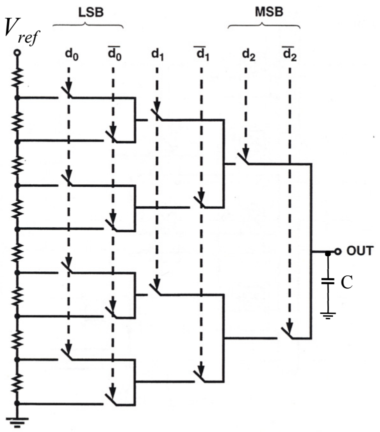
EE247
Lecture 14

- D/A converters continued:
  - Resistor string DACs (continued)
  - Serial charge redistribution DACs
  - Charge scaling DACs
  - R-2R type DACs
  - Current based DACs
  - Static performance of D/As
    - Component matching
    - Systematic & random errors
  - Practical aspects of current-switched DACs
  - Segmented current-switched DACs

R-String DAC

- Advantages:
  - Takes full advantage of availability of almost perfect switches in MOS technologies
  - Simple, fast for <8-10bits
  - Inherently monotonic
  - Compatible with purely digital technologies
- Disadvantages:
  - 2^B resistors & ~2×2^B switches for B bits → High element count & large area for B >10bits
  - High settling time for high resolution DACs:

    $\tau_{max} \sim 0.25 \times 2^B RC$



Ref:

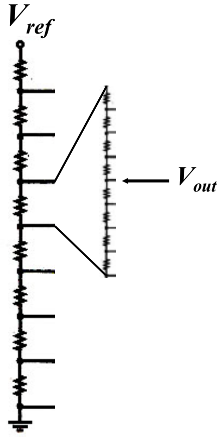
M. Pelgrom, "A 10-b 50-MHz CMOS D/A Converter with 75-W Buffer," JSSC, Dec. 1990, pp. 1347

R-String DAC Including Interpolation

Resistor string DAC + Resistor string interpolator increases resolution w/o drastic increase in complexity
e.g. 10bit DAC→ (5bit +5bit→ $2x2^5=2^6$ # of Rs) instead of direct 10bit→2^{10}

Considerations:

- ❑ Main R-string loaded by the interpolation string resistors
- ❑ Large R values for interpolating string→ less loading but lower speed
- ❑ Can use buffers

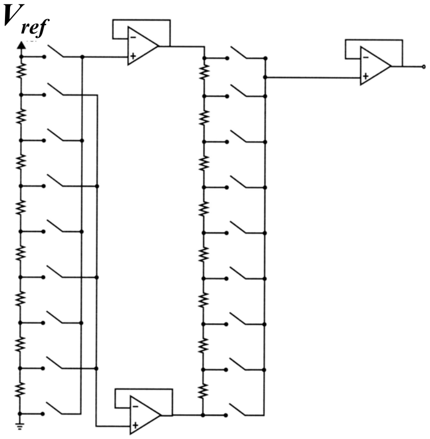


R-String DAC Including Interpolation

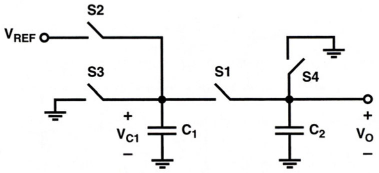
Use buffers to prevent loading of the main ladder

Issues:

- → Buffer DC offset
- → Effect of buffer bandwidth limitations on overall speed



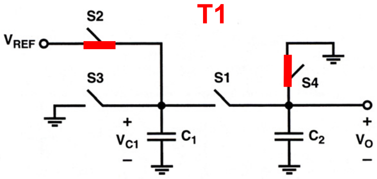
Charge Based: Serial Charge Redistribution DAC Simplified Operation

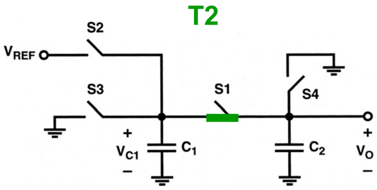


Nominally $C_1=C_2$

- Operation based on redistribution of charge associated with C1 & C2 to perform accurate division by factor of **2**

Charge Based: Serial Charge Redistribution DAC Simplified Operation: Conversion Sequence





$$Q_{C_1}^{T1}=V_{REF}\times C_1 \quad \& \qquad Q_{C_1}^{T1}=0$$

$$\rightarrow Q_{C_1}^{T1}+Q_{C_2}^{T1}=V_{REF}\times C_1$$

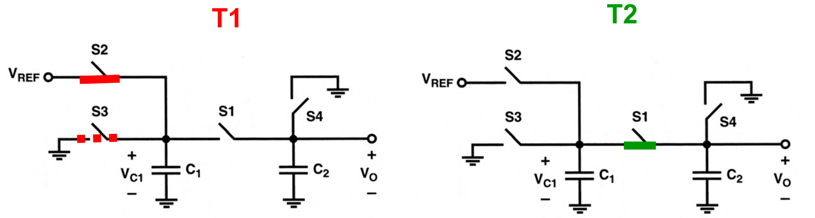
$$Q_{C_1}^{T1}+Q_{C_2}^{T1}=Q_{C_1}^{T2}+Q_{C_2}^{T2}=(C_1+C_2)V_O$$

$$V_{REF}\times C_1=(C_1+C_2)V_O$$

$$V_O=V_{REF}\times\frac{C_1}{C_1+C_2}$$

$$Since\ C_1=C_2 \quad \rightarrow \quad \boxed{V_O=\frac{V_{REF}}{2}}$$

Serial Charge Redistribution DAC Simplified Operation (Cont'd)

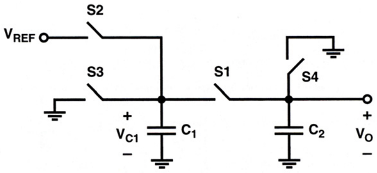


- Conversion sequence:
  - Next cycle
    - If S3 closed $V_{C1}=0$ then when S1 closes $V_{C1} = V_{C2} = V_{REF}/4$
    - If S2 closed $V_{C1}=V_{REF}$ then when S1 closes $V_{C1} = V_{C2} = V_{REF}/2+V_{REF}/4$

Serial Charge Redistribution DAC

- Conversion sequence:
  - Discharge C1 & C2→ S3& S4 closed
  - For each bit in succession beginning with LSB, b_1:
    - S1 open- if b_i=1 C1 precharge to V_{REF} if b_i=0 discharged to GND
    - S2 & S3 & S4 open- S1 closed- Charge sharing C1 & C2

