A Behavioral Representation for Nyquist Rate A/D Converters

Edward Liu, Alberto L. Sangiovanni-Vincentelli, Georges Gielen, and Paul R. Gray
Department of Electrical Engineering & Computer Sciences
University of California
Berkeley, California 94720

Abstract

This paper presents a behavioral representation for the class of Nyquist rate A/D converters. The representation captures the nominal A/D behavior, as well as all the statistical variations. The variations are classified into noise and process variations according to how these non-idealities affect the A/D behavior. To describe noise effects we use a joint probability density function. To describe behavioral effects due to process variations we use a variance-covariance matrix, Σ_t, which is a generalization of the integral nonlinearity vector. Σ_t's rank characterizes the testability of an A/D; its decomposition yields efficient strategies for A/D testing. Finally, parameter extraction results obtained from prototypes are presented.

1 Introduction

A comprehensive **constraint-driven top down design** approach for complex mixed analog/digital systems is being developed. In such approach a mixed analog/digital system is represented by a hierarchy of behavioral models as shown in figure 1, where the level of abstraction increases from the implementation details at the bottom to an abstract mathematical model at the top. Each data node in this **model hierarchy** is a behavioral representation. We are researching behavioral representations for many types of mixed A/D circuits. In this paper we propose a behavioral representation for the class of Nyquist rate A/D. The representation captures the nominal A/D behavior, as well as all the statistical variations. The behavior of an A/D is affected by two basic statistical effects: (1) *noise*, (2) *process variations*. Noise can be of different sources, e.g, thermal noise, flicker noise, and noise coupled from digital circuitry. Noise can cause the same die to behave differently even if the same inputs are applied. Process variations cause different dice to have different behavior.

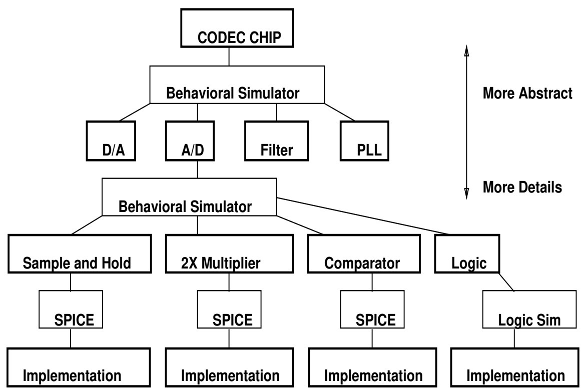

Figure 1: Model Hierarchy for Mixed A/D System

2 Previous work & our contribution

Circuit designers usually model noise effects by adding an input-referred noise to the input of an ideal, noiseless A/D. This approach is problematic because the A/D is nonlinear. So, noise cannot be referred to the input in general. They usually model process variations effects by several parameters such as offset error, gain error, integral nonlinearity (INL), and differential nonlinearity (DNL). This approach is inefficient because these parameters are not totally independent. In fact, all of these parameters can be derived from Σ_t, a variance-covariance matrix.

The traditional CAD approach to A/D behavioral modeling is to model the A/D transfer function[2] only. For a Nyquist rate A/D, the transfer function maps a continuous input value to an output code as shown in figure 2. This type of modeling can be expressed mathematically as

$$code = f(V_{in}, \hat{v}), \hat{v} = realization(v) \quad (1)$$

where V_{in} is the input, and $\hat{v}$ is a set of parameters describing process variations. As mentioned in the pre-

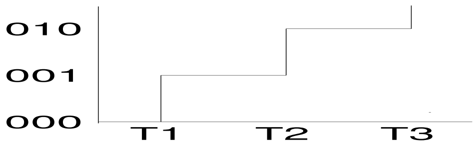

Figure 2: Nyquist A/D transfer curve

vious section, the sources of non-idealities in analog circuits are statistical, so $\hat{v}$ is a realization of a multivariate distribution v. There are two problems with this behavioral modeling approach. (1) Suppose v is an m-variate; m can be very large because every circuit element that affects the transfer function has process variations. Each of these variations contributes to a variate in v. (2) From this behavioral model, it is unclear which realizations $\hat{v}$ should be selected from v. For example, to verify worst case system performance without detailed knowledge of the A/D architecture, 2^m simulations have to be performed.

