Top-Down, Constraint-Driven Design Methodology Based Generation of n-bit Interpolative Current Source D/A Converters

Henry Chang Edward Liu Robert Neff Eric Felt Enrico Malavasi
Edoardo Charbon Alberto Sangiovanni-Vincentelli Paul R. Gray

Electronics Research Laboratory
Department of Electrical Engineering and Computer Sciences
University of California, Berkeley, CA 94720, U.S.A.

Abstract

To accelerate the design cycle for analog circuits and mixed-signal systems, we have proposed a top-down, constraint-driven design methodology [1]. In this paper we present a design which demonstrates the two principal advantages that this methodology provides– a high probability for first silicon which meets all specifications and fast design times. We examine the design of three different 10-bit digital-to-analog (D/A) converters beginning from their performance and functional specifications and ending with the testing of the *fabricated parts*. Critical technology mismatch information gathered from the testing phase is provided.

1 Introduction

As the level of system integration increases, one apparent bottleneck is the design of the analog circuit components. Though these components are often small in area, their design times can be high. To address this problem we have proposed a top-down, constraint-driven design methodology [1], and with it, have provided a variety of design tools such as behavioral level simulators, optimization tools, and physical layout tools. Higher-level tools such as module generators [2] have also been written. To show the effectiveness of this methodology, we have undertaken design examples beginning with industrial-strength specifications, and not ending until the designs have been fabricated and tested. This paper presents a complete design flow of an interpolative switched current source D/A converter. As a by-product, a new module generator for this class of 2-stage, n-bit D/A converters has also been developed.

This paper is divided into three parts. First, the application of the design style to the synthesis phase is described beginning with the architecture and the performance specifications and ending with the layout generation. Compared to some previous works which had also adopted this top-down design style, this design adheres much more rigorously to the methodology. In particular, behavioral level simulators with their ability to dramatically decrease simulation times over circuit simulators were made essential. Using behavioral level simulations at the high-level in conjunction with circuit simulations at the low-level, the integral nonlinearity (INL) and differential nonlinearity (DNL) statistics for the entire D/A converter can be calculated on the order of minutes [3]. Costly full chip circuit simulation is not necessary. By decreasing simulation times, more system-level simulations can be completed, and thus, errors can be caught earlier in the design cycle; and trade-offs at the system level can be better examined. Also more rigorously followed, is the design's adherence to hierarchy to better manage system complexity.

Presented in the second part of this paper will be data on the three 10-bit D/A converters that were fabricated. The D/A's have varied architectures and were fabricated on two different 2.0μm CMOS processes. Two were integrated on the same chip. Each D/A has no more than 2.0 lsb of INL error, 0.5 lsb of DNL error, and can settle for a full scale change at the output to 1 lsb in 20ns.

Finally, in the last section, the determination of critical component mismatches and process variations will be presented. Using the measured INL data and the modeling information retained from the behavioral level simulations, the current source mismatches were extracted. Then using this data, the underlying process variations on W, L, and V_{TH} were estimated.

By adopting this design scheme, the design time for this type of D/A has been greatly reduced, and a significant amount of data has been gathered for fault diagnosis. As an example of the design speed-up, the second of the two chips fabricated required only *five days* from the time of receiving the design specifications to sending it out for fabrication. All three D/A's required only one fabrication run each.

2 Design Specifications

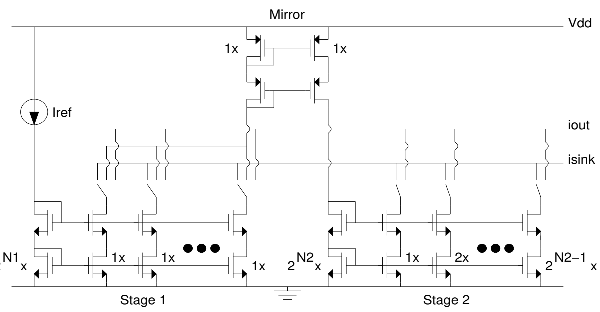

Figure 1: D/A Architecture

The converter that has been built is an interpolative switched current source D/A [4] (Figure 1). Given an N-bit digital word (D_{in}) and a reference current (I_{ref}), it returns an output current (I_{out}) which is proportional to D_{in}.

