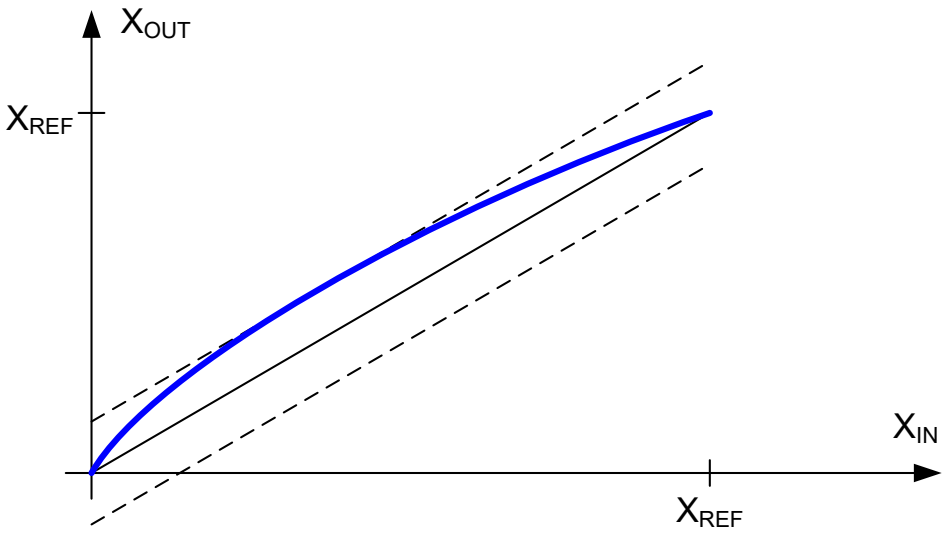
- **INL is a key parameter that is attempting to characterize the overall linearity of a DAC !**
- **INL is a key parameter that is attempting to characterize the overall linearity of an ADC !**
- **DNL is a key parameter that is attempts to characterize the local linearity of a DAC !**
- **DNL is a key parameter that is attempts to characterize the local linearity of an ADC !**

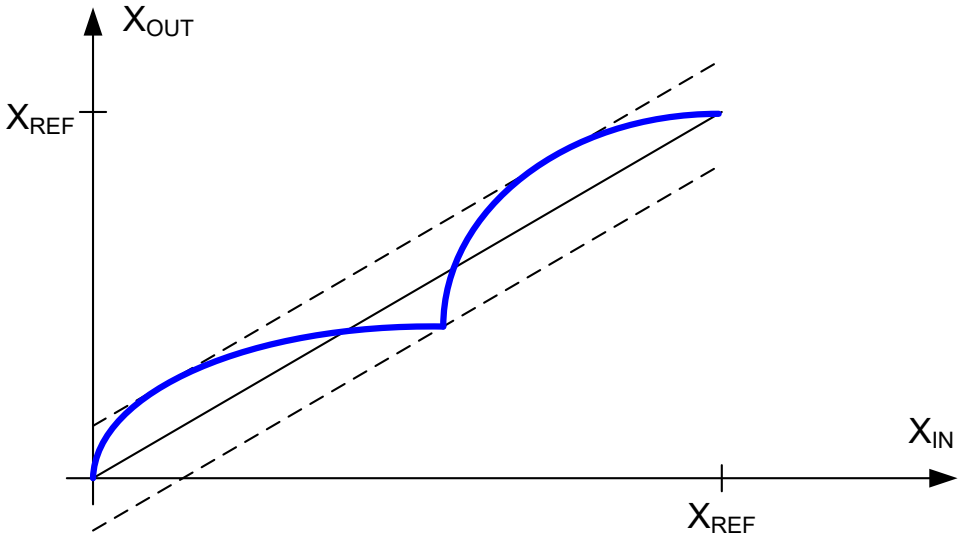
Are INL and DNL effective at characterizing the linearity of a data converter?

Limitations of INL & DNL in Characterizing Linearity

Consider the following 4 transfer characteristics, all of which have the same INL

