Dynamic Matching and Mismatch Shaping

Gabor Temes, Oregon State University

DATA CONVERTERS

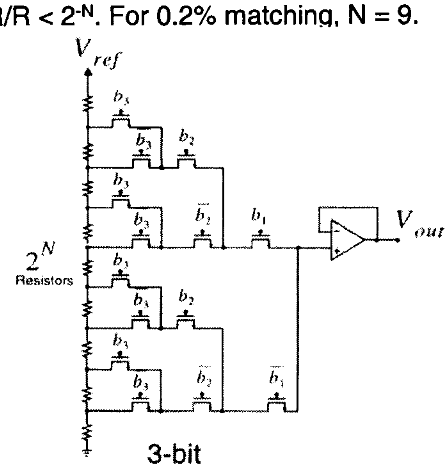
Typical A/D and D/A converters rely on the accurate matching of analog elements and/or ideal amplifier performance.
Examples:

- R-STRING DAC
- CURRENT-MODE DAC
- SC DAC
- FLASH ADC
- CHARGE-REDISTRIBUTION ADC

DATA CONVERTERS

RESISTOR-STRING DAC

Monotonic, but for N bits an INL = 0.5 LSB requires a matching accuracy of $\Delta R/R < 2^{-N}$. For 0.2% matching, N = 9.

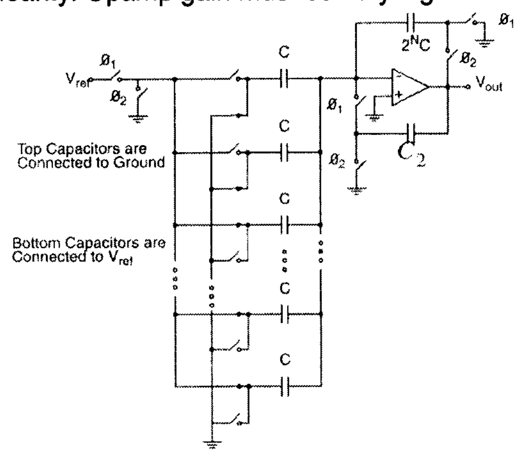


3-bit

DATA CONVERTERS

SWITCHED-CAPACITOR DAC

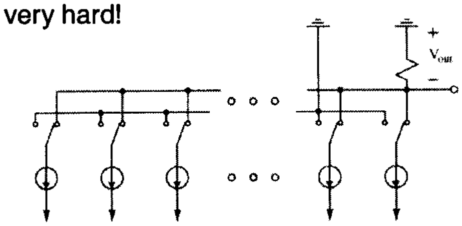
Monotonic, but input caps must be matched so that $\Delta C/C < 2^{-N-1}$ for sufficient linearity. Opamp gain must be very high for large N!

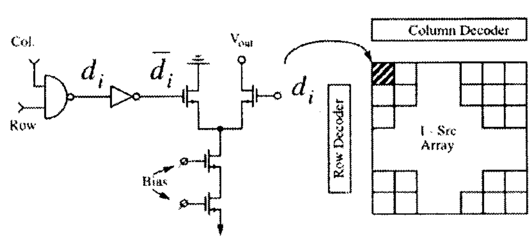


DATA CONVERTERS

CURRENT-MODE DAC

Current sources must be matched to 2^{-N-1} relative accuracy. 10 bits possible; 12 bits very hard!

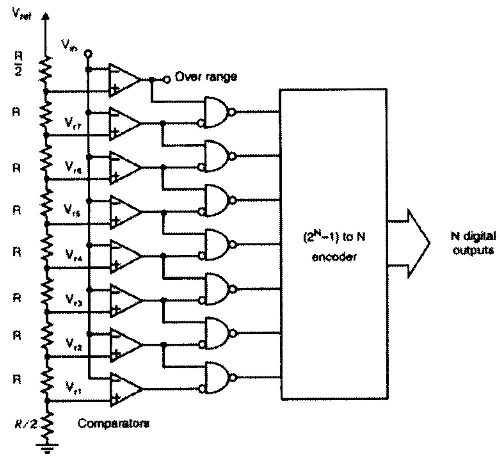




DATA CONVERTERS

FLASH ADC

$\Delta R/R < 2^{-N}$ required for N-bit resolution.