2.1 Derivation of new behavioral model

A Nyquist rate A/D is a nonlinear dynamic circuit, where the term dynamic refers to the effect of past input values. The dynamic effects originate only from the front-end sample-and-hold (S/H) because the signal is held constant by the S/H at the S/H output for subsequent circuit stages. Subsequent stages process this static signal independent of past signals so that the subsequent stages are not dynamic. As a result, we can represent the A/D by a dynamic block corresponding to the S/H followed by a nonlinear static block corresponding to the subsequent stages. This paper focuses on the modeling of the nonlinear static block.

2.1.1 Nominal transfer curve

The nominal transfer curve plots the output code on the range against the input continuous value on the abscissa(figure 2). Assuming the A/D monotonic and with no missing codes, the transfer curve can be characterized by 2^N real numbers, $t_1, t_2, ...$ located at the transition points of the staircase in figure 2, where N is the number of bits in the A/D. Collectively, these numbers are combined in a vector μ_t, representing the expected value of the transition points.

2.1.2 Variations in transfer curve

Due to statistical process variations, the transfer curve has variations. This has been characterized historically using the integral nonlinearity curve, INL, which plots the variance of each of the 2^N transitions mentioned in section 2.1.1 against increasing output code value. For more accurate modeling we use, instead, the variance-covariance matrix Σ_t of the transition points. Σ_t is a generalization of the INL because the diagonal entries of Σ_t can be plotted as a curve corresponding to the INL. To find Σ_t we make the following assumptions. (1) *Process variations are the cumulative effects of many integrated circuit fabrication steps. Thus, according to the Central Limit Theorem in probability theory, the distribution of the process variations, v, is approximately multivariate normal.* (2) *In general, an A/D is designed to be insensitive to process variations such as capacitor mismatches, which are relatively small. Thus, the transition points t change relatively little with respect to process variations. So, to a good approximation, t change linearly with respect to the process variations.*

From assumptions 1 and 2, we conclude,

$$\Sigma_t = S_v \Sigma_v S_v^T \tag{2}$$

where v is a random vector of parameters, corresponding to process variations such as errors in resistors and capacitors values, Σ_v is the variance-covariance matrix of v, and S_v is the sensitivity matrix $\frac{\partial t}{\partial v}$. Thus, the distribution of t is

$$t \sim normal(\mu_t, \Sigma_t) \tag{3}$$

The Σ_t matrix is not full rank in general, since not all the transition points are independent. Thus, it is more convenient to use the singular value decomposition of Σ_t. For storage and computational efficiency we express t as a linear transformation of a smaller zero mean r-variate normal distribution c,

$$t = Uc + \mu_t, c \sim normal(0, \Sigma_c) \tag{4}$$

$$\Sigma_t = U \Sigma_c U^T \tag{5}$$

where U is an 2^N x r transformation matrix, Σ_c is an r x r diagonal matrix, and r is the rank of Σ_t. Using this procedure, c is a multivariate normal distribution with r statistically *independent* variates. The columns of U are called the **error signatures** [3] and form a set of orthonormal vectors spanning the **error space**.

While t completely describes the distribution of the transition points, often the differences between the transition points provide more insight into some A/D architectures. We can derive another distribution d as follows,

$$d = D_+ t \tag{6}$$

where d is a first order difference of t and D_+ is the difference operator. Correspondingly, the variance-covariance matrix, Σ_d, is given by:

$$\Sigma_d = D_+ \Sigma_t D_+^T \tag{7}$$

Note that Σ_d is a generalization of the differential non-linearity vector (DNL) because the diagonal entries of Σ_d can be plotted as a curve corresponding to the DNL.

