ERROR CORRECTION TECHNIQUES IN HIGH-SPEED A/D AND D/A CONVERTERS

BY:
TIMO RAHKONEN

ELECTRONICS LABORATORY
DEPARTMENT OF ELECTRICAL ENGINEERING AND INFOTECH OULU
UNIVERSITY OF OULU

PO BOX 4500
90014 OULU
FINLAND

email timo.rahkonen@ee.oulu.fi

CONTENTS

1. Introduction
   Standards
   Spectral effects
   Types of converters

2. Background
   Definitions
   Characterisation basics
   Figures of merit

3. Distortion mechanisms
   Static vs. dynamic

4. D/A case studies

5. A/D case studies
1. INTRODUCTION

Characteristics

- Standards
- Figures of merit
- Correction of nonlinearities
- Spectral effects of nonlinearity & sampling

Types of converters

- flash, folding, interpolating
- sub-ranging and pipeline
- sigma-delta
- binary and thermometer weighted DACs
- parallelism

STANDARDS


- definitions
- characterisation techniques


- extension to the previous
FIGURES OF MERIT

INL
• Integral nonlinearity, deviation from ideal quantization curve

SNR
• Signal power / Noise power over frequency band of interest (dB)

SNRD
• Signal power / Noise & distortion power over frequency band of interest (dB)

ENOB (Effective number of bits)
• Effective number of bits

\[ ENOB = \frac{SNR - 1.7}{6.023} \] bits

SFDR (Spurious Free Dynamic range)
• Distance to highest spurious signal (dBc)

NPR (Noise Power Ratio)
• Measures the depth of a deep notch in wideband noise input (dB)

SNR AND SFDR

N-bit quantization sets certain noise floor so that for a full-scale single-tone sinusoid,

\[ SNR \leq 6.023 \cdot N + 1.7 \] dB

However, the dynamic performance is mostly limited by spurious signal that are results of nonlinearities and aliasing. The power of the spurious components depends on signal frequency but is otherwise rather constant. 1- or 2-tone test produce small amount of tones with high peak power, while the spurious signals of wideband transmissions may have lower peak power. This is important for adapting digital correction: the spurious components of wideband transmissions are not so easily detected.
DISTORTION MECHANISMS

Memoryless nonlinearity
- static nonlinearity does not depend on input frequency
- plain quantization
- component mismatches, comparator offsets, gain errors in residue amplifiers, ...

Dynamic nonlinearity
- in most ADCs and DACs, SFDR drops with increasing frequency
- distortion in samplers, timing errors in parallel samplers, RC time-constants in resistor strings
- slew rate, timing errors, ...

Dynamic nonlinearity may be proportional to
- slope of the signal (continuous-time parts: samplers, S/H, flash converters)
- previous sample (sampled devices like pipeline and ΣΔ converters)
- Nth previous sample (in pipeline AD, crosstalk from digital side is delayed by N cycles)

CORRECTION OF NON-LINEARITIES

Mismatches of component values limit the achievable linearity to 10-12 bits. This can be improved by

- spatial filtering. Device mismatches can be averaged using proper layout geometry.
- production trimming, which is expensive and may require special process (e.g. NiCr resistors)
- electrical trimming. Requires accurate and linear trimming circuitry parallel to actual ADC/DAC.
- spectral shaping of the errors. If signal is oversampled, excess noise caused by non-linearities can be translated out of the interesting frequency band.
- digital correction. The non-linearity error is estimated and cancelled in the digital domain. This may require a calibration cycle or operate invisibly on background.

Most linearization methods require that
- the error can be measured or estimated, and
- a correction procedure. This is very dependent on the device architecture.
CORRECTING AD AND DA CONVERTERS

ADC
- Ideal signal must be estimated
- All spurious components are within fs/2
- All digital techniques are available

DAC
- Ideal signal is known
- Distortion is analog and may extend beyond fs/2
- Correction is limited to predistortion type correctors

The present state of the art is 12-14 bit 50-100 MS/s D/A converters and 10-12 bit 50 MS/s converters. Thus, it is still easier to build high-performance DA than AD.

REMINDER OF IMPORTANT TRANSFORM PAIRS

In linear systems, time domain convolution can be seen in frequency domain as multiplication of spectrums. However, in nonlinear systems, the spectral effects of time domain multiplication can be found by convolving the two-sided spectrums of the inputs.