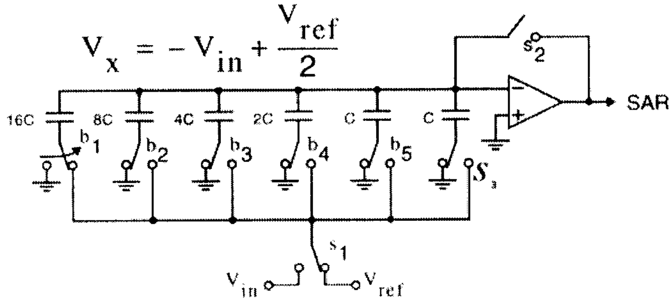


DATA CONVERTERS

CHARGE-REDISTRIBUTION ADC

For INL = 0.5 LSB, $\Delta C/C < 2^{-N}$ required. Limited to 11-12 bits.

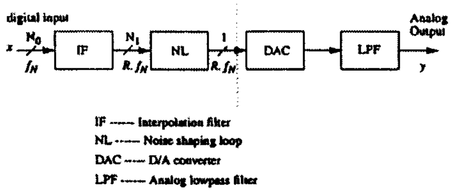
$$V_x = -V_{in} + \frac{V_{ref}}{2}$$

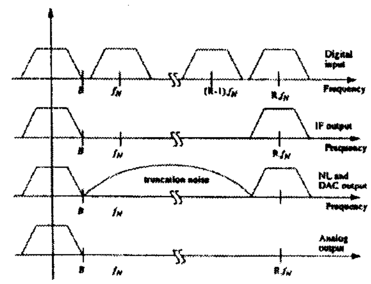


DATA CONVERTERS

DELTA-SIGMA DACS

They use digital preprocessing (oversampling & noise-shaped truncation) to produce a fast, low-resolution (often 1-bit) input signal for the actual DAC:



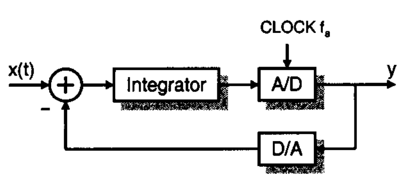


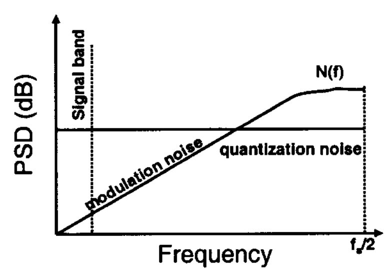
1-bit DAC is (ideally) absolutely linear! Analog postfilter difficult.

DATA CONVERTERS

DELTA-SIGMA ADCS

Analog loop generates oversampled noise-shaped low-resolution (often 1-bit) digital output, allows linear DAC operation:



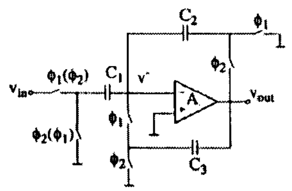


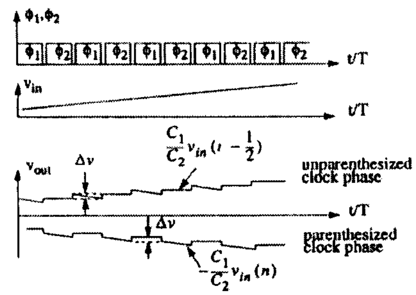
Loop design difficult, but ENOB > 20 has been achieved. Covered in *Delta-Sigma Data Converters*, S. Norsworthy, R. Schreier, G. Temes, Eds., IEEE Press, 1997.