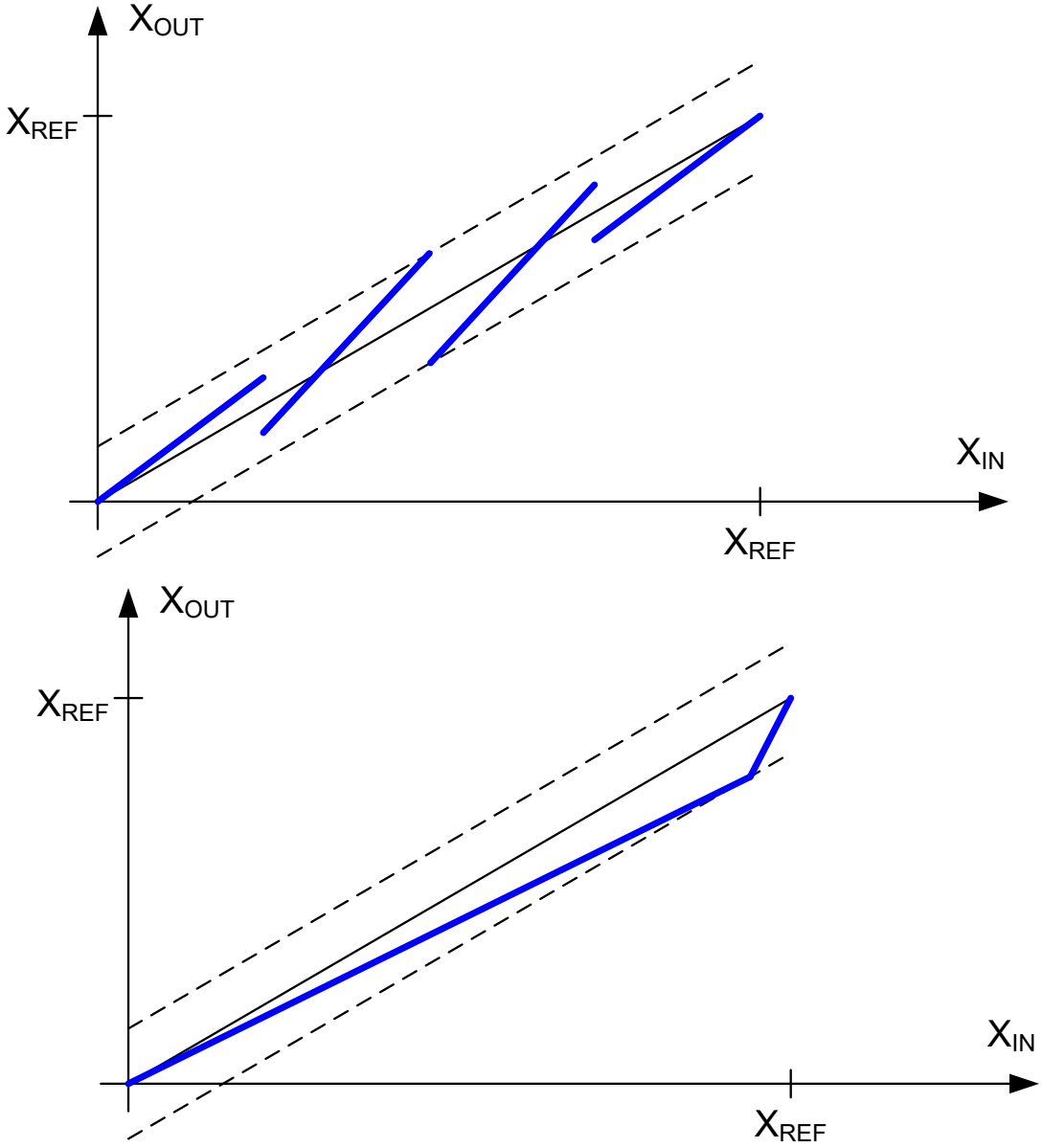
See Lecture 4





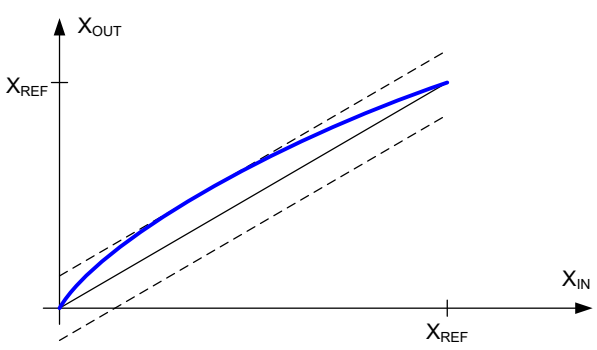
Limitations of INL & DNL in Characterizing Linearity

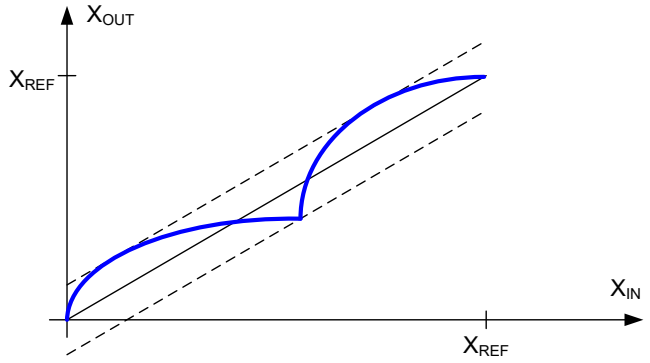
See Lecture 4

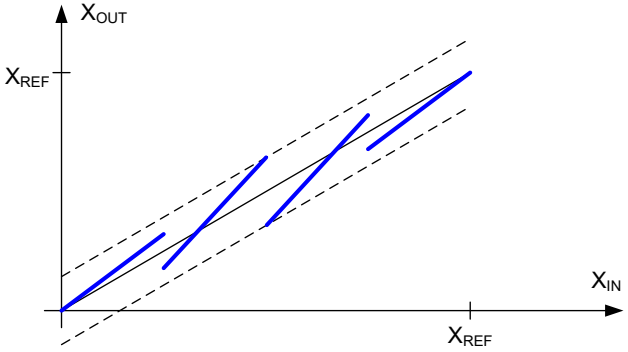


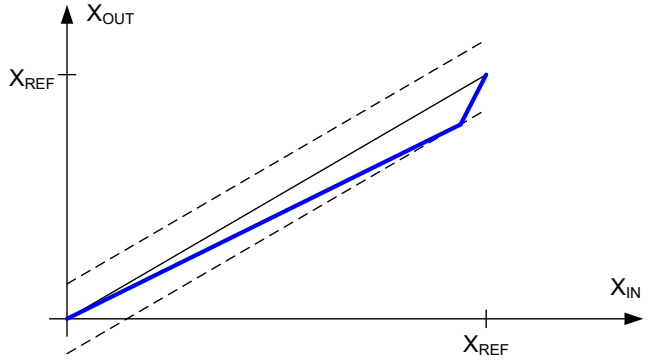
Limitations of INL & DNL in Characterizing Linearity

See Lecture 4









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 in many applications) linearity measure for data converters

Stay Safe and Stay Healthy !

End of Lecture 3