      → ½ of precharge on C1 +½ of charge previously stored on C2→ C2



$$V_o(1) = \frac{b_N}{2} V_{REF}$$

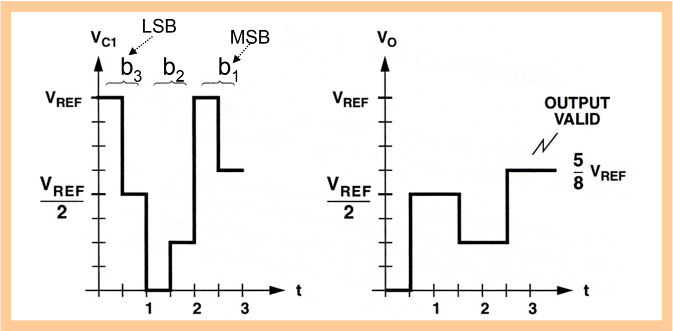
$$V_o(2) = \frac{1}{2}\left(b_{N-1} + \frac{b_N}{2}\right) V_{REF}$$

$$\vdots$$

$$V_o(N) = \left(\sum_{i=1}^{N} \frac{b_i}{2^i}\right) V_{REF}$$

Serial Charge Redistribution DAC

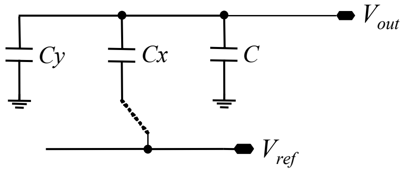
Example: Input Code 101



- Example input code 101→ output (4/8 +0/8 +1/8)V_{REF} =5/8 V_{REF}
- Very small area
- For an N-bit DAC, N redistribution cycles for one full analog output generation → quite slow

Parallel Charge Scaling DAC

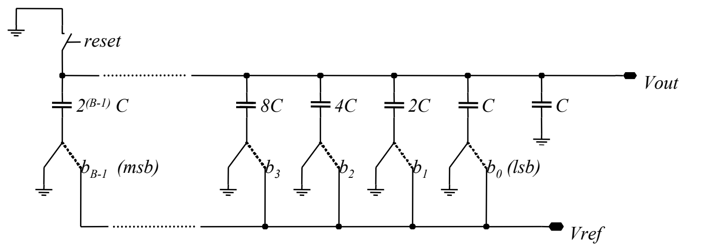
- DAC operation based on capacitive voltage division



$$V_{out} = \frac{Cx}{Cx+Cy+C}V_{ref}$$

→ Make *Cx* & *Cy* function of incoming DAC digital word

Parallel Charge Scaling DAC

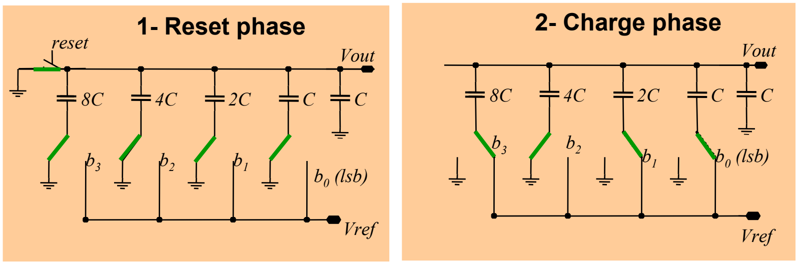


- E.g. “Binary weighted”

- B+1 capacitors & switches (Cs built of unit elements → 2^B units of C)

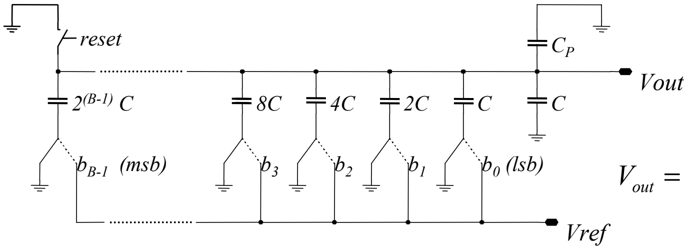
$$V_{out} = \frac{\sum_{i=0}^{B-1} b_i 2^i C}{2^B C} V_{ref}$$

Charge Scaling DAC
Example: 4Bit DAC- Input Code 1011



$$V_{out} = \frac{2^0 C + 2^1 C + 2^3 C}{2^4 C} V_{ref} = \frac{11}{16} V_{ref}$$

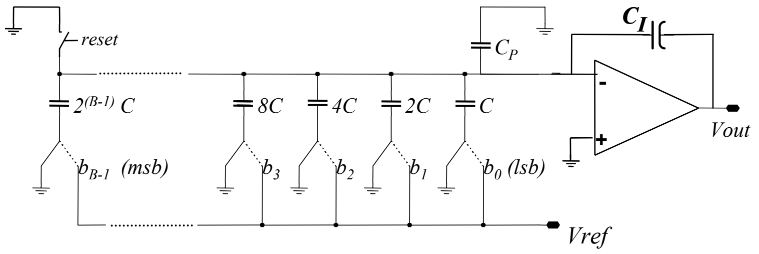
Charge Scaling DAC



$$V_{out} = \frac{\sum_{i=0}^{B-1} b_i 2^i C}{2^B C + C_P} V_{ref}$$

- Sensitive to parasitic capacitor @ output
  - – If C_p constant → gain error
  - – If C_p voltage dependant → DAC nonlinearity
- Large area of caps for high DAC resolution (10bit DAC ratio 1:512)
- Monotonicity depends on element matching (more later)

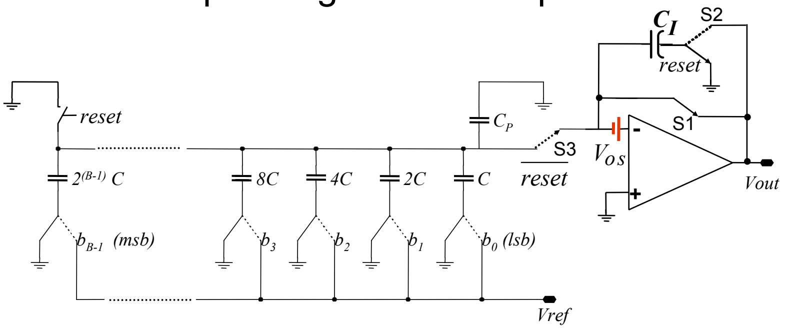
Parasitic Insensitive Charge Scaling DAC



$$V_{out} = -\frac{\sum_{i=0}^{B-1} b_i 2^i C}{C_I} V_{ref} \;, \quad C_I = 2^B C \;\rightarrow V_{out} = -\frac{\sum_{i=0}^{B-1} b_i 2^i}{2^B} V_{ref}$$