CORRECTION TECHNIQUES

CORRELATED DOUBLE SAMPLING

Noise (1/f, offset, vestigial signal) is stored and subtracted.
Example: SC amplifier:



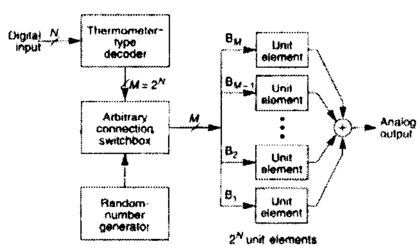


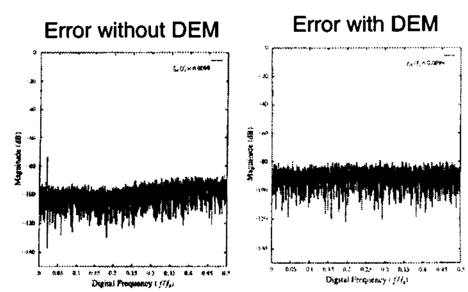
During reset ($\Phi_1 = 1$) C_1 & C_2 store virtual ground voltage. Result: offset cancelled, 1/f noise and input signal high-pass filtered, gain boosted, THD reduced.

CORRECTION TECHNIQUES

DYNAMIC MATCHING (ELEMENT SCRAMBLING)

Static nonlinearity causes distortion. To decorrelate the DAC error from the signal, dithering can be used. It occupies some of the signal range, and hence reduces the dynamic range. Better: randomize the unit element selection.



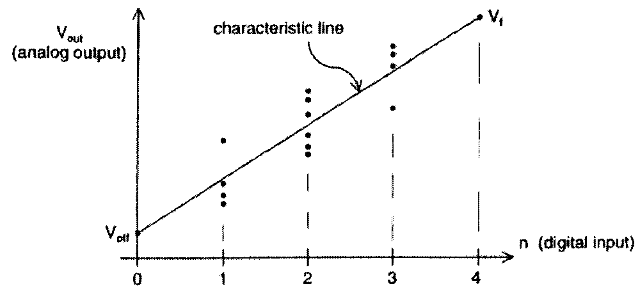


Harmonics converted to random noise. (0.1% DAC error assumed.)

SPECTRAL ERROR SHAPING

Instead of correcting or cancelling the error signal generated by mismatch, it can be filtered out of the signal band, just as the quantization error is in the delta-sigma converter.

Example: 4-level SC DAC with input capacitor mismatch. If every choice of the input capacitors corresponding to each code is plotted, the result may be as shown:



Note: The average of outputs falls on the characteristic line. Hence, if each combination is used equally often, the average codes will give a linear operation.

SPECTRAL ERROR SHAPING

Simple techniques:

- **Barrel shifting** - start each new conversion one index higher;
- **Individual level averaging** - keep track of all past conversions, make average usage of each element the same;
- **Data-weighted averaging** - start where the previous conversion ended. Wrap-around after the last C_{IN} was used.

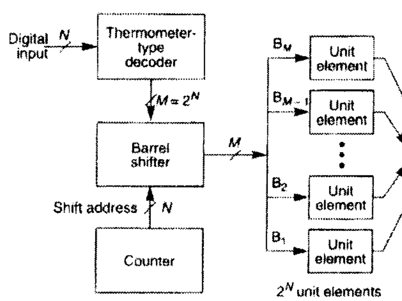
This introduces a $(z - 1)$ highpass factor into the noise transfer function $\Rightarrow$ first-order noise shaping.

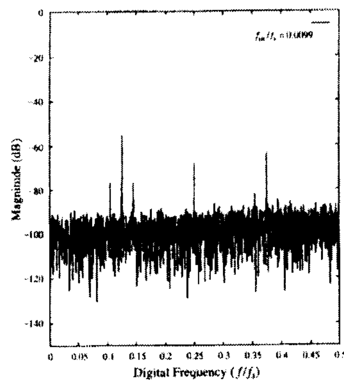
To achieve second-order noise shaping, M second-order digital delta-sigma loops may be used to derive the element selection sequence.

SPECTRAL ERROR SHAPING

BARREL SHIFTING ALGORITHM

Modulates the mismatch noise. Can be combined with randomization. Will generate tones due to mixing of noise, signal and modulation carrier.