The converter has two stages. The first stage consists of unity or linearly weighted current sources each having a nominal value of $\frac{1}{2^{N_1}} I_{ref}$. This stage decodes the first N_1 bits of D_{in}. Depending on these bits, the currents are either channeled to the output (I_{out}), the current mirror, or the current sink (I_{sink}). If it is determined that n current sources need to be switched to I_{out} then the first n current sources are switched to I_{out}; the $(n+1)^{th}$ current source is switched to the current mirror, and the remaining unused current sources are switched to I_{sink}. The second stage is made of binary weighted current sources with the smallest current source having a nominal value of $\frac{1}{2^N}\ I_{ref}$. It decodes the remaining $(N - N_1)$ bits (N_2) of D_{in}. Depending on the remaining bits, these current sources are either switched to I_{sink} or to I_{out}. The currents are summed at the I_{out} node producing the analog output current.

Type	Specifications
Architecture	N_1, N_2
Performance	Max 3σ bound on INL, max 3σ bound DNL
Operation	Supply voltages, output voltage (V_{out}) range, I_{ref}
Technology	Design rules, SPICE parameters, process variation (σ_W, σ_L), threshold voltage variation ($\sqrt{a_{\mathrm{VFB}}}$) [5]
Layout	Input/output terminal locations, non-default cell placement, minimum wire widths for critical signals, use of bonding pads for a stand-alone part, etc.
Optimization	Non-default cost function(s)

Table 1: Design Specifications

Table 1 lists the set of design specifications for the D/A. They are broken down into different categories; each of which is handled differently as will be described later. The user enters these parameters to the D/A module generator.

3 Synthesis Path

Figure 2 shows the two-level hierarchy used in the synthesis process. The five partitions or blocks into which the D/A has been divided each contain a set of constraints and a set of parameters to be determined by the generator for that block. The synthesis begins at the top-most block and descends by choosing values for the parameters of that block. These chosen values then become constraints for the next block. This continues until all of the final low level parameters (e.g. transistor sizes) are chosen. The following sections explain the constraints and parameters applied in this design.

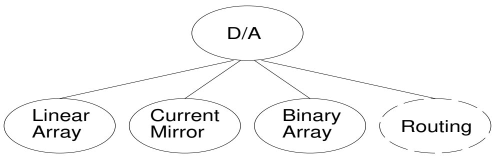

Figure 2: D/A Hierarchy

3.1 High Level Synthesis

The top block in the design as shown in Figure 2 is the D/A. The constraints at this level are the overall D/A performance specifications, INL and DNL. The parameters for this block are shown in Table 2. Two tools are applied to select values for these parameters.

Parameter	Description
$\sigma_{iL}, \sigma_{iM}, \sigma_{iB}$	Standard deviations of the current sources in the linear, mirror, and binary array respectively.
R_L, R_M, R_B	Output resistances of the current sources in the linear, mirror, and binary array respectively.
$\triangle$INL, $\triangle$DNL	Errors introduced from block level routing.

Table 2: Parameters for D/A Block

The first is the behavioral simulator [3]. Its primary input parameters are the ones listed in Table 2, as well as some of the architectural, operational, and technological information found in Table 1. The simulator returns the statistical INL and DNL characteristics for the D/A converter. It is important to note that the behavioral simulator is implementation *independent*; the current sources do not have to be the ones shown in Figure 1. The simulations performed are valid for any current source circuit that can be characterized by the parameters in Table 2.