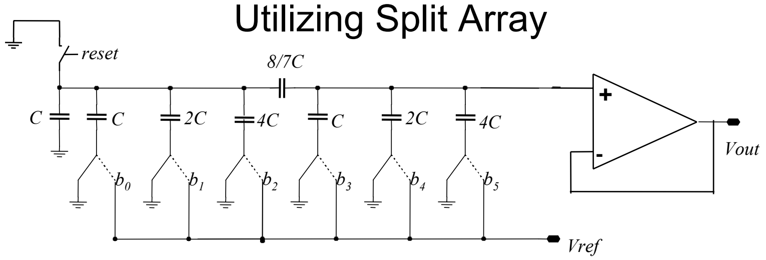
- Opamp helps eliminate the parasitic capacitor effect by producing virtual ground at the sensitive node since C_P has zero volts at start & end
  - – Issue: opamp offset & speed

Charge Scaling DAC
Incorporating Offset Compensation



- During reset phase:
  - Opamp disconnected from capacitor array via switch S3
  - Opamp connected in unity-gain configuration (S1)
  - C_I Bottom plate connected to ground (S2)
  - $V_{out} \sim - V_{os} \rightarrow V_{CI} = -V_{os}$
- This effectively compensates for offset during normal phase

Charge Scaling DAC
Utilizing Split Array



$$C_{series} = \frac{\sum all\, LSB\, array\, C}{\sum all\, MSB\, array\, C} C$$

- Split array→ reduce the total area of the capacitors required for high resolution DACs
  - E.g. 10bit regular binary array requires 1024 unit Cs while split array (5&5) needs 64 unit Cs
  - Issue: Sensitive to parasitic capacitor

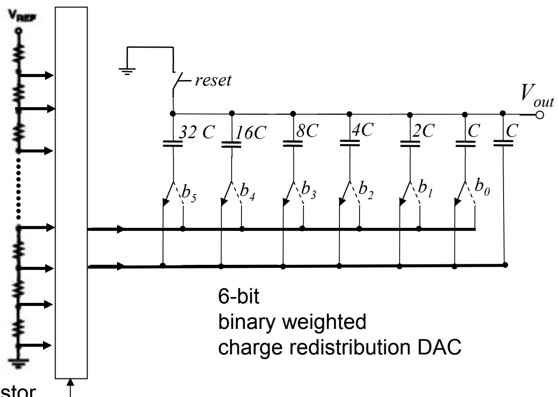
Charge Scaling DAC

- Advantages:
  - Low power dissipation → capacitor array does not dissipate DC power
  - Output is sample and held → no need for additional S/H
  - INL function of capacitor ratio
  - Possible to trim or calibrate for improved INL
  - Offset cancellation almost for free
- Disadvantages:
  - Process needs to include good capacitive material → not compatible with standard digital process
  - Requires large capacitor ratios
  - Not inherently monotonic (more later)

Segmented DAC

Resistor Ladder (MSB) & Binary Weighted Charge Scaling (LSB)

- Example: 12bit DAC
  - 6-bit MSB DAC→ R- string
  - 6-bit LSB DAC → binary weighted charge scaling
- Complexity much lower compared to full R-string
  - Full R string→ 4096 resistors
  - Segmented → 64 R + 7 Cs (64 unit caps)



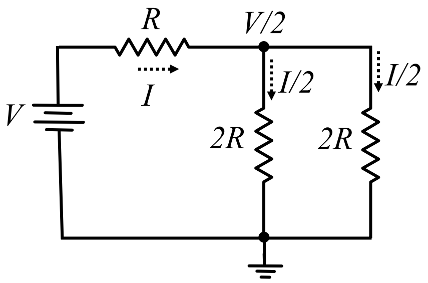
6bit resistor ladder

Switch Network

6-bit binary weighted charge redistribution DAC

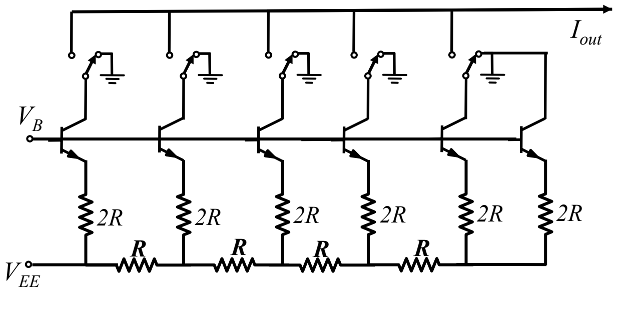
Current Based DACs
R-2R Ladder Type

- R-2R DAC basics:
  - Simple R network divides both voltage & current by 2



Increase # of bits by replicating circuit

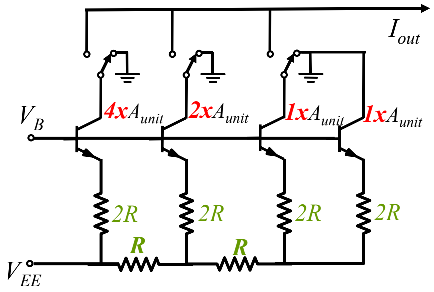
R-2R Ladder DAC



Emitter-follower added to convert to high output impedance current sources

R-2R Ladder DAC
How Does it Work?

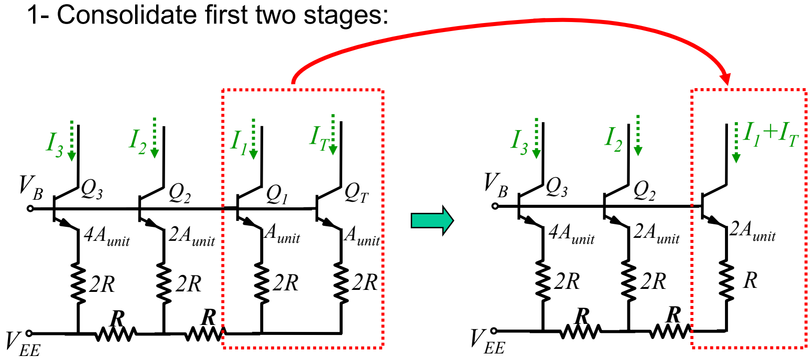
Consider a simple 3bit R-2R DAC:



R-2R Ladder DAC
How Does it Work?

Simple 3bit DAC:

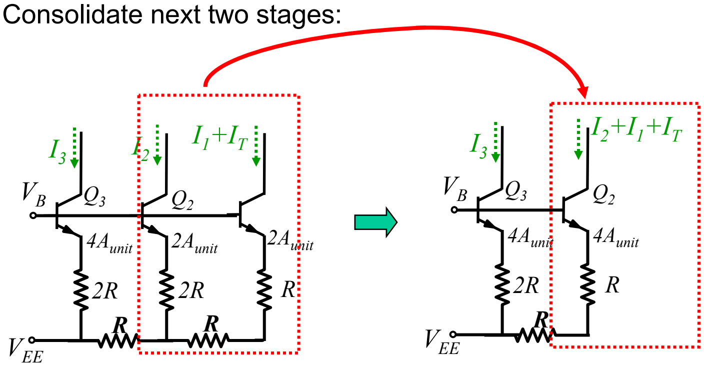
1- Consolidate first two stages:



R-2R Ladder DAC
How Does it Work?