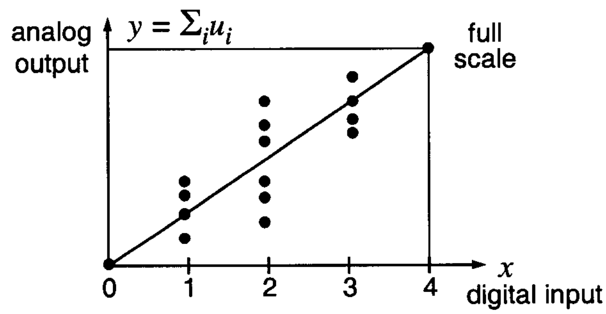




SPECTRAL ERROR SHAPING

INDIVIDUAL LEVEL AVERAGING

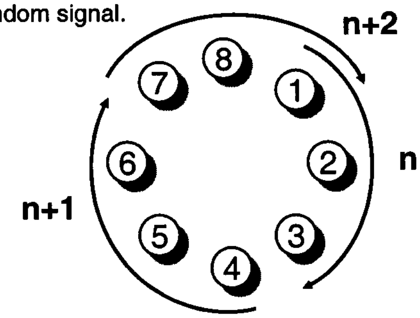
Since the mean of all possible outputs for a given input code falls on the ideal curve, by keeping track of previous element choices for each code and selecting them sequentially, the average (ideal result) will be approached eventually, but after a longer time than in DWA. Various algorithms are available to implement it.



SPECTRAL ERROR SHAPING

DATA-WEIGHTED AVERAGING

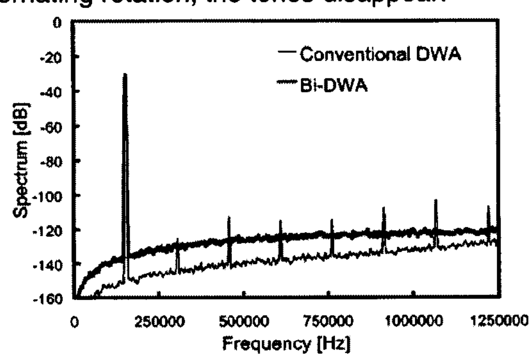
The elements are selected sequentially, so the difference in the # of usage is < 1. Assuming that the sum of all errors δ_i is zero, the max. accumulated error $s(p)$ after p periods is just the sum of all positive or negative errors among the elements. $s(p)$ is strictly bounded at a relatively small value M. The average value is $M/p \rightarrow 0$, as $p \rightarrow \infty$. The error made in the n[th] period is $e(n) = s(n) - s(n-1)$, hence the error power spectral density (PSD) is $PSD_e = |1 - z^{-1}|^2 \cdot PSD_s$. If the input is busy and random, $s(n)$ is a bounded random white noise, and hence $e(n)$ will be a lowpass-filtered (shaped) random signal.



SPECTRAL ERROR SHAPING

TONE GENERATION IN DWA

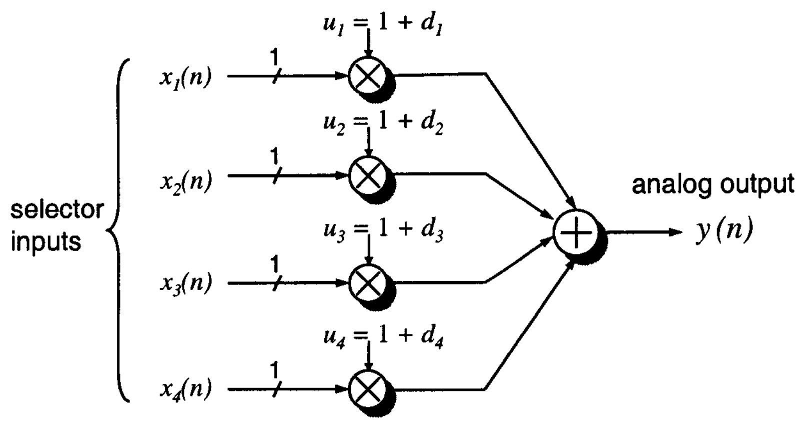
If the input is DC or varies slowly, and there is a rational relation between its value and the # of elements, tones will be generated in $e(n)$. They can be eliminated by introducing a random component into the element selection logic. This is similar to dither, but doesn't reduce the dynamic range. For alternating rotation, the tones disappear:



