Simulation-based High-level Synthesis of Pipeline Analog-to-Digital Converters

J. Ruiz-Amaya, J.M. de la Rosa and M. Delgado-Restituto

Instituto de Microelectrónica de Sevilla, IMSE-CNM (CSIC),
Ed. CNM-CICA, Av. Reina Mercedes s/n, 41012 Sevilla, SPAIN.
Phone: +34955056666, Fax: +34955056686, E-mail: {ruiz|jrosa|mandel}@imse.cnm.es

***Abstract*–This paper presents a toolbox for the time-domain simulation and optimization-based high-level synthesis of pipeline analog-to-digital converters in MATLAB®. Behavioral models of building blocks, including their critical error mechanisms, are described and incorporated into SIMULINK® as C-compiled S-functions. This approach significantly speeds up system-level simulations while keeping high accuracy – verified with HSPICE – and interoperability of different subcircuit models. Moreover, their combined use with an efficient optimizer makes the proposed toolbox a valuable alternative for the design of broadband communication analog front-ends. As a case study, an embedded 0.13μm CMOS 12bit@80MS/s ADC for a PLC chipset is designed to show the capabilities of the presented tool.†**

I. INTRODUCTION

In recent years we are witnessing an unprecedented development of telecommunications, giving rise to multiple applications which span from portable wireless devices to broadband wireline modems. This trend is parallel to the exponential increase of the capabilities of digital CMOS circuits (favoured by the evolution of technologies towards deep submicron) prompting the integration of complete systems onto a single chip. In such Systems-on-Chip (SoC), most of the signal processing is carried out by digital circuitry, and the role of analog circuits basically reduces to implement the necessary signal conditioning and data conversion interfaces [1][2]. In spite of this apparently minor role, the design of high-performance analog circuitry (usually, in adverse digital-oriented technologies) most often represents an important bottleneck for a short time-to-market deployment [3]. This has motivated the interest for CAD tools which can optimize and shorten their synthesis procedure. This paper contributes to this topic and focuses on the synthesis of pipeline Analog-to-Digital (A/D) converters.

Pipeline A/D Converters (ADCs) have demonstrated to be a good alternative for interfaces requiring medium-high resolution at video-range conversion rates and beyond [4][5]. This scenario is becoming commonplace in modern communication applications, like VDSL and PLC, where signals are sampled at 40-80MSamples/second (MS/s) and must be converted to 12-14 bit digital outputs [1]. Although such data rates are easily achievable with flash or folding/interpolation ADCs, their area and power consumptions become so significant at resolutions beyond 10 bit, that makes their deployment not competitive at least for SoC applications [6][7]. On the other hand, the use of ΣΔ modulator topologies is neither a viable solution for high signal bandwidths (beyond 15MHz) because of the prohibitive sampling frequencies which are required to achieve medium-high resolution [8][9].

Several tools for the design and verification of pipeline ADCs have been reported in the open literature [10]-[14]. Most of them are based on an iterative optimization procedure in which the design problem is translated into a cost function minimization problem that can be evaluated through numerical methods. Evaluation of the cost function is normally performed by means of equations [11][12][14], so that very short computation times are obtained. As a drawback, this approach results in closed tools because equations must be changed every time the topology is changed.

This paper aims at palliating this problem by using simulation instead of equations for cost function evaluation. To this end, a complete toolbox for the high-level synthesis of arbitrary pipeline ADCs has been developed in the MATLAB® environment [15]. The embedded simulator uses SIMULINK® S-functions [16] to describe all required subcircuits including their main non-idealities. This approach considerably reduces computational costs as compared to using standard library blocks as in [13]. For all subcircuits, the accuracy of the behavioral models has been verified by electrical simulations using HSPICE. Additionally, the toolbox includes an efficient hybrid optimizer which uses statistical techniques for design space exploration and deterministic techniques for fine tuning [17]. Other important features of the platform are a friendly Graphical User Interface (GUI), high flexibility for tool expansion†† and wide signal processing capabilities [15].

As a case study on the use of the proposed synthe-

†. This work has been supported by the MEDEA+ (A110 MIDAS) Project.

††. Indeed, the toolbox has been already extended to cover other converter topologies, such as full flash ADCs and current-steering Digital-to-Analog Converters (DACs).

sis tool, a 0.13μm CMOS 12-bit@80MS/s A/D interface for PLC is synthesized and designed. Different experiments show the effectiveness of the proposed methodology.