Simple 3bit DAC-

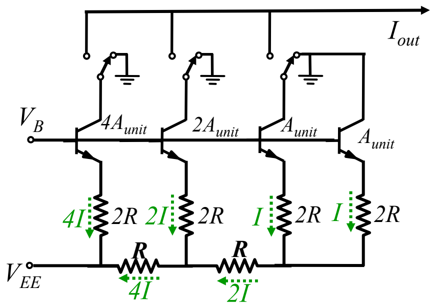
2- Consolidate next two stages:



$$I_3 = I_2 + I_1 + I_T \rightarrow I_3 = \frac{I_{Total}}{2}, I_2 = \frac{I_{Total}}{4}, I_1 = \frac{I_{Total}}{8}$$

R-2R Ladder DAC
How Does it Work?

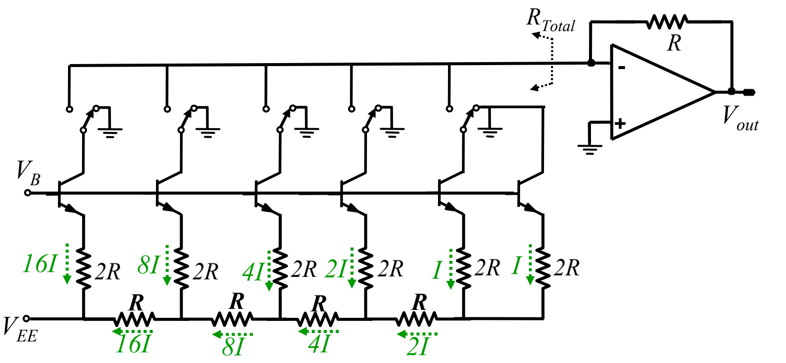
Consider a simple 3bit R-2R DAC:



In most cases need to convert output current to voltage

Ref: B. Razavi, "*Data Conversion System Design*", IEEE Press, 1995, page 84-87

R-2R Ladder DAC



Trans-resistance amplifier added to:

- Convert current to voltage
- Generate virtual ground @ current summing node so that output impedance of current sources do not cause error
- Issue: error due to opamp offset

R-2R Ladder DAC
Opamp Offset Issue

$$V_{os}^{out} = V_{os}^{in}\left(1+\frac{R}{R_{Total}}\right)$$

If $R_{Total} = large,$

$\rightarrow V_{os}^{out} \approx V_{os}^{in}$

If $R_{Total} = not\ \ large$

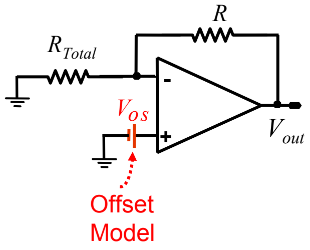
$\rightarrow V_{os}^{out} = V_{os}^{in}\left(1+\frac{R}{R_{Total}}\right)$

Problem:

Since R_{Total} *is code dependant*

$\rightarrow V_{os}^{out}$ *would be code dependant*

$\rightarrow$ *Gives rise to INL & DNL*

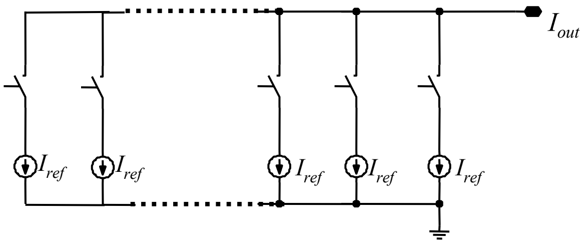


R-2R Ladder Summary

- Advantages:
  – Resistor ratios only x2
  – Does not require precision capacitors

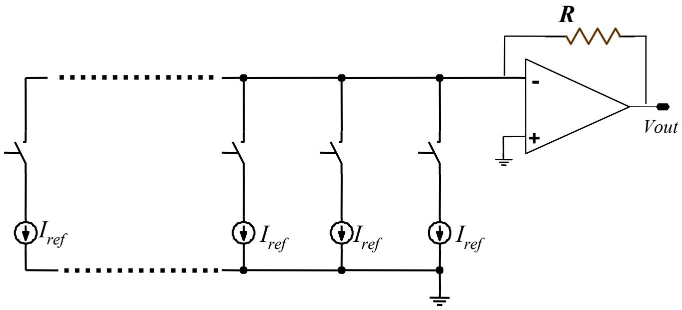
- Disadvantages:
  – Total device emitter area → $A_E^{unit} x\ 2^B$
    → Not practical for high resolution DACs
  – INL/DNL error due to amplifier offset

Current based DAC
Unit Element Current Source DAC



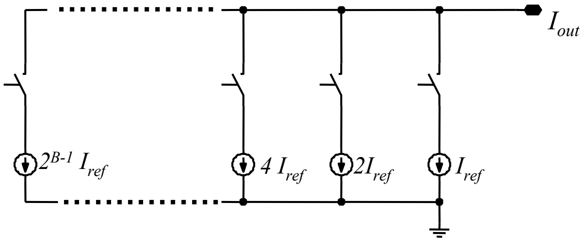
- "Unit elements" or thermometer
- 2^B-1 current sources & switches
- Suited for both MOS and BJT technologies
- Monotonicity does not depend on element matching and is guaranteed
- Output resistance of current source → gain error
  – Cascode type current sources higher output resistance → less gain error

Current Source DAC
Unit Element



- Output resistance of current source → gain error problem
  → Use transresistance amplifier
  - Current source output held @ virtual ground
  - Error due to current source output resistance eliminated
  - New issues: offset & speed of the amplifier

Current Source DAC
Binary Weighted



- "Binary weighted"
- B current sources & switches (2^B-1 unit current sources but less # of switches)
- Monotonicity depends on element matching →not guaranteed

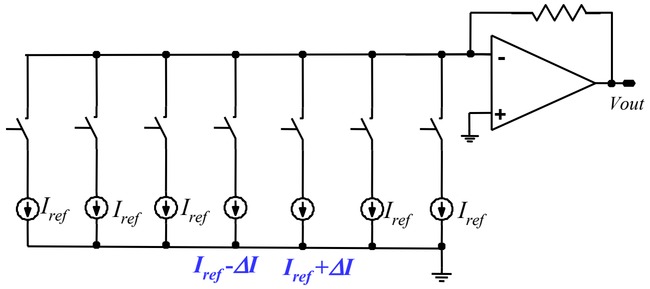
Static DAC Errors -INL / DNL

Static DAC errors mainly due to component mismatch

- Systematic errors
  - Contact resistance
  - Edge effects in capacitor arrays
  - Process gradients
  - Finite current source output resistance
- Random variations
  - Lithography etc…
  - Often Gaussian distribution (central limit theorem)

*Ref: C. Conroy et al, “Statistical Design Techniques for D/A Converters,” JSSC Aug. 1989, pp. 1118-28.