SPECTRAL ERROR SHAPING

DELTA-SIGMA ELEMENT SELECTION SCHEME I

5-level DAC



SPECTRAL ERROR SHAPING

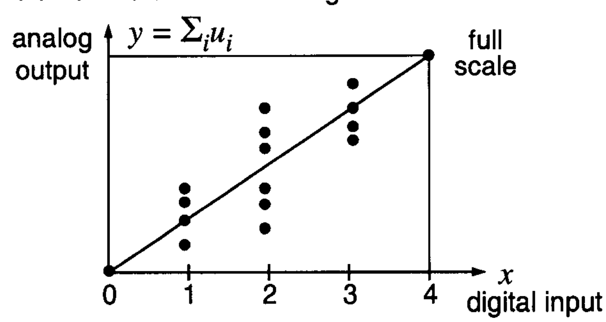
DELTA-SIGMA ELEMENT SELECTION SCHEME II

Assumption

We accept the full-scale DAC output as accurate.
Then:

$$d_1 + d_2 + \ldots + d_N = 0$$

and the ideal element value is the average of all element values.



Note the symmetry!

SPECTRAL ERROR SHAPING

DELTA-SIGMA ELEMENT SELECTION SCHEME III

Conditions on the Selector Sequences

1. All $x_i(n)$ values are 0 or 1
2. $\Sigma_i x_i(n) = v(n) =$ digital output of the $\Delta\Sigma$ loop
3. Each input can be written as:

$$x_i(n) = f(n) + h(n) * e_i(n)\ ,$$

where $f(n)$ is a bounded function;
$Z[h(n)] = H(z) = (1 - z^{-1})^m$, a high-pass filter function;
and $e_i(n)$ is a bounded pseudo-random signal.
Note that $f(n)$ and $h(n)$ are the same for all x_i, and the order m is 1 or 2.

SPECTRAL ERROR SHAPING

DELTA-SIGMA ELEMENT SELECTION SCHEME IV

The Shaped DAC Output Signal

Clearly

$$y(n) = \Sigma_i\, x_i(n)(1+d_i)$$
$$y(n) = v(n) + \Sigma_i\, [f(n) + h(n) * e_i(n)]\, d_i$$
$$y(n) = v(n) + 0 + h(n) * \Sigma_i\, d_i\, e_i(n)$$

Thus, the DAC error is high-pass filtered, as required for mismatch shaping.

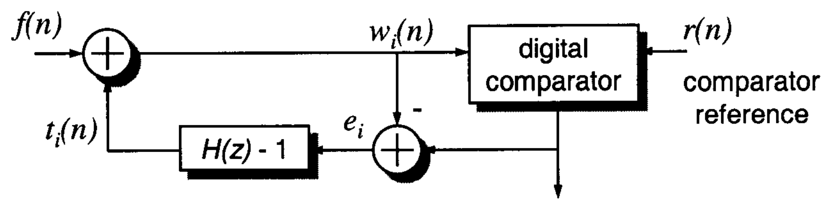
SPECTRAL ERROR SHAPING

DELTA-SIGMA ELEMENT SELECTION SCHEME V

The Generation of the Selector Sequences

Each $x_i(n)$ can be obtained as the output of a digital ΔΣ loop, with an input $f(n)$ and truncation error $e_i(n)$.
Using the error-feedback configuration, the i th loop is of the form:

$$x_i = \begin{cases} 1 \\ 0 \end{cases} \text{if } w_i(n) \begin{matrix} \geq \\ < \end{matrix} r(n)$$



$$x_i(n) = f(n) + h(n) * e_i(n)$$

Gábor C. Temes — *23/46* — *Oregon State University*

SPECTRAL ERROR SHAPING

DELTA-SIGMA ELEMENT SELECTION SCHEME VI

The Choice of $r(n)$ and $f(n)$

In each step, $r(n)$ is adjusted so that $v(n)$ of the $x_i(n)$ will be 1, the rest 0. In practice, the x_i outputs of $v(n)$ loops with the largest $w_i(n)$ will be simply set to 1, the rest to 0. This will minimize $\Sigma_i\, x_i(n)e_i^2(n)$, and thus the mismatch noise power.

Also, choosing $f(n) = -\, min_i\, [t_i(n)]$ will minimize $\Sigma_i w_i^2(n)$ and will keep it bounded.