**Linear system**
\[ y(t) = h(t) \ast x(t) \]
\[ = \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau \]
\[ Y(s) = H(s) \cdot X(s) \]

**Nonlinear system**
\[ y(t) = x(t) \cdot x(t) \]
\[ Y(s) = X(s) \ast X(s) \]
SPECTRAL EFFECTS

Sampling
Sampling is time domain multiplication with a series of impulses. Thus, the input spectrum will be convolved with a series of tones
\[ y(t) = x(t) \cdot h(t) \leftrightarrow Y(f) = X(f) \otimes H(f) \]

Sinc response
BW limitations in samplers and S/H response in DACs case filtering to the signal.

Distortion
Polynomial distortion is easiest to handle analytically, as time domain multiplication is convolution in frequency domain. For example, nonlinearity of form \( x(t)^3 \) is simply
\[ x(t)^3 \leftrightarrow X(f) \otimes X(f) \otimes X(f) \]

OUTPUT RESPONSE

The output format of a DAC affects its frequency response. Classical sample-and-hold has a well-known linear phase sinc response
\[ H(f) = \frac{\sin(\pi(f \cdot t_H))}{\pi(f \cdot t_H)} \]
where \( t_H \) is the hold time. Shorter hold time (return-to-zero response, RZ) can be used to broaden the bandwidth, or analog or digital inverse sinc filters can be used to cancel sinc droop.

Also other output formats can be used, e.g. a non-zero risetime in S/H response causes another sinc term, and left is also shown the frequency response of a bipolar output pulse.

Left, the response of a fs/4 multi-tone signal and wideband noise is shown to see the shape of frequency response.
MATLAB CODE FOR SHOWING SAMPLING SPECTRAS

function spectras

switch 1

case 1, % simple sampling case
nol = zeros(1,15); h = [nol 1 nol 1 0 0 ]; % clk spectrum
N1 = length(h);
x = [ 0 1 0.8 0.6 0.4 zeros(1,N1-5)]; % signal

case 2, % gain error in parallel samplers
nol = zeros(1,15); h = [nol 0.2 nol 1 nol 0.2 nol 1 0 0 ];
N1 = length(h);
x = [ 0 1 0.8 0.6 0.4 zeros(1,N1-5)]; % signal
end

xh = 1:N1; xh = [fliplr(-xh) 0 xh]; % x axis
h = [fliplr(conj(h)) 1 h]; % 2-sided clk spectra
x = [fliplr(conj(x)) 0 x]; % 2-sided signal spectra
N = length(h);
y = conv(x,h); y = strip(y,N); % convolve & truncate

figure(1)
subplot(311), stem(xh,x), ylabel('X')
subplot(312), stem(xh,h), ylabel('H')
subplot(313), stem(xh,y), ylabel('Y')

function [y]=strip(x,N) % truncate spectra
NN = length(x);
ero = (NN-N)/2;
y = x((ero+1):(ero+N));

NUMERICAL EXAMPLE: OUTPUT SPECTRUM OF A 2-TONE TEST

Supposing a 2-tone test, the output spectrum of Nth order nonlinearity can simply be obtained by convolving a 2-sided spectrum N times with itself. Note that the phase of the negative frequency components is opposite to positive frequency components.

In Matlab, it is easily done using function conv()
TYPICAL OUTPUT SPECTRUM

Image
Due to sampling, everything appearing in the input will be replicated around all integer multiples of the sampling frequency $f_s$.

Harmonics
If the distortion appears before sampling (e.g. in the S/H stage), also the distortion is sampled and - if not fitting into Nyquist band - will appear as aliased image $fs/N$ images.

In a case of $N$ parallel AD converters, any gain or timing mismatch appears as residual sampling at frequency $fs/N$. This creates spurious images around $fs/N$ with level proportional to matching error.

