4. ERROR CORRECTION IN D/A CONVERTERS

Continuous-time vs. sampled-output DACs

Layout techniques
- Matching
- Systematic errors

Static error correction
- Static nonlinearities
- Cancelling tones in test generators
- Trimming
- Dynamic element matching techniques

Dynamic errors
- Slew rate
- Switching timing errors
- Pattern jitter
- Time-interleaving

DIFFERENT APPLICATIONS OF DACS

D/A converters are commonly used in two different ways that differ very much by their requirements:

Inside a sampled system
If the output of the D/A is sampled (e.g. inside a ΣΔ ADC or by a deglitcher circuit) many of its dynamic errors will be masked: glitches, and errors due to slew-rate or timing skew are not significant. Basically, only static errors matter now.

One interesting feature of ΣΔ ADCs is that although a low resolution DAC is used, its linearity still needs to be very high. For example, a 4-bit DAC with 16 bit linearity may be required.

As a continuous-time output
This is the most demanding application, as dynamic errors create very wide analog bandwidth of spurious components. With digital techniques, only components below fs/2 can be cancelled.
MATCHING ERRORS

Most DACs are based on matching of some unit elements (R,C,MOS). There are two types of matching errors:

Random errors

All component values have random variations. The only ways to minimize random errors is to increase the physical size (area) of the element and to maximize the symmetry of the surroundings by employing dummy structures etc.

Process gradients

Component properties vary along the chip surface, causing systematic errors. These can be compensated by spatial filtering, i.e. splitting the elements in several (2-16) sub-elements that are scattered so that the effect of 1st and 2nd order systematic variations are minimized.

\[ I_D = \beta (V_{GS} - V_T)^2 \]

\[ \frac{\Delta I_D}{I_D} = \frac{\Delta \beta}{\beta} + \frac{2(\Delta V_{GS} - \Delta V_T)}{V_{GS} - V_T} \]

\[ \left(\frac{\Delta I_D}{I_D}\right)^2 = \frac{A_{\beta}^2}{WL} + \frac{4}{(V_{GS} - V_T)^2} \frac{A_{V_T}^2}{WL} \]

Typical W*L in 12-14 bit DAC is 50..100 \( \mu \)m²

RANDOM ERRORS IN CURRENT-STEERING DACS

Random errors in MOS current sources depend on both gain and threshold voltage mismatches that both depend on the area of the transistor.

The required accuracy can be obtained from Monte Carlo simulations of the chosen topology. Below is an example where the achieved yield for 8-14 bit linearity (0.5 lsb INL) is calculated as a function of \( \Delta I/I \) in a 6-bit thermometer coded block.

Example:

To achieve 14-bit linearity, in (van der Plas et al jSSC99), unit transistors with \( W/L = 4/100 \) were used. The thermometer units were composed of 16 parallel unit elements, resulting in further 1:4 reduction in deviation. This is not excessive, and 12-14 bit static INL is actually achievable by using large enough devices.
SYSTEMATIC ERRORS (SPATIAL FILTERING)

On silicon chip, the device characteristics vary along the chip due to process gradients, thermal variations, and mechanical stress due to chip edges and top metallization. Also voltage drops in supply and bias lines appear as location dependence.

To achieve INL better than 8-10 bits, spatial averaging is already needed. Upper left, common centroid placement for binary and thermometer weighted codes are shown. Here, all elements have the same centre of gravity although they are scattered around the matrix.

The drawback of common centroid placement is that it cancels only linear and odd order trends in device characteristics. Second order trends are still untouched and cause less errors in centre elements than those placed in the periphery.

\[
x_c = \frac{\sum x_i \cdot m_i}{\sum m_i} \quad y_c = \frac{\sum y_i \cdot m_i}{\sum m_i}
\]

ANALYSIS OF SYSTEMATIC ERRORS

To analyse the effect of systematic errors, do as follows:
- Define a grid of unit element positions.
- Using these coordinates, generate systematic errors of form

\[
\begin{align*}
er r 1 &= e_{max} \cdot (x \cdot \cos(\phi) + y \cdot \sin(\phi)) \\
er r 2 &= e_{max} \cdot (x^2 + y^2)
\end{align*}
\]

where err1 corresponds to linear gradient in direction \( \phi \) and err2 corresponds to symmetric (square) error.
- Then, define weights for sources in each coordinate point and calculate total error as a function of input code, i.e. INL.
- Now by varying the position of sources grouped to a certain weight, different placements can be experimented.
Example

Left is shown a linear process gradient in 45 degrees direction, placement of unit elements (6 belongs to MSB, 1 to LSB) and the resulting INL. If the units elements were placed so that MSB is at the bottom of the array, then MSB-1 and so forth, an INL shown below results.