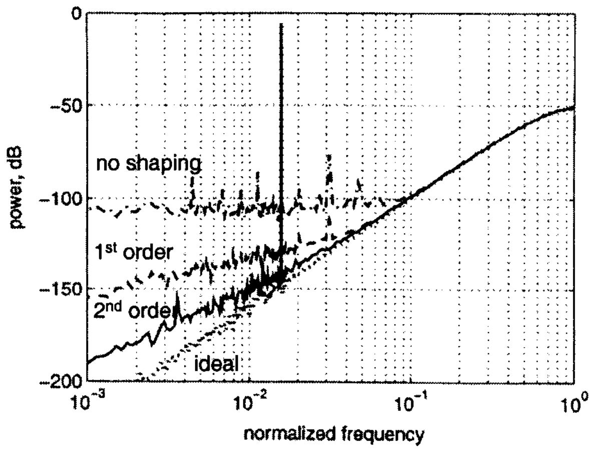
The system can also be used as the feedback DAC of a ΔΣ ADC loop.

Gábor C. Temes — *24/46* — *Oregon State University*

SPECTRAL ERROR SHAPING

DELTA-SIGMA ELEMENT SELECTION SCHEME VII

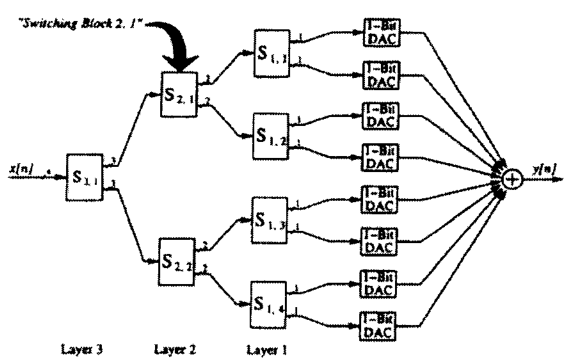
Example: Third-Order ΔΣ ADC



SPECTRAL ERROR SHAPING

TREE STRUCTURE SCHEME I

Number Conservation Rule



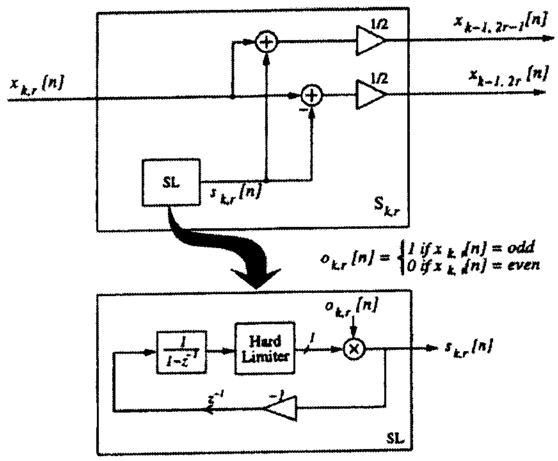
Each switching block satisfies the *number conservation rule*:

1. The SB output ∈ {0, 1, …, 2^{k-1}}, where k is the layer number
2. The sum of the SB's outputs equals the SB's input

SPECTRAL ERROR SHAPING

TREE STRUCTURE SCHEME II

1st-order Switching Block DSP

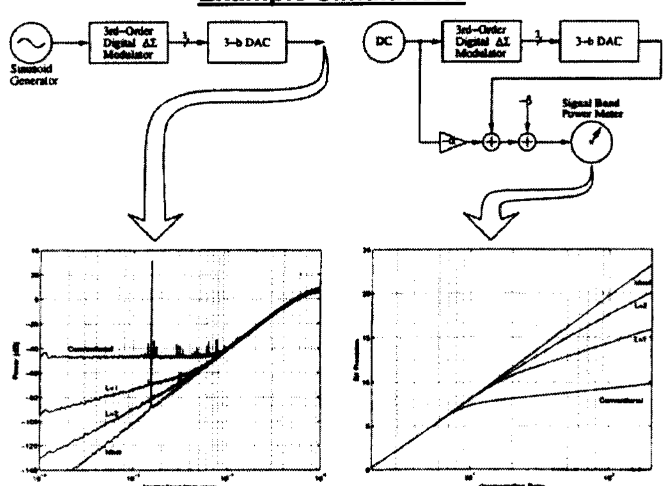


Gábor C. Temes | 27/46 | Oregon State University

SPECTRAL ERROR SHAPING

TREE STRUCTURE SCHEME III

Example Simulations



Gábor C. Temes | 28/46 | Oregon State University

OTHER CORRECTION TECHNIQUES

For making matching suitable for ENOB > 10 bits.