FORMS OF NONLINEARITY

Even nonlinearity (e(x) = e(-x))
- Even nonlinearity ($x^2$, $x^4$, ...) causes rectification to DC and even harmonics
- cancelled by differential circuitry

Odd nonlinearity (e(x) = - e(-x))
- Odd nonlinearity ($x^3$, $x^5$, ...) causes gain compression and intermodulation

Remember also
- multiple mixing
- self-heating
GENERAL TRENDS IN ADCS AND DACS

Performance of ADCs is limited especially by sampling jitter and SFDR. Performance of some commercial devices is listed below. The performance is improving much slower than Moore’s law, Walden estimated 1.5 bits / 8 years improvement rate.


<table>
<thead>
<tr>
<th>Table 1:</th>
<th>fas</th>
<th>bits</th>
<th>SNR</th>
<th>SFDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>DAC:</td>
<td>100</td>
<td>12-14</td>
<td>70-75</td>
<td>65-85</td>
</tr>
<tr>
<td>ADC</td>
<td>65</td>
<td>14</td>
<td>75</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td>500</td>
<td>8</td>
<td>48</td>
<td>60</td>
</tr>
</tbody>
</table>

FLASH ADC

Good
- Fastest A/D topology
- Short latency
- Resistor string is inherently monotonic

Bad
- One comparator for each decision level
- \(2^N - 1\) comparators
- Large input capacitance

Error mechanisms
- Resistor and comparator matching
- Resistor string reference levels
- Time constants of the resistor string
- Bubbles caused by comparator offsets & metastability
FOLDING AND INTERPOLATION

One way to increase the resolution of flash converters is to employ analog preprocessing. There are two ways to perform this: interpolation and folding.

Interpolation principle

Certain signals or reference levels are not generated. Instead, the existing signals are used to interpolate these signals (using resistor strings and pre-amps) before applying them to the comparators. The main advantage of this is the reduction of input capacitance.

Folding principle

The folding principle aims in reducing the number of comparators by performing continuous-time sub-ranging conversion. Here, a piece-wise linear folding amplifier is used to produce a residue voltage that is digitized using a flash ADC. This requires very high BW from the input stage.

FOLDING ADC ERRORS

Error sources

- Reference inaccuracy
- Folding amplifier input offset
- Tail-current mismatch (gain error of segments)
- Interpolation error

Offset trimming in a combined folding/interpolating circuit is described by (Choe et al., j SSC December 2000)
**SUBRANGING CONVERTERS**

- smaller flash converters with interstage residue amplifiers
- smaller number of comparators
- needs accurate D/A and residue amplifiers and
- redundancy in the latter stage is needed to avoid clipping due to offset in AD, DA or amplifier
- remove redundancy when combining the outputs of the subconverters

**Example:**

8-b flash ADC needs $2^8 = 256$ comparators. A 2-step flash converter with one redundant bit in the latter stage needs $2^4 + 2^{4+1} = 16 + 32 = 48$ comparators.

**PIPELINE CONVERTERS**

- Typically, 1 bit (3-level)/stage sub-ranging stages: each stage has 3 levels, but only 1 bit is coded
- High throughput: input can be sampled at every clock cycle
- Latency: due to pipeline structure it takes N cycles to have a complete result

In the conventional pipeline, the amplifiers are idle half of the time. In a double sampling stage, two banks of sampling capacitors is used so that a new sample can be taken while the previous one is still being evaluated. This gives a full cycle of settling time but may result in different gain for even and odd samples.

(see IEEE papers by Lewis, for example)
SIGMA-DELTA CONVERTERS

Quantization noise of a 1-bit (or low-resolution) quantizer is shaped out of the signal by band strong error feedback

- Requires high oversampling ratio (OSR)
- Quantizer gain and thus loop response may depend on signal conditions
- Thus loop gain and pole positions are signal dependent
- Essentially an RF device: quantization noise and signal are spectrally separated
- Several choices for loop filter topologies
- Multi-bit quantizers and feedback becoming popular to reduce OSR

(book and Matlab toolbox by Schreier)

STABILITY ANALYSIS OF SIGMA-DELTA A/D CONVERTERS

In sigma-delta converters, the gain of the quantizers affects stability as shown in signal and noise transfer functions below

\[
STF(z, \lambda_{\text{sig}}) = \frac{\lambda_{\text{sig}} \cdot H(z)}{1 + \lambda_{\text{sig}} \cdot H(z)} \quad NTF(z, \lambda_{\text{noi}}) = \frac{1}{1 + \lambda_{\text{noi}} \cdot H(z)}
\]