Thus, with more evenly distribute placement of the bits, the cumulation of the worst-case INL of +/-1 LSB has been effectively prevented.

Q^2 RANDOM WALK

To cope with higher order gradients, the unit elements need to be distributed even more effectively. By making a unit element (of the thermometer coded msbs) of 16 unit cells, more spatial averaging is achieved. In a technique called quad quadrants (Q^2), 4 unit cells are placed in each quadrant of the matrix. In a 16x16 matrix, altogether 16 thermometer coded sources can be placed.

This still leaves some residual error, and its effect is minimised by ordering the switching so that positive and negative errors (supposing quadratic errors) alternatively cancel each other. Hence, the buildup of INL is minimised.

(v.d Plasse et al.: A 14-bit Intrinsic Accuracy Q^2 Random Walk CMOS DAC, J SSC Dec99)
OUTPUT IMPEDANCE IN CURRENT-STEERING DACS

On of the main sources of static distortion is the finite output impedance $G_s$ of the current sources. A constant $G_s$ per source would only cause attenuation, but the outputs will see $x$ and $N-x$ sources in parallel, in which case the output impedance depends on signal. Left, output voltages $v_p$ and $v_m$ are derived as functions of output amplitude $x_a$. In a differential output the even order terms get cancelled

$$v_p = \frac{I_{lsb} \cdot x}{(G + xG_s)}$$

$$= \frac{I_{lsb}}{G_{mid}} \left( x_{mid} + \frac{G \cdot x_a}{G_{mid}} - \frac{GG_s \cdot x_a^2}{G_{mid}^2} + \frac{GG_s^2 \cdot x_a^3}{G_{mid}^3} - \ldots \right)$$

$$v_m = \frac{I_{lsb} \cdot (N-x)}{(G + (N-x)G_s)}$$

$$= \frac{I_{lsb}}{G_{mid}} \left( x_{mid} - \frac{G \cdot x_a}{G_{mid}} - \frac{GG_s \cdot x_a^2}{G_{mid}^2} + \frac{GG_s^2 \cdot x_a^3}{G_{mid}^3} - \ldots \right)$$

which greatly helps the situation. Still, this easily ruins the DC performance.

Example

14 bit DAC has maximally $6 \times 14 = 84$ dB SNR. To keep the distortion caused by signal dependent output impedance below noise, the output impedance of the current sources needs to be $>10^9$ times the load resistance. Hence, for $RL=50$ ohm and $Imax = 20mA$, $1/G_s$ needs to be $> 50$ Gohm. With a unit current $i_{lsb}=20mA/2^{14}=1.2$ uA this would require device Early voltage $V_A$ of 60 kV, while the typical $V_A$ in high-speed processes is $< 10V$. Hence, cascading with high loop gain is necessary.

DIGITAL PREDISTORTION

The static curvature caused by finite output impedance is one of the few mechanisms that could be compensated digitally by predistorting the input signal polynomially. However, the real distortion is a continuous time phenomena while the digital predistortion may alias around the $fs/2$. Hence, the degree of the predistorter can not be high (perhaps limited to 3 ?). Moreover, the inherent resolution of the DAC prevents correcting errors smaller than 1 LSB.

The amplitude of the distortion products can generally attenuated by the amount of loop gain $G$ when feedback is used. The feedback structure shown left is not directly applicable due to latency in A/D conversion. Instead, the error signal can be used to adapt the correction block $C$.

CANCELLATION OF SINGLE TONES

In some test systems, only 1- or 2-tone test signals are generated. In these, the spurious signals are discrete tones that can be cancelled by injecting them in the opposite phase in digital domain.

One method to estimate the amplitude and phase of the spurious tones in yin from plain amplitude measurements is proposed by Friel (M.Sc. Univ Maine, 2000):

\begin{align*}
A1 &= \text{spurious ampl. in yin} \\
A2 &= \text{spurious ampl. in yin + } a\cos(fspur) \\
A3 &= \text{spurious ampl. in yin + } b\sin(fspur)
\end{align*}

where $a= b = 0.75*A1$. Now the spurious tone can be cancelled by adding a signal

$$y_{spur} = A_{est}\cos(f_{spur}) + B_{est}\sin(f_{spur})$$

Estimate the amplitude of spurious tones by three spectrum measurements:
- plain yin
- yin + 75% cosine
- yin + 75% sine
ON-LINE TRIMMING

Especially multi-bit SD modulators require DACs that have low number of bits but very good linearity. One way to improve the linearity of the DAC is to employ a parallel trimming DAC that corrects the small linearity errors in the levels of the actual DAC. The proper correction values must be found somehow, e.g. by measuring the actual levels of the main DAC.