A nonlinear optimizer [6] is the other tool that is invoked. This tool requires the performance and optimization information found in Table 1. The performances are used as constraints for the optimization problem. The objective function maximizes the "flexibility" or the chances that the low-level blocks can be implemented. For example, the flexibility function implemented is one which estimates how difficult based on basic device mismatch data a certain σ_i is to obtain, and how difficult a certain R is to achieve based on the architecture and operating conditions. For example, the flexibility is increased when σ_i is large, since current sources with larger mismatches are easier to build, and likewise, the flexibility is decreased when R is large, since current sources with large output resistances are more difficult to build. Figure 3 shows example flexibility functions for σ_i and R. It is also important to note that the marginal flexibility increase for σ_i decreases as σ_i becomes large. This accounts for the fact that beyond a certain point, the mismatches are so great that almost no design benefit is gained from building worse matched current sources from such ill-matched current sources at the start. Similarly, there is a decrease in the absolute value of the slope as R becomes small. A hyperbola and a parabola are used respectively for the flexibility functions.

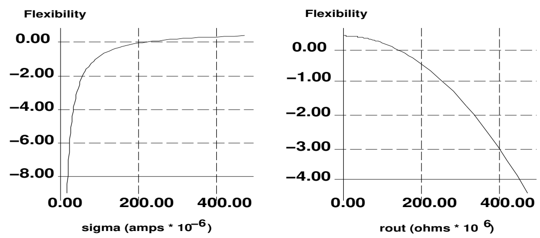

Figure 3: Example flexibility functions for σ_i and R.

Working together, the nonlinear optimizer chooses numerical values for the parameters in the D/A. The performance of the D/A is evaluated using the behavioral simulator, and more values are chosen. This process continues until all of the performance criteria have been met and the flexibility function has been maximized.

3.2 Low Level Synthesis

The selected values (denoted by a *) from the high level synthesis above now become constraints for the four blocks in the next level of hierarchy. For example, σ^*_{iL} and R^*_L are the constraints for the linear array generator. As long as the linear array generator produces a current source array with $\sigma_i \leq \sigma^*_{iL}$, and an $R \geq R^*_L$, then based on the behavioral level simulation results, it can be asserted that if all of the low-level blocks meet the constraints on σ_i and R then the D/A performance constraints (i.e. INL and DNL) will be satisfied. This assertion is checked during the final extraction and verification phase (Section 3.4) at the end of the design cycle.

Given the high-level constraints, the linear, mirror, and binary array generators are invoked to select the widths and lengths of the main and cascode transistors. As before, a simulator and an optimizer are used. SPICE enhanced with a statistical package receives as inputs a set of W's and L's corresponding to the main and cascode MOS devices along with architectural, operational, and technological information from Table 1, and returns σ_i and R. The optimizer, employing a cutting plane algorithm [7], minimizes the layout area subject to the constraints on σ_i, R, and operating condition constraints from Table 1. Then, as at the high-level, the optimizer chooses values the parameters, in this case the W's and L's; invokes the circuit simulator to determine whether or not the constraints are satisfied; and continues until the objective function has been minimized and the constraints have been met.

If completed successfully, the selected values, W^*'s and L^*'s, will satisfy the σ_i, R, and operating condition constraints. Therefore, with the previous assertion on the relationship between σ_i's and R's to INL and DNL, the D/A generated should meet the performance constraints on INL and DNL.

3.3 Physical Layout

The layout generation phase consists of two steps. First, the three arrays based on the results from the low-level synthesis phase are generated. The linear array, the binary array, and the current mirror generators use templates to create subcells, which are tiled to generate the desired blocks. The binary array current sources are tiled together, and the summed currents are passed to a bank of switches. The linear array uses a standard two-dimensional structure [8], with the addition of new decoding circuitry to select the interpolated output. To minimize sensitivity to process gradients, the binary array current sources are connected in a common centroid pattern, and the linear array devices are switched using a symmetrical switching pattern [8]. A standard cell generator was used to synthesize the logic and latches needed for the D/A.

With complete schematics from the low-level blocks, $\triangle$INL* and $\triangle$DNL* constraints, and layout information from the design specifications, the three analog blocks along with the digital block, are routed. [3] describes how $\triangle$INL and $\triangle$DNL are calculated.

3.4 Extraction and Verification

During this final phase, a full layout extraction including routing parasitics is performed. The results for each block is fed into the circuit simulator which returns σ_{iL}, σ_{iM}, σ_{iB}, R_L, R_M, and R_B. Then this plus $\triangle$INL and $\triangle$DNL is fed into the behavioral simulator which returns the INL and DNL statistics for the D/A generated.

Sensitivity analysis at the behavioral level [3] is used to insure that extra parasitics introduced in the routing has not degraded the D/A's performance beyond the $\triangle$INL and $\triangle$DNL amount allocated.

4 Experimental Results

We have fabricated three 10-bit D/As. The first chip (die photo shown in Figure 4) contains a D/A with N_1=5, N_2=5 (5/5) partition. The active area is 1.92mm^2. The second chip (die photo shown in Figure 5) contains two D/A's one with N_1=6, N_2=4 (6/4) and the other with N_1=4, N_2=6 (4/6). The active areas are 2.86mm^2 and 3.81mm^2, respectively. The areas compare reasonably with previously published D/A's [9][10]. The performance specifications in all three cases was that there be no more than 2 lsb of INL error and 0.5 lsb of DNL error. Figure 6 depicts the measured INL. The top set of graphs show the data gathered from all of the parts while the bottom set depict only the components which met the performance specifications.

Although not in the design specifications, transient testing on the D/A shows that for a full-scale switch at the output, the D/A can settle to 1 lsb for 10 bits from the start-of-conversion in 20ns. The delay incurred from the falling edge of the clock to the start-of-conversion is 10ns. A plot of this is shown in Figure 7. The first falling edge is the clock assertion, and the second edge is the current output fed into a current-to-voltage converter implemented using a 100 MHz operational amplifier with a feedback shunt resistor.

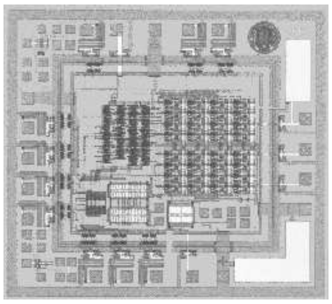

Figure 4: One 10-bit D/A (MOSIS 2.0μm, N-well)

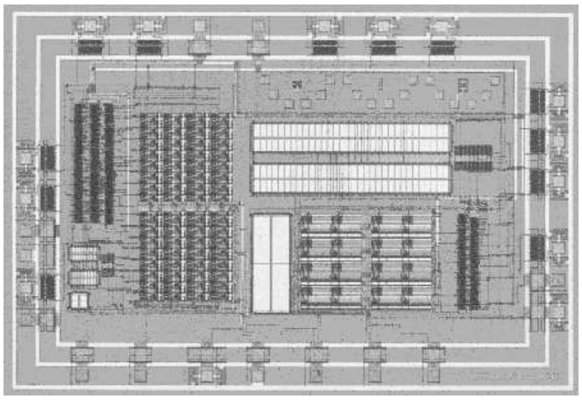

Figure 5: Two 10-bit D/A's (MOSIS 2.0μm, P-well)

5 Mismatch Extraction

In addition to its use during the synthesis phase, the behavioral simulator also provides a vehicle for fault diagnosis once the design has been fabricated, since the behavioral model used captures all of the essential mismatch effects. From the measured INL curves, the mismatch for each component (i.e. current sources) can be derived for all of the experimental chips. Using linear regression, the *best* estimate can be computed for each mismatch parameter assuming that the mismatch distributions are Gaussian.

For example, shown on the left in Figure 8 are the measured

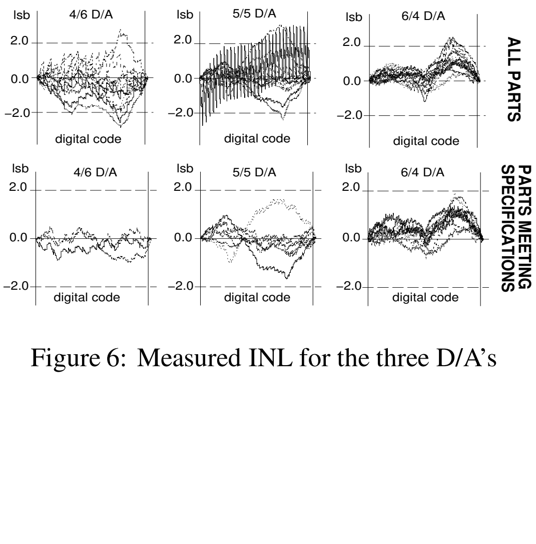

Figure 6: Measured INL for the three D/A's

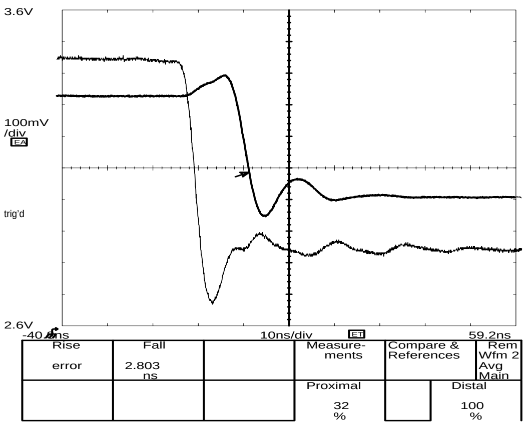

Figure 7: Falling Edge for the 10-bit 6/4 D/A

INL results for one of the 5/5 D/A's. On the the right is shown its estimated component mismatches. The validity of these results was checked by re-computing the INL based on these estimates and comparing them to the measured INL. The maximum difference between the curves was 0.08 lsb.

This component mismatch information can be used for fault diagnosis. For example, Figure 8 shows that most of the mismatches are due to the center "s1" (linear array) elements. Future designs can try to improve upon this.

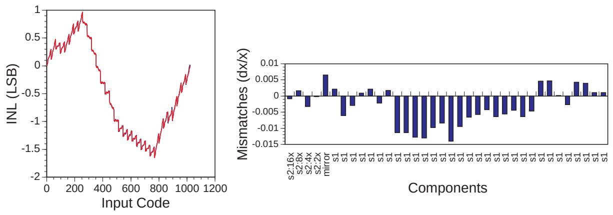

Figure 8: Measured INL and Component Mismatches

Component mismatches, in general, are caused by underlying basic statistical process variations. In this design, assumed were underlying process variations [11] [12] such as the flat band voltage variations, a_{VFB}, and the standard deviation of the MOS transistor widths and lengths. Using the component mismatch data measured at two reference currents and using a basic mismatch formula [13], the process mismatches can estimated. Table 3 shows the results. The close agreement between the assumption and the extracted results validates the assumptions made during design.

Constants	units	Assumed	5/5 D/A
$\sqrt{a_{\mathrm{VFB}}}$	V μm	0.0623	0.0555
σ_W	μm	0.05	0.0437

Table 3: Basic Statistical Process Variations

6 Conclusion

Following the top-down, constraint-driven paradigm, a complete design cycle for a class of D/A's has been performed. The cycle was begun with a set of high-level specifications and low-level assumptions on the technology, and was completed with the verification of both the design specifications and the gathered technology information from the finished products. The fabricated parts worked the first time. Table 4 gives an idea of the design times required for each of the design phases. The times shown in Table 4 are estimates on how long it would take to re-do each of these phases given a completely *new* set of design specifications (those found in Table 1).