II. BEHAVIORAL MODELING OF PIPELINE ADCs USING SIMULINK® C-CODED S-FUNCTIONS

Fig.1(a) shows the conceptual block diagram of a generic pipeline ADC, consisting of an arbitrary cascade of k stages and a Sampled-and-Hold (S/H) circuit at the front [1][2][11]. Each stage resolves partial code words of length n_j, $j = 1, \ldots, k$, which are all re-ordered and combined at the digital correction block to obtain the N bit output of the converter. The inner structure of a pipelined stage comprises four blocks, as illustrated in Fig.1(b): a flash sub-ADC with $N_j \le 2^{n_j}$ output codes, a sub-DAC with N_j output levels, a substractor, and a S/H residue amplifier with gain G_j. The latter three blocks are implemented in practice by a single subcircuit which is often referred to as Multiplying DAC (MDAC).

All the critical blocks in the pipeline architecture, namely, S/H circuit, sub-ADCs and MDACs, have been modelled and coded in the proposed toolbox as described below.

A. S/H circuit

Fig.2(a)† shows the conceptual schematic of a typical S/H block topology which operates with two non-overlapped clock phases. Its model in the proposed toolbox includes the most critical error mechanisms which are computed according to the flow diagram in Fig.3. The flow graph has two branches corresponding to the two clock phases. During the sampling phase, the input-equivalent thermal noise, (Δv_n), is calculated and added to the voltage stored at the sampling capacitor C_s. This is computed taking into account the non-linear switch on-resistance effects (r_{on}). Next, an iterative procedure is started to calculate the output voltage V_o by solving the equivalent circuit of Fig.2(b), which models the effects of finite and non-linear opamp DC-gain ($A_v = G_m(V_a)/g_o$), opamp offset (E_{os}), non-linear sampling capacitor, transient response (comprising both linear incomplete settling and slew-rate limitation), parasitic capacitances (C_p, C_b, C_{load}), output range limitations and charge injection error. During the hold phase, a similar procedure is applied to solve the equivalent circuit shown in Fig.2(c). As the value of state signals are important only at the end of each clock phase, a set of finite difference equations have been generated to describe the operation of real S/H circuits.

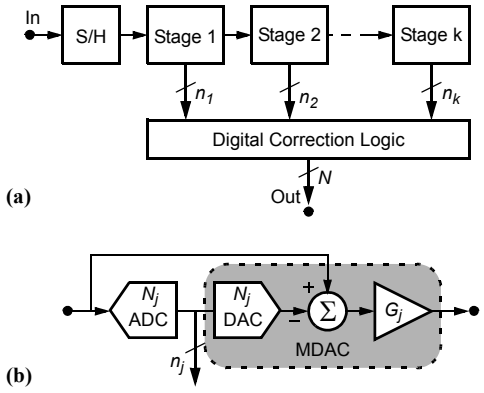


Fig. 1: Generic pipeline ADC architecture. (a) Conceptual block diagram; (b) structure of a single stage.

†. For illustration purposes, schematics are shown in its single-ended version, although actually the fully-differential structures have been modelled.

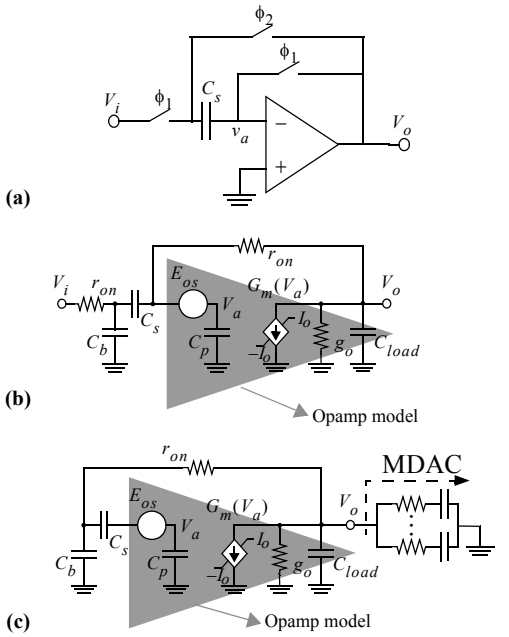


Fig. 2: S/H (a) schematic, (b) equivalent circuit in sampling phase and (c) equivalent circuit in hold phase.