Current Source DAC DNL/INL Due to Element Mismatch



- Simplified example:
  - 3-bit DAC
  - Assume only two of the current sources mismatched (# 4 & #5)

Current Source DAC
DNL/INL Due to Element Mismatch

$$DNL[m] = \frac{segment[m] - V[LSB]}{V[LSB]}$$

$$DNL[4] = \frac{segment[4] - V[LSB]}{V[LSB]}$$

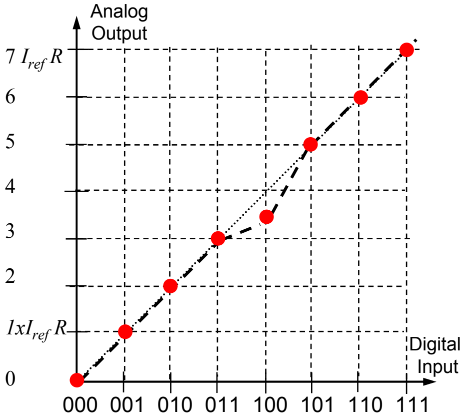
$$= \frac{(I - \Delta I)R - IR}{IR}$$

$$\boxed{DNL[4] = -\Delta I / I\ [LSB]}$$

$$DNL[5]\ = \frac{(I + \Delta I)R - IR}{IR}$$

$$\boxed{DNL[5]\ = \Delta I / I[LSB]}$$

$$\boxed{\rightarrow INL_{max} = -\Delta I / I\ [LSB]}$$



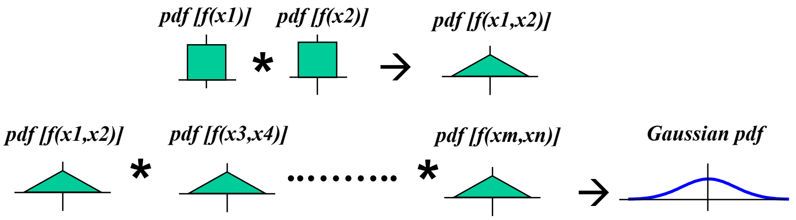
Component Mismatch
Probability Distribution Function

- **Component parameters → Random variables**

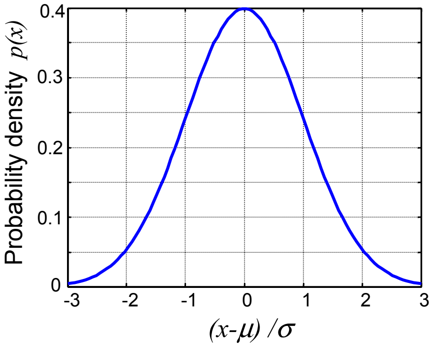
- **Each component is the product of many fabrication steps**
- **Most fabrication steps includes random variations**

→Overall component variations product of several random variables

→Assuming each of these variables have a uniform pdf distribution:
→Joint pdf of a random variable affected by two uniformly distributed variables → convolution of the two uniform pdfs…….



Gaussian Distribution



$$p(x)=\frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

where:
μ *is the expected value and*
standard deviation: $\sigma=\sqrt{E(X^2)-\mu^2}$
$\sigma^2 \rightarrow$ *variance*

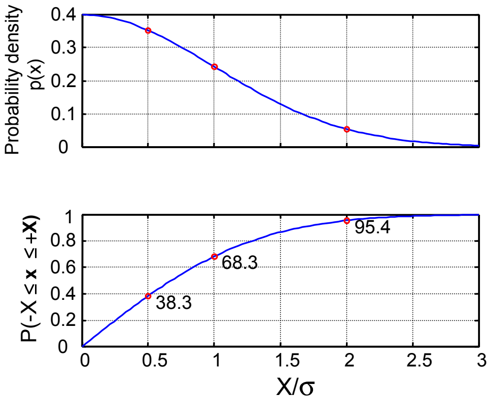
Yield

In most cases we are interested in finding the percentage of components (e.g. R) falling within certain bounds:

$$P(-X \le x \le +X)=$$

$$=\frac{1}{\sqrt{2\pi}}\int_{-X}^{+X} e^{-\frac{x^2}{2}}dx$$

$$=erf\left(\frac{X}{\sqrt{2}}\right)$$



Yield

X/σ	$P(-X \le x \le X)$ [%]	X/σ	$P(-X \le x \le X)$ [%]
0.2000	15.8519	2.2000	97.2193
0.4000	31.0843	2.4000	98.3605
0.6000	45.1494	2.6000	99.0678
0.8000	57.6289	2.8000	99.4890
1.0000	68.2689	3.0000	99.7300
1.2000	76.9861	3.2000	99.8626
1.4000	83.8487	3.4000	99.9326
1.6000	89.0401	3.6000	99.9682
1.8000	92.8139	3.8000	99.9855
2.0000	95.4500	4.0000	99.9937

Example

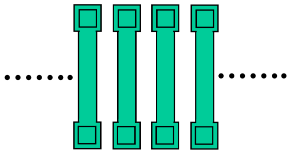
- Measurements show that the offset voltage of a batch of operational amplifiers follows a Gaussian distribution with σ = 2mV and μ = 0.

- Find the fraction of opamps with $|V_{os}| <$ 6mV:
  - $X/\sigma = 3$ → 99.73 % yield

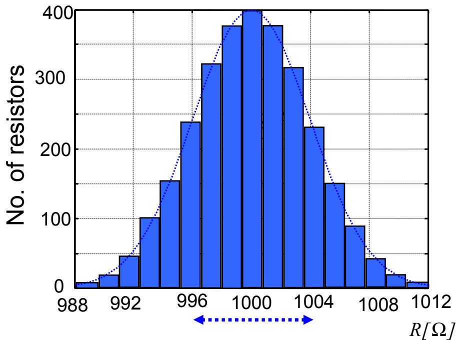
- Fraction of opamps with $|V_{os}| <$ 400μV:
  - $X/\sigma = 0.2$ → 15.85 % yield

Component Mismatch

Example: Resistors layouted out side-by-side



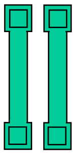
After fabrication large # of devices measured & graphed → typically if sample size large shape is Gaussian



E.g. Let us assume in this example 1000 Rs measured & 68.5% fall within +-4OHM or +-0.4% of average
→ 1σ for resistors→ 0.4%