Example

Suppose a 4-bit DAC with max. 1% matching errors and requirement for 10-bit linearity (<0.1%) matching. Now the trimming DAC must handle e.g. +/- 2% LSB range with 2%/0.1% = 20 = 5 bit dynamic range. Thus, a combination of 4-bit DAC with 1% accuracy and 5-bit DAC with +/- 2% control range gives total of 10-bit accuracy.

DYNAMIC ELEMENT MATCHING

Dynamic element matching is based on either constant electrical tuning or randomizing and averaging the errors.

Electrical calibration

One element at a time is compared against a reference and tuned electrically

Random element selection

Left is shown a randomizer logic where a tree-type binary-to-linear switching is arranged by steering the switches with pseudo-random sequences. Quantization errors are shaped as excess white noise.

Noise-shaping

Noise-shaping is applied so that quantization errors are randomized and shaped away from the desired signal band.
Dynamic element matching is based on the fact, that if all unit sources are used equally often to present all desired quantization levels, then the average quantization levels will be free of error. This has the following general features:

**Good**
- simple: no calibration needed

**Problems**
- limited to unit sources only
- bandwidth limited: only the average is accurate, immediate values still have errors
- how to guarantee equally frequent use of all elements?

\[
\begin{align*}
    w_1 &= E\{s_i\} = s_{ave} \\
    w_2 &= E\{s_i + s_j\} = 2s_{ave} \\
    w_3 &= E\{s_i + s_j + s_k\} = 3s_{ave} \\
    &\vdots
\end{align*}
\]

"1" may be A or B or C...
"2" may be A+B or A+C or F+G or ...
"3" may be A+B+C or B+E+G or ...

**SPECTRAL SHAPING OF INL**

DAC linearity errors are especially harmful in multi-bit sigma-delta AD and DA converters, where high amount of quantization noise (dark) is shaped away from the signal band. Now the nonlinearities in the feedback DA converter mix the quantization noise back to the signal band (gray), causing reduced SNR and SFDR. This excess noise exists, but by properly alternating the order of switching the DAC elements, the excess noise can be shaped away from the signal band, too.

Several INL shaping or randomizing techniques exist, and the main criteria is to use all elements in the DAC uniformly, in which case the average output levels will be free of INL. The randomizing techniques differ mostly in the speed: how quickly they can re-use all the DAC elements. Low repetition rate may result in in-band (audible) tones and is bad for performance.
DWA (DATA WEIGHTED AVERAGING)

DWA (Baird, Fietz ISCAS95) is one of the most efficient techniques for randomizing linearity errors. It can be applied in thermometer coded DACs with unitary cells, and the idea is to always use unit elements that were not used in the previous sample. This can be illustrated as a wheel: if elements A have been used to produce the previous sample (5), group B will be selected to produce the current sample (3).

The block diagram shows the needed hardware: an integrator is used to create pointer P that shows the point of first unused unit element. The code is converted to thermometer code, and a logarithmic shifter is used to move the origin of the thermometer code to the new position.

In general, DWA is very effective, but:

- it gets complicated with large m
- with low OSR, a high order DWA is needed

NOTES ON DWA

DWA is one of the most effective DEM methods nowadays - left is an example of using DWA in a 3-bit, 1st order sigma-delta DAC with 5% DAC weight errors. The amount of distortion is clearly reduced.

However, DWA still does not guarantee full averaging, as always adjacent elements are chosen (e.g. combination c1+c3+c5 is not possible).

DWA can also be built to have e.g. bandpass response to clean up BP type sigma-delta converters. In general, the complexity of the DWA implementation increases rapidly as the number of bits increases.

To reduce the complexity, e.g. Galton has proposed several partial DEM algorithms.
SCALE ADJUSTMENT

Attenuator can be used to reduce the size of current source matrix: in the 10-bit example left, two 31 source banks and a 1:32 attenuator are needed instead of a single 1024 source matrix. However, now there is a risk of gain mismatch.

The gain of the lsb sub-DAC can be calibrated by comparing the currents \((M \cdot I_{\text{msb}} + 32 \cdot I_{\text{lsb}})\) and \(((M+1) \cdot I_{\text{msb}} + 0 \cdot I_{\text{lsb}})\). Based on the comparison, the reference current of the lsb DAC can be trimmed. For calibration, the following hardware is needed:

- control logic
- offset-cancelled current comparator
- 1 msb current source
- 31+1 lsb current sources
- trim DAC for lsb Iref

DYNAMIC ERRORS

12-14 bit DC linearity is achievable. However, DACs suffer from dynamic errors that reduce the SFDR very rapidly with increasing frequency. These effects are difficult to model and they include

- timing skew between the switching sources
- output slew rate

Both errors create rich spectrum of spurious signals that can not be affected by filtering, any more.
TIMING SKEW

One of the biggest causes of dynamic distortion in current-steering DACs is error in transition instant. Suppose a situation where 5 current sources need to be switched. On the left, the A switches immediately, B and C a few ps later, then D, and finally E. This creates an error signal shown below that is combination of pulse amplitude and width modulation (PAM&PWM) and has hence a very wide spectrum with tone amplitudes proportional to Bessel functions of the input.