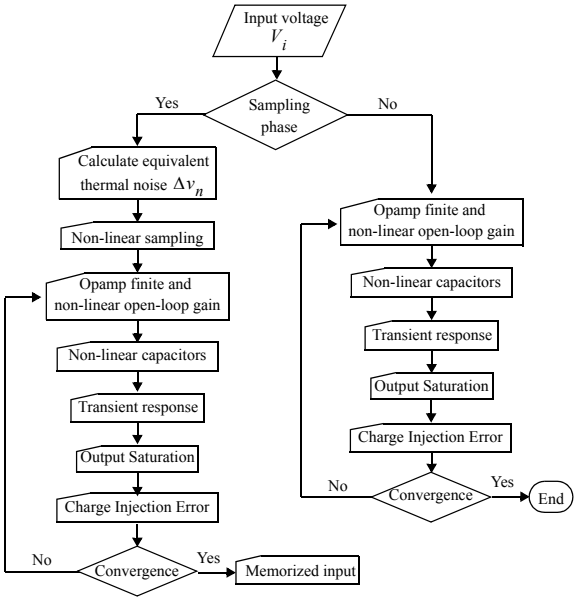


Fig. 3: Flow diagram of the S/H model.

The above procedure has been codified in C and incorporated as an S-function into the SIMULINK® environment [16]. This approach allows to drastically speed up the simulation CPU-time† (up to 2 orders of magnitude) as compared to previous approaches – based on the use of SIMULINK® elementary blocks [13]. Moreover, S-functions are more suitable for implementing a more detailed description of the circuit. As an example of the accuracy of the behavioral model, Fig.4 compares the transient response for a constant input voltage by using HSPICE and our model showing a good agreement.

B. Pipeline stage: flash sub-ADC and MDAC

Other important building blocks of pipeline ADCs are the flash ADC and MDAC circuits.

The generic schematic of the flash quantizer is shown in Fig.5. Both binary- and termometric-coded flash have been modelled considering their most critical limiting factors, namely: resistor mismatches and non-linearities and comparator errors (offset, hysteresis and dynamics).

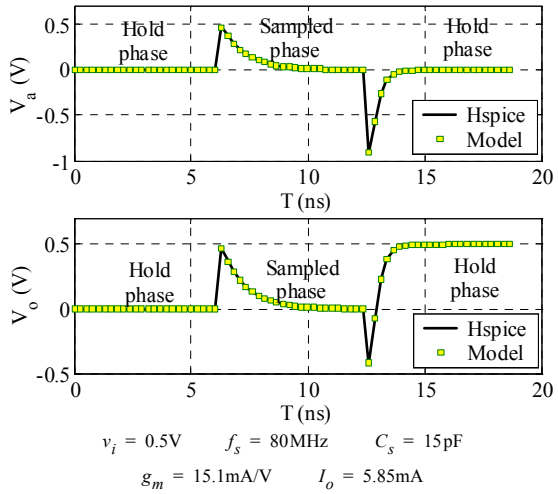


Fig. 4: Transient response of the S/H: comparison between HSPICE and our behavioral model.

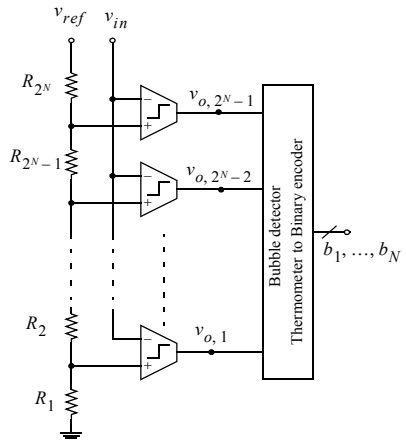


Fig. 5: Flash quantizer schematic.

†. A 32738-samples simulation takes 2-3 seconds. All simulations shown in this paper were done using a PC with an AMD XP2400 CPU@2GHz @512MB-RAM.

As far as the MDAC is concerned, critical aspects affecting linearity and transient response have been included such as: capacitance mismatch and non-linearities ($C_u = C_{uo}(1 + cnl_1 v + cnl_2 v^2)$), finite switch -on resistances (r_{on}) and opamp errors (offset (v_{off}), finite and non-linear DC gain, thermal noise, incomplete settling and slew-rate). The flow diagram of the operation of the MDAC is similar to the one S/H, but now the equivalent circuits used to evaluate the internal nodes are the ones which are shown in Fig.6(b) and Fig.6(c). For the opamp, a two-pole model (g_m, C_{load}, g_o, g_{mh}, C_h, g_{oh}) using Miller compensation (C_c) has been developed. This model matches very well with HSPICE predictions as shown in Fig.7.