Component Mismatch

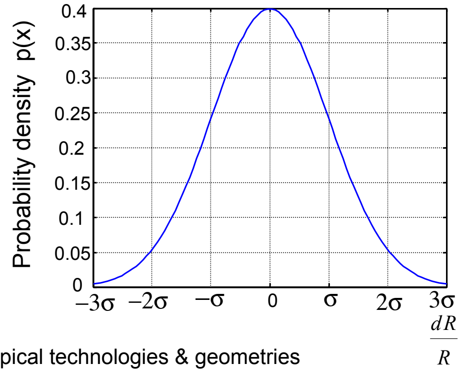
Example: Two resistors layouted out side-by-side



$$R = \frac{R_1 + R_2}{2}$$

$$dR = R_1 - R_2$$

$$\sigma^2_{\frac{dR}{R}} \propto \frac{1}{Area}$$



For typical technologies & geometries
1σ for resistors → *0.02 to 5%*

In the case of resistors σ is a function of area

DNL Unit Element DAC

E.g. Resistor string DAC:
Assumption: No systematic error- only random error

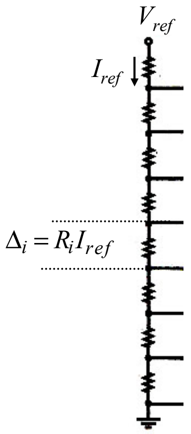
$$\Delta = R_{median}\, I_{ref} \quad where \quad R_{median} = \frac{\sum_{o}^{2^B-1} R_i}{2^B}$$

$$\Delta_i = R_i I_{ref}$$

$$DNL_i = \frac{\Delta_i - \Delta_{median}}{\Delta_{median}}$$

$$= \frac{R_i - R_{median}}{R_{median}} = \frac{dR}{R_{median}} \approx \frac{dR}{R_i}$$

$$\boxed{\sigma_{DNL} = \sigma_{\frac{dR_i}{R_i}}}$$



To first order → DNL of unit element DAC is independent of resolution!
Note: Similar results for other unit-element based DACs

DNL Unit Element DAC

E.g. Resistor string DAC:

$$\boxed{\sigma_{DNL} = \sigma_{\frac{dR_i}{R_i}}}$$

Example:

If $\sigma_{dR/R}$ = 0.4%, what DNL spec goes into the DAC datasheet so that 99.9% of all converters meet the spec?

Yield

X/σ	P(-X ≤ x ≤ X) [%]	X/σ	P(-X ≤ x ≤ X) [%]
0.2000	15.8519	2.2000	97.2193
0.4000	31.0843	2.4000	98.3605
0.6000	45.1494	2.6000	99.0678
0.8000	57.6289	2.8000	99.4890
1.0000	68.2689	3.0000	99.7300
1.2000	76.9861	3.2000	99.8626
1.4000	83.8487	3.4000	99.9326
1.6000	89.0401	3.6000	99.9682
1.8000	92.8139	3.8000	99.9855
2.0000	95.4500	4.0000	99.9937

DNL Unit Element DAC

E.g. Resistor string DAC:

$$\boxed{\sigma_{DNL} = \sigma_{\frac{dR_i}{R_i}}}$$

Example:

If $\sigma_{dR/R}$ = 0.4%, what DNL spec goes into the datasheet so that 99.9% of all converters meet the spec?

Answer:

From table: for 99.9%

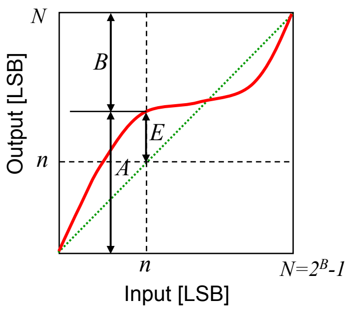
→ X/σ = 3.3

σ_{DNL} = $\sigma_{dR/R}$ = 0.4%

3.3 σ_{DNL} = 3.3x0.4%=1.3%

→DNL= +/- 0.013 LSB

DAC INL Analysis



	Ideal	Variance
$A=n+E$	n	$n.\sigma_\varepsilon^2$
$B=N-n-E$	$N-n$	$(N-n).\sigma_\varepsilon^2$

$E = A-n \qquad r = n/N \qquad N=A+B$

$= A-r(A+B)$

$= (1-r).A - r.B$

→ *Variance of E:*

$$\sigma_E^2 = (1-r)^2.\sigma_A^2 + r^2.\sigma_B^2$$

$$= N.r.(1-r).\sigma_\varepsilon^2 = n.(1-n/N).\sigma_\varepsilon^2$$

DAC INL

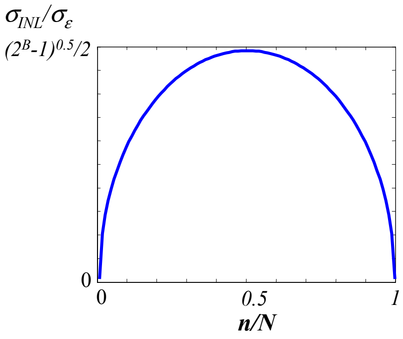
$$\sigma_E^2 = n\left(1-\frac{n}{N}\right)\times\sigma_\varepsilon^2$$

$$To\ find\ max.\ variance: \quad \frac{d\sigma_E^2}{dn}=0$$

$$\rightarrow n=N/2 \rightarrow \sigma_E^2 = \frac{N}{4}\times\sigma_\varepsilon^2$$

- Error is maximum at mid-scale (*N/2*):

$$\boxed{\begin{aligned}&\sigma_{INL}=\frac{1}{2}\sqrt{2^B-1}\,\sigma_\varepsilon\\&with\quad N=2^B-1\end{aligned}}$$



- INL depends on both DAC resolution & element matching σ_ε
- While $\sigma_{DNL} = \sigma_\varepsilon$ is to first order independent of DAC resolution and is only a function of element matching

Ref: Kuboki et al, TCAS, 6/1982

Untrimmed DAC INL

Example:

Assume the following requirement for a DAC:

$\sigma_{INL} = 0.1\ LSB$

Find maximum resolution for:

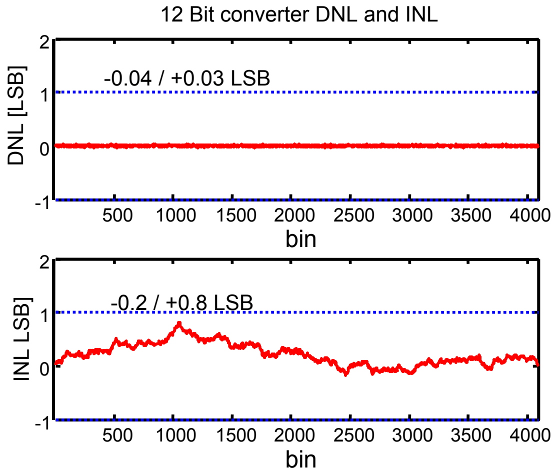
$\sigma_\varepsilon = 1\%$
$\sigma_\varepsilon = 0.5\%$
$\sigma_\varepsilon = 0.2\%$
$\sigma_\varepsilon = 0.1\%$

$$\sigma_{INL} \cong \frac{1}{2}\sqrt{2^B - 1}\ \sigma_\varepsilon$$

$$B \cong 2 + 2\log_2\left[\frac{\sigma_{INL}}{\sigma_\varepsilon}\right]$$

$$\sigma_\varepsilon = 1\% \rightarrow B_{max} = 8.6bits$$
$$\sigma_\varepsilon = 0.5\% \rightarrow B_{max} = 10.6bits$$
$$\sigma_\varepsilon = 0.2\% \rightarrow B_{max} = 13.3bits$$
$$\sigma_\varepsilon = 0.1\% \rightarrow B_{max} = 15.3bits$$

Simulation Example



σ_ε = 1%
B = 12
Random # generator used in MatLab

Computed INL:
σ_{INL}^{max} = 0.3 LSB (midscale)

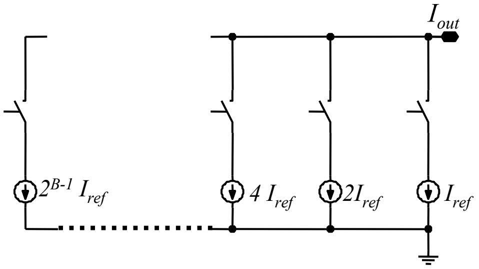
Why is the results not as expected per our derivation?

INL & DNL for Binary Weighted DAC

- INL same as for unit element DAC
- DNL depends on transition
  - Example:

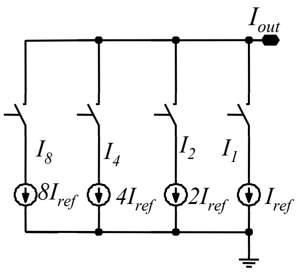
    0 to 1 → $\sigma_{DNL}^2 = \sigma_{(dI/I)}^{\ 2}$

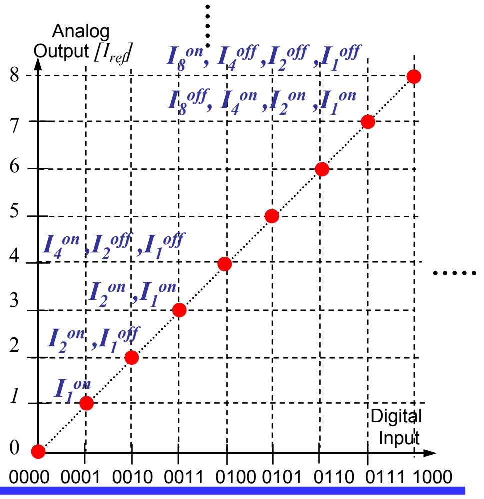
    1 to 2 → $\sigma_{DNL}^2 = 3\sigma_{(dI/I)}^{\ 2}$



- Consider MSB transition: 0111 … → 1000 …

DAC DNL
Example: 4bit DAC



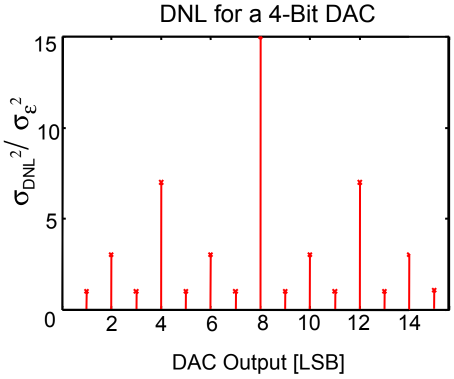


- DNL depends on transition
  - Example:

    0 to 1 → $\sigma_{DNL}^2 = \sigma_{(dIref/Iref)}^{\ 2}$

    1 to 2 → $\sigma_{DNL}^2 = 3\sigma_{(dIref/Iref)}^{\ 2}$

Binary Weighted DAC DNL



- Worst-case transition occurs at mid-scale:

$$\sigma_{DNL}^2 = \underbrace{\left(2^{B-1}-1\right)\sigma_\varepsilon^2}_{0111\ldots} + \underbrace{\left(2^{B-1}\right)\sigma_\varepsilon^2}_{1000\ldots}$$

$$\cong 2^B \sigma_\varepsilon^2$$

$$\sigma_{DNL_{max}} = 2^{B/2}\sigma_\varepsilon$$

$$\sigma_{INLmax} \cong \frac{1}{2}\sqrt{2^B-1}\;\sigma_\varepsilon \cong \frac{1}{2}\sigma_{DNL_{max}}$$

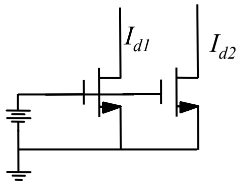
- Example:
  B = 12, σ_ε = 1%
  → σ_{DNL} = 0.64 LSB
  → σ_{INL} = 0.32 LSB

MOS Current Source Variations Due to Device Matching Effects

$$I_d = \frac{I_{d1}+I_{d2}}{2}$$

$$\frac{dI_d}{I_d} = \frac{I_{d1}-I_{d2}}{I_d}$$

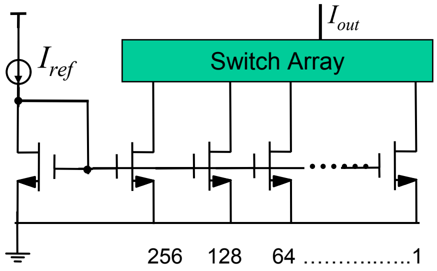
$$\frac{dI_d}{I_d} = \frac{d\,W/L}{W/L} + \frac{2\times dV_{th}}{V_{GS}-V_{th}}$$



- Current matching depends on:
  - Device W/L ratio matching
    → Larger device area less mismatch effect
  - Current mismatch due to threshold voltage variations:
    → Larger gate-overdrive less threshold voltage mismatch effect

Current-Switched DACs in CMOS

$$\frac{dI_d}{I_d} = \frac{d\,{}^{W}\!/_{L}}{{}^{W}\!/_{L}} + \frac{dV_{th}}{V_{GS}-V_{th}}$$



Example: 8bit Binary Weighted

- Advantages:
  Can be very fast
  Reasonable area for resolution < 9-10bits

- Disadvantages:
  Accuracy depends on device W/L & V_{th} matching

Unit Element versus Binary Weighted DAC

<u>Unit Element DAC</u>

$$\sigma_{DNL} = \sigma_{\varepsilon}$$

$$\sigma_{INL} \cong 2^{{}^{B}\!/_{2}-1}\,\sigma_{\varepsilon}$$

<u>Binary Weighted DAC</u>

$$\sigma_{DNL} \cong 2^{{}^{B}\!/_{2}}\,\sigma_{\varepsilon} = 2\sigma_{INL}$$

$$\sigma_{INL} \cong 2^{{}^{B}\!/_{2}-1}\,\sigma_{\varepsilon}$$

Number of switched elements:

$$S = 2^{B}$$

$$S = B$$

Key point: Significant difference in performance and complexity!