Design Phase	Time Required
Behavioral model development	4 hours
High-level optimization	10 hours
Low-level synthesis	5 hours
Incorporation of new unit cell to layout tool	2 days
Layout generation	2 hours
Extraction/Verification	5 hours

Table 4: Design Times

Future work on the D/A design example include adding transient constraints into the design specifications and the introduction of new circuits such as ones for higher speed and ones for self-calibration based on the test results. Future work on the design scheme includes building larger systems. It is hoped that the design environment of these systems will also be improved, and that the methodology will be augmented as well, giving it eventually the power necessary to design quickly and effectively the large mixed-signal systems which are so prevalent today.

7 Acknowledgements

The authors would like to thank Micro Linear for providing design examples. This research has been partially supported by The National Science Foundation; The Intel Foundation; SRC (93-DC-008); and DARPA.

References

[1] H. Chang, A. Sangiovanni-Vincentelli, F. Balarin, E. Charbon, U. Choudhury, G. Jusuf, E. Liu, E. Malavasi, R. Neff and P. Gray, "A Top-down, Constraint-Driven Design Methodology for Analog Integrated Circuits," in *Proc. IEEE Custom Integrated Circuits Conference*, pp. 841–846, May 1992.

[2] G. Jusuf, P.R. Gray and A. Sangiovanni-Vincentelli, "A Performance-Driven Analog-To-Digital Converter Module Generator," in *Proc. IEEE Int. Symposium on Circuits and Systems*, pp. 2160–2163, May 1992.

[3] E. W. Y. Liu, H. C. Chang and A. L. Sangiovanni-Vincentelli, "Analog System Verification in the Presence of Parasitics using Behavioral Simulation," in *Proc. Design Automation Conference*, pp. 159–163, June 1993.

[4] H. Schouwenaars, D. Groeneveld and H. Termeer, "A Low-Power Stereo 16-bit CMOS D/A Converter for Digital Audio," in *Proc. IEEE International Solid-State Circuits Conference*, 1988.

[5] M. J. M. Pelgrom, A. C. J. Duinmaijer and A. P. G. Welbers, "Matching Properties of MOS Transistors," *IEEE Journal of Solid State Circuits*, vol. 24, pp. 1433–1440, October 1989.

[6] B. A. Murtagh and M. A. Saunders, "MINOS 5.1 User's Guide," Technical Report Rep. SOL 83-20R, Dept. of Operations Research, Stanford University, Stanford, CA, January 1987.

[7] D. Luenberger, *Linear and Nonlinear Programming*, Addison-Wesley Publishing Company, 2nd Ed., 1984.

[8] Y. Nakmura and T. Miki et. al., "A 10-b 70-MS/s CMOS D/A Converter," *IEEE Journal of Solid State Circuits*, vol. 26 no. 4, pp. 637–642, Apr 1991.

[9] M. J. M. Pelgrom, "A 10-b 50-Mhz CMOS D/A Converter with 75-Ω Buffer," *IEEE Journal of Solid State Circuits*, vol. 25, pp. 1347–1352, December 1990.

[10] Y .Nakamura, T. Miki, A. Maeda, H. Kondoh and N. Yazawa, "A 10-b 70-MS/s CMOS D/A Converter," *IEEE Journal of Solid State Circuits*, vol. 26, No. 4, pp. 637–642, April 1991.

[11] C. Michael and M. Ismail, "Statistical Modeling of Device Mismatch for Analog MOS Integrated Circuits," *IEEE Journal of Solid State Circuits*, vol. 27 no. 2, pp. 154–166, Feb 1992.

[12] K. R. Lakshmikumar, R. A. Hadaway and M. A. Copeland, "Characterization and modeling of Mismatch in MOS Transistors for Precision Analog Design," *IEEE Journal of Solid State Circuits*, vol. SC-21, n. 6, pp. 1057–1066, December 1986.

[13] P. R. Gray and R. G. Meyer, *Analysis and Design of Analog Integrated Circuits*, pp. 717–718, J. Wiley & Sons, 1977.