(Doris, Leenarts, Roermund: Time Non Linearities in D/A converters. ECCTD01, Helsinki 2001)

Example

Left is plotted the amplitude of 2nd order spurious signals in a DAC where all msb current sources (i.e. the DAC is binary-weighted, not thermometer code) are delayed by $M \text{s}$. It is seen that even very small delay (5ps/10ns = 0.05%) in switching may produce noticeable spurious signals. The level of higher order products is roughly equal.

Eliminating these spurious signals is very difficult due to their wideband nature.
OUTPUT SLEW RATE

Constant slew rate of the continuous-time output of a DAC is a nonlinear operation, because the base width of the error pulse varies together with its amplitude. The amount of IM3 is difficult to predict, as it results in rich number of distortion (or modulation) terms, but one reasoning is given left.

The derivative of the error is easier to model. The sharp rise is caused by an impulse of area \( y(i) - y(i-1) \). This is a linear operation and causes only frequency dependent gain. Constant slew-rate is created by integrating a pulse of constant amplitude (that sets the constant slew rate) and equal area to the impulse.

Hence, the distortion caused by the slew rate can be modelled as a combination of discrete differentiation that gives the step height, pulse-width modulation that produces infinitely wide FM type spectrum, and finally, continuous integration that models the slewing portion

Numerically calculated effect of the slew rate are shown left, where the spectrums of ideal and slew-rate limited 2-tone signals are compared in a case where worst case slewing time is ca. 30% of the sample period. A rich spectrum of spurious tones is clearly seen.

Due to the PWM modulated origin, IM3 shows weird frequency dependence, increasing roughly 30 dB/decade with increasing signal frequency.
**PATTERN JITTER IN 1-BIT DAC**

In theory, 1-bit DAC is inherently linear. However, in continuous-time applications the finite rise and fall times affect the total energy of the bit, and e.g. the weight of bits 1, 2 and 3 are different, although they all should present logic 1.

The most common method to break this pattern dependency is to use RZ (return-to-zero) coding, where all symbols have both rising and falling edges. This reduces the weight of the bit and requires higher bandwidth but has in general given better results.

This effect is harmful in the feedback loop of continuous-time \( \Sigma \Delta \) AD converters but it is not so serious in the output of a 1-bit \( \Sigma \Delta \) DAC. Outside the feedback loop (when reduced weights are not injected in the loop anymore), linear errors appear simply as filtering and slew-rate errors cause PWM, but due to due to \( \Sigma \Delta \) modulated signal it has such a high modulation BW that the power density remains low.

**CIRCUIT TECHNIQUES**

**Cascode current sources**

are used to increase the output impedance of the unit current sources. Sometimes, the switch transistors serve as cascode transistors as well.

**Synchronization of the control signals**

is essential in reducing the glitch energy: as shown before, even very small timing skew may ruin the AC performance. Hence, every control signal must be separately synchronised to reduce the skew.

**Overlapping current switch drivers**

are used to guarantee that current is always steered to either output.
Interpolating output

The transition in the current switch can be smoothed, which results in an interpolating output. This reduces the effects of finite slew rate and also deepens slightly the \( \sin x / x \) output frequency response.


S/H deglitcher

Effect of output glitches can be reduced by using S/H in the output. This may be normal voltage mode S/H stage, or a current memory cell shown left. Here current is stored in either one of the memories while outputting current from the held memory cell.

Multiplexing

One easy way to mask the glitches and to increase the sampling rate is to use two DAC at \( fs/2 \) and choose out that one that has already settled. However, this has the same problems as parallel ADCs: offset and gain errors between the branches appears as residual sampling at \( fs/2 \), and \( fs/2 \) tones due to offset errors and \( fs/2 \) images due to gain errors are seen. 1% gain mismatch will create -40 dB spurious components, so typically matching better 0.1% is needed.

As such a high gain accuracy is needed, the resolution on the digital side may not be sufficient. Instead, the reference currents of the two DACs may be trimmable.
GAIN TUNING IN MULTIPLEXING

DC comparison
A DC method to tune the gain can be implemented as follows: first, the offset for code=0 is tuned to zero. Then, the amplitudes at the maximum code are tuned to be equal. For tuning, offset and gain control DACs and offset-compensated comparator is needed. The trim-DACs are drawn as current-output devices.

AC measurement
Correlation of the aliased band?