The gain of the comparator varies \(V_{\text{out}}/V_{\text{in}(\text{max})}\) to infinity. It can be estimated statistically by driving the converter with a set of dc voltages, measuring dc and ac powers in the input of the comparator and their correlation with the output \(y(t)\):

\[
\lambda_{\text{sig}} = \frac{E\{y(t) \cdot u_{\text{dc}}\}}{u_{\text{dc}}^2} \quad \lambda_{\text{noi}} = \frac{E\{y(t) \cdot u_{\text{ac}}(t)\}}{E\{u_{\text{ac}}^2(t)\}^2}
\]

From this it is apparent that when input dc voltage is increased, the effective comparator gain decreases, which moves to closed loop poles and may result in instability.
SIGMA-DELTA CONVERTER TOPOLOGIES

SDMs can be built in several forms:

**Error feedback**
This requires very accurate gain of 1, and is hence used mostly in digital $\Sigma\Delta$ DACs.

**Multiloop feedback**
Most common. Needs several DACs

**Feedforward structure**
Needs only one DAC, easy to adapt to multibit structure

**MASH**
Good stability properties, but requires accurate gain match and (adaptive) digital combining logic.

---

TYPICAL DAC TOPOLOGIES

**R-2R ladders and cap DACs**
Binary weighted R-2R ladders are very area efficient implementations, but usually, thermometer coding is needed above 8-9 bits. R-2R is also sensitive to DC offset in the current sensing node.

**Algorithmic/pipeline**
Also DACs can be built as algorithmic or pipelined devices

**$\Sigma\Delta$ DAC**
In a $\Sigma\Delta$ DAC, a digital $\Sigma\Delta$ modulator is followed by a 1-bit output and possibly a semi-digital decimating filter

**Current steering DAC**
The fastest DAC type
CURRENT STEERING DACS

Left are shown the most typical DAC architectures. Usually, thermometer coding in 4-8 highest bits is used to guarantee monotonicity in the highest bits. As thermometer coding is expensive ($2^M$ different controls), the lowest bits are usually binary coded. Now one error source is the gain error between thermometer and binary blocks.

To reduce the size of the binary matrix, it can be built of two equal blocks that are summed using an attenuator. Now the gain and delay of the attenuator may create an additional error source that needs to be compensated somehow.

THERMOMETER CODE

Thermometer coding is used to guarantee monotonicity. In binary coded DAC transition from state 3 to 4 means that sources E-G (weights 2 and 1) are disconnected and sources A-D are connected. As these have different errors, output for state 4 may actually be less than for state 3.

In thermometer code transition from state 3 to 4 simply means that the fourth source (D) is connected in parallel with E-G. The sources are unitary, they all have weight 1.

Thermometer coding needs a lot of control and is usually limited to 4-6 bits. To reduce hardware complexity, coding is sometimes implemented segmentally (one coder for 4 upper and another 4 lower bits), although this is not as effective: it is still possible to have discontinuity when lower bits are zeroed and upper incremented by one.
TIME-INTERLEAVING

Two samplers are connected in parallel, sampling alternatively. This doubles the settling/conversion time but causes new problems:

- gain errors cause $fs/2$ image, as now the sampling gains are $G$, $G+\Delta G$, $G$, $G+\Delta G$, ..., i.e. constant $G,G,G,G,...$ plus $0$, $\Delta G$, $0$, $\Delta G$, $0$, $\Delta G$, ...
- offset errors cause $fs/2$ spurious signal: offset sequence is of form: $0$, off$s$, $0$, off$s$, $0$, off$s$, ...
- timing skew in sampling causes frequency dependent $fs/2$ image. Now every second sample is advanced/retarded by $\Delta t$, and this causes voltage error of form

$$\Delta u = \frac{du}{dt} \Delta t$$

Example of $fs/2$ images

Left is illustrated a case where the actual sampling is performed at frequency 32. Due to gain error in parallel samplers sampling at $fs/2$, also sampling at frequency 16 appears at lower gain. This creates sampling images in the spectrum.

Timing error in even or odd samples creates similar images, the amplitude of which depend on the signal frequency - at low frequencies the images are low but increase with increasing tone frequency.

Offset errors between the channels create single spurious tones at $fs/N$. 