2.1.3 Output uncertainty

Due to noise, an A/D produces a wrong output with finite probability. For example, if the input is near a transition point, the probability of getting a wrong output code, the code at the other side of the transition point, is non-zero. Mathematically, this is represented by a probability distribution function of the output codes for any input value. Our behavioral model captures this behavior by a joint probability density function of the input V_{in} and the output code for a given realization of the transfer curve $\hat{t}$:

$$f(code, V_{in}, \hat{t}), \hat{t} = realization(t) \tag{8}$$

2.2 Model summary

In our approach, we separate the effects due to noise and process variations. In the latter case, we found that process variations v are not an optimal choice for describing A/D behavioral variations. For example, circuit architectures are usually designed to be insensitive to process variations; therefore, the behavioral variations due to process variations can be sufficiently characterized by the minimal r-variate distribution c with statistically independent variates. We also found a good visual representation for A/D behavioral variations, Σ_t, which is a generalization of the traditional INL vector. For effects due to noise, we propose a joint probability density function. Summarizing, our entire model consists of three equations,

$$f(code, V_{in}, \hat{t}), \hat{t} = realization(t) \tag{9}$$

$$t \sim normal(\mu_t, \Sigma_t) = Uc + \mu_t \tag{10}$$

$$c \sim normal(0, \Sigma_c) \tag{11}$$

3 Applications of model

3.1 System performance prediction

The A/D model presented is computationally efficient for performance prediction of systems containing an A/D as a block because the A/D non-idealities are all summarized statistically in the model. The relevant system parameters being estimated are usually signal-to-noise and distortion ratios (SNR and SDR), both of which depend on the non-idealities of the A/D block. To compute the SDR and its variations due to process variations, the traditional approach necessitates computation of the SDR sensitivity with respect to an m-variate distribution v. In contrast, our approach only requires the sensitivity with respect to a minimum r-variate distribution c. To compute the SNR the traditional modeling approach uses Monte Carlo techniques which are very inefficient for a reliable system with high SNR. In contrast, our A/D model provides the joint probability density function that can be used by a behavioral simulator at the system level to calculate the SNR directly.

Finally, it cannot be overemphasized that the *non-idealities are statistically summarized* as parameters in the model. The parameter extraction for an A/D is performed only *once*, and the resulting model is used many times at the system level. For example, a system with four channels of time interweaved pipelined A/D is currently under evaluation for effects due to mismatch between channels. We extracted once the parameters for a single channel to be used many times in the system level simulation to find the SNR and the SDR.

3.2 A/D testing using behavioral model

We can find the complete transfer curve of a sample chip of a Nyquist rate A/D with a minimal set of measurements. From the characterization results we can conclude whether the chip is out of specifications. We can formulate the testing problem by noting that a sample of t called $\hat{t}$ is composed of the nominal value plus a location in the error space (realization of (10)):

$$\hat{t} = U\hat{c} + \mu_t \tag{12}$$

where $\hat{c}$ is a realization of c and is an r x 1 vector of unknown coefficients to be determined by linear regression. Notice that $U\hat{c}$ completely specifies the location in the error space. To estimate $\hat{c}$, we use (12):

$$D_+\hat{t} = D_+U\hat{c} + D_+\mu_t$$

$$\hat{d} = D_+U\hat{c} + D_+\mu_t \tag{13}$$

Using (13), $\hat{c}$ can be determined by linear regression with data from measurements of the entries in $\hat{d}$. Recall that each piece of data in $\hat{d}$ is the measured difference between two transition points, and the transition points are obtained by a search algorithm. The minimum number of data measurements needed is r which is much less than the usual 2^N in full testing.

4 Parameter extraction results

Behavioral simulators have been implemented using the C++ language on a DECstation 3100 for the following circuits and their parameter extraction results are presented below. (a) Σ_t for a 4 bit resistor string A/D (figure 3, CPU time negligible) (b) Σ_t for a 4 bit Cyclic A/D generated by CADICS[1] (figure 4, CPU time 0.5 secs) (c) Error signatures for A/D in (b) The first signature corresponds to offset error. The second corresponds approximately to gain error. The third is not known in prior literature. (figure 5) (d) Joint probability density function of a cyclic A/D (figure 6, CPU time 266.3 secs).

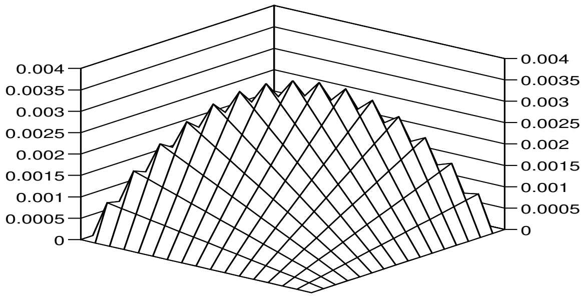

Figure 3: Σ_t of R-String A/D always has full rank

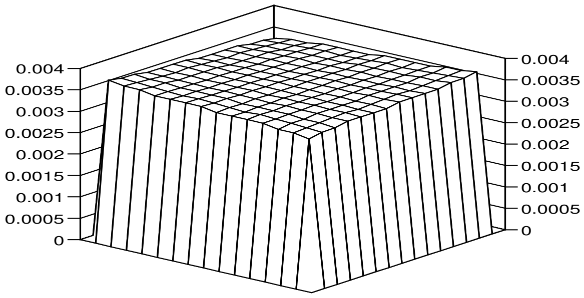

Figure 4: Σ_t of A/D by CADICS has rank 3

5 Conclusion

We have presented a behavioral representation for the class of Nyquist rate A/D converters. The representation captures the nominal A/D behavior, as well as all the statistical variations. Also, we presented parameter extraction results obtained from prototypes. At present we are applying the same concept of hierarchical modeling and behavioral characterization of circuit functionality to model other types of analog circuits. Specifically, we are researching behavioral representations for phase locked loops, oversampling A/D, and filters.

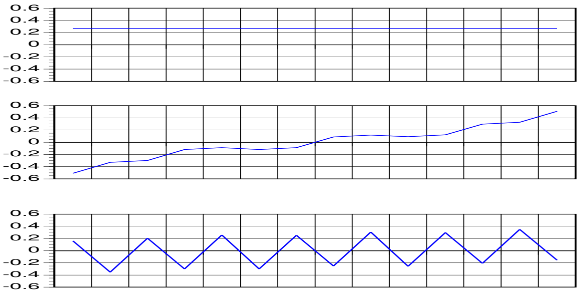

Figure 5: The 3 error signatures of A/D by CADICS

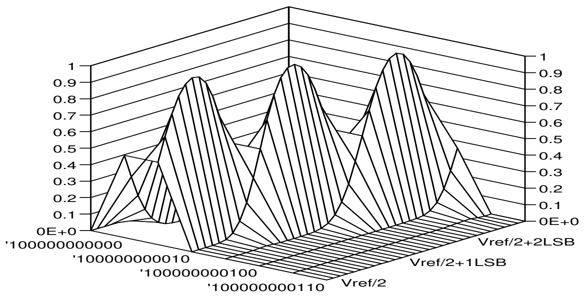

Figure 6: Joint probability density function of output code and input value for cyclic A/D

Acknowledgements

The authors acknowledge the many helpful discussions by the members of CAD and IC groups at U. C. Berkeley. This project is supported by the Semiconductor Research Corporation. Its support is gratefully acknowledged.

References

[1] G. Jusuf, P. R. Gray, A. L. Sangiovanni-Vincentelli "CADICS-Cyclic Analog-To-Digital Converter Synthesis", ICCAD 1990 .

[2] Ruan, Genhong "A Behavioral Model of A/D Converters Using a Mixed-Mode Simulator", *IEEE Journal of Solid-State Circuits*, Vol. 26, No. 3, March 1991

[3] T. M. Souders, G. N. Stenbakken, "Cutting the high cost of testing", *IEEE Spectrum*, March 1991.