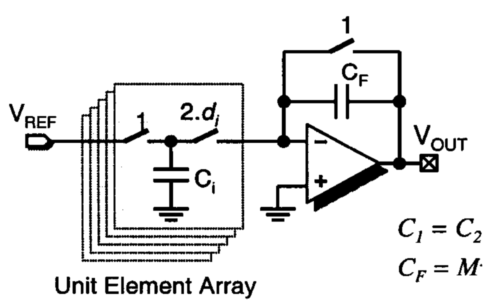
Techniques:

1. Digitally Controlled Analog Correction
2. Digital Correction
3. Digitally Controlled Analog Error Cancellation
4. Spectral Error Filtering

OTHER CORRECTION TECHNIQUES

ANALOG CORRECTION I

M-level switched-capacitor DAC with thermometer-coded input:



$C_1 = C_2 = \ldots = C_M = C$

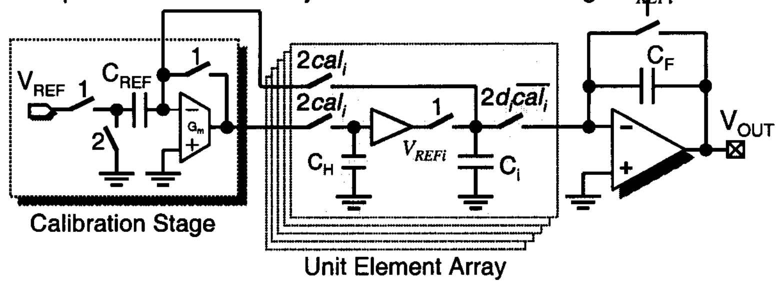
$C_F = M \cdot C$

Unit-element charge input; less sensitive to mismatch, better DNL, guaranteed monotonicity, reduced glitch noise. Still, hard to get over 10-bit linearity and accuracy.

OTHER CORRECTION TECHNIQUES

ANALOG CORRECTION II

To improve accuracy & linearity, provide a "fine" adjustment of the charge in each input branch via an adjustable reference voltage V_{REFi}:

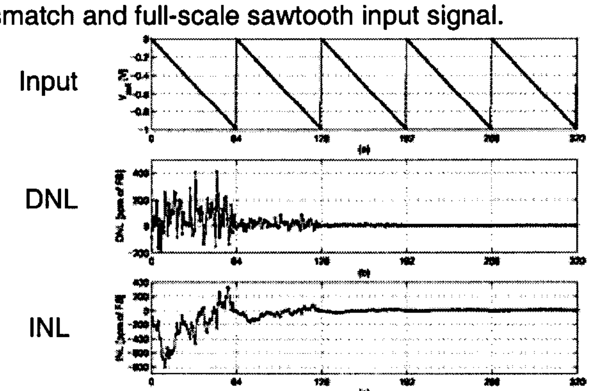


The calibration stage compares the stored charge in each branch with the desired exact charge $V_{REF} \cdot C_{REF}$, iteratively corrects charge error. The process is insensitive to the offset of the G_M and charge injection mismatch. For M nonzero levels, $M+1$ branches are needed. The opamp needs CDS to eliminate the error signals at its input.

OTHER CORRECTION TECHNIQUES

ANALOG CORRECTION III

Example: 6-bit (64-level) DAC, with 1% rms random capacitance mismatch and full-scale sawtooth input signal.



Without correction (first cycle), INL = 1.65% of LSB, DNL = 0.83% of LSB. After convergence (4th cycle) INL = 0.024% of LSB, DNL = 0.0046% of LSB ≅ 1.6 ppm ≈ 19 ENOB.