Unit Element versus Binary Weighted DAC Example: B=10

Unit Element DAC

$$\sigma_{DNL} = \sigma_{\varepsilon}$$

$$\sigma_{INL} \cong 2^{B/2-1}\sigma_{\varepsilon} = 16\sigma_{\varepsilon}$$

Binary Weighted DAC

$$\sigma_{DNL} \cong 2^{B/2}\sigma_{\varepsilon} = 32\sigma_{\varepsilon}$$

$$\sigma_{INL} \cong 2^{B/2-1}\sigma_{\varepsilon} = 16\sigma_{\varepsilon}$$

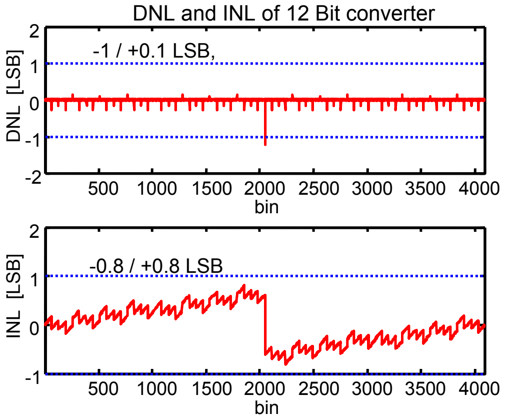
Number of switched elements:

$$S = 2^{B} = 1024$$

$$S = B = 10$$

Significant difference in performance and complexity!

“Another” Random Run …

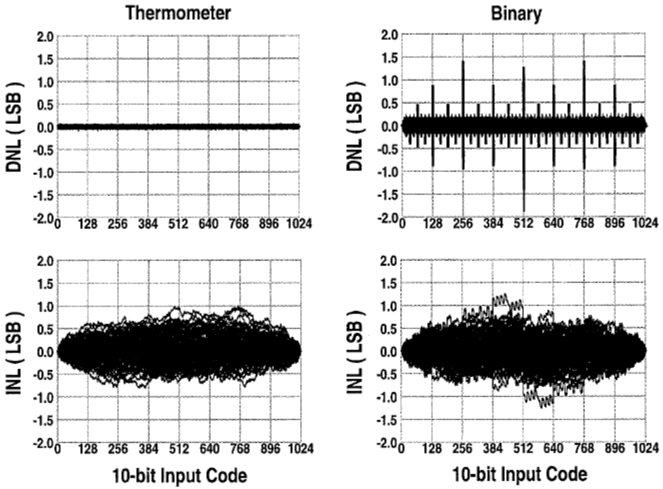


Now (by chance) worst DNL is mid-scale.

Statistical result!

10Bit DAC DNL/INL Comparison Plots: 100 Simulation Runs Overlaid

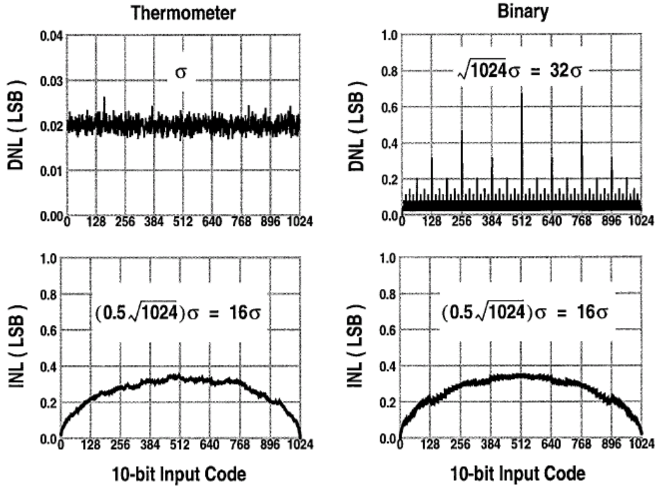
Ref: C. Lin and K. Bult, "A 10-b, 500-MSample/s CMOS DAC in 0.6 mm2," *IEEE Journal of Solid-State Circuits,* vol. 33, pp. 1948 - 1958, December 1998.



Note: σ_ε=2%

10Bit DAC DNL/INL Comparison Plots: RMS for 100 Simulation Runs

Ref: C. Lin and K. Bult, "A 10-b, 500-MSample/s CMOS DAC in 0.6 mm2," *IEEE Journal of Solid-State Circuits,* vol. 33, pp. 1948 - 1958, December 1998.



Note: σ_ε=2%

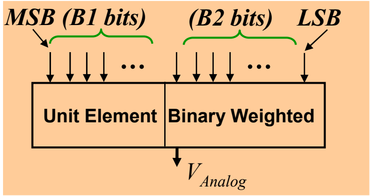
DAC INL/DNL Summary

- DAC choice of architecture has significant impact on DNL
- INL is independent of DAC architecture and requires element matching commensurate with overall DAC precision
- Results assume uncorrelated random element variations
- Systematic errors and correlations are usually also important and may affect final DAC performance

Ref: Kuboki, S.; Kato, K.; Miyakawa, N.; Matsubara, K. Nonlinearity analysis of resistor string A/D converters. IEEE Transactions on Circuits and Systems, vol.CAS-29, (no.6), June 1982. p.383-9.

Segmented DAC

- Objective:
  Compromise between unit element and binary weighted DAC



$B_{Total} = B_1+B_2$

- Approach:
  B_1 MSB bits → unit elements
  B_2 LSB bits → binary weighted
- INL: unaffected same as either architecture
- DNL: Worst case occurs when LSB DAC turns off and one more MSB DAC element turns on → Same as binary weighted DAC with (B_2+1) # of bits
- Number of switched elements: $(2^{B1}-1) + B_2$

Comparison

Example:

$$B = 12, \quad B_1 = 5, \quad B_2 = 7$$

$$\underbrace{B_1 = 6,}_{\text{MSB}} \quad \underbrace{B_2 = 6}_{\text{LSB}}$$

$$\sigma_{DNL} \cong 2^{(B_2+1)/2}\sigma_{\varepsilon} = 2\sigma_{INL}$$

$$\sigma_{INL} \cong 2^{B/2-1}\sigma_{\varepsilon}$$

$$S = 2^{B1} - 1 + B_2$$

Assuming: $\sigma_{\varepsilon} = 1\%$

DAC Architecture (B1+B2)	$\sigma_{INL[LSB]}$	$\sigma_{DNL[LSB]}$	# of switched elements
Unit element (12+0)	*0.32*	*0.01*	*4095*
Segmented (6+6)	*0.32*	*0.113*	*63+6=69*
Segmented (5+7)	*0.32*	*0.16*	*31+7=38*
Binary weighted(0+12)	*0.32*	*0.64*	*12*