III. HIGH-LEVEL SYNTHESIS TOOLBOX

The models described above have been included in a SIMULINK®-based simulator. This simulator, used for performance evaluation, is combined with a statistical optimizer for design parameter selection as described below.

A. Conceptual diagram of the toolbox

For synthesis purposes, deterministic optimization methods, like those available in the MATLAB® standard distribution [15], are not suitable because initially designers may have little or no idea of an appropriate design point. Therefore, the optimization procedure is quickly trapped in a local minimum. For this reason, we developed an optimizer which combines an adaptive statistical optimization algorithm inspired in sim-

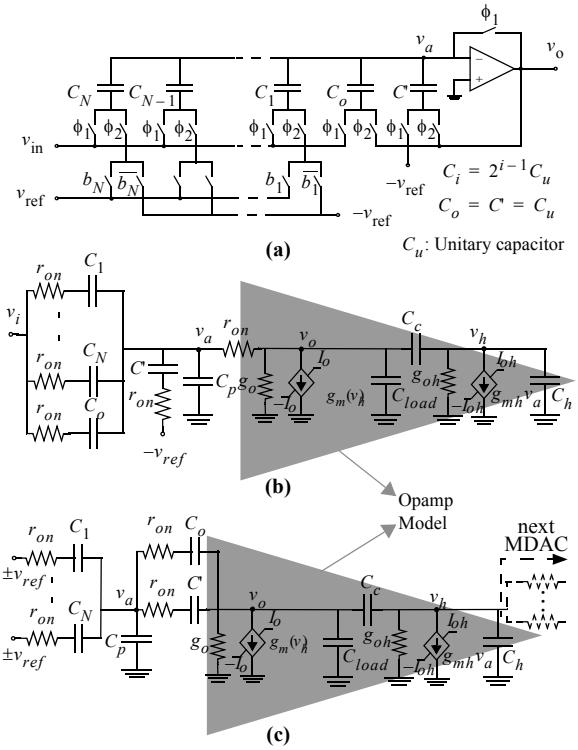
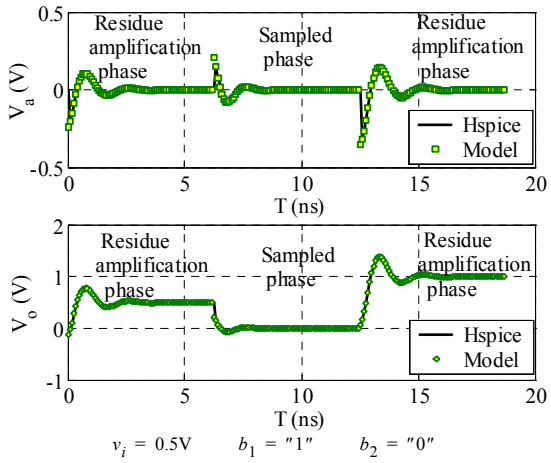


Fig. 6: MDAC (a) schematic, (b) equivalent circuit in sampling phase and (c) equivalent circuit in residue amplification phase. (Opamp offset not shown for simplicity).



v_i = 0.5V b_1 = "1" b_2 = "0"

f_s = 80MHz C_u = 0.5pF g_m = 0.5mA/V I_o = 4mA

g_{mh} = 20mA/V I_{oh} = 3mA C_c = 0.1pF

Fig. 7: Transient response of a 2-bit MDAC considering a constant input voltage: comparison between HSPICE and two-pole behavioral model.

ulated annealing (local minima of the cost function can then be avoided) with a design-oriented formulation of the cost function (which accounts for the ADC performances). This optimizer has been integrated in the MATLAB®/SIMULINK® platform by using the MATLAB® engine library [15], so that the optimization core runs in background while MATLAB® acts as a computation engine.

Fig.8 shows the flow diagram of the optimizer where starting from an ADC topology, e.g., an arbitrary ADC whose design parameters (building block specifications, resolution per stage, etc.) are not known and arbitrary initial conditions, a set of design parameter perturbations is generated. With the new design parameters, a set of simulations are done to evaluate the circuit performance. From the simulation results, it automatically builds a cost function (that has to be minimized). The type and value of the perturbations as well as the iteration acceptance or rejection criteria depend on the selected optimization method. The optimization process is divided into two steps. The first step explores the design space by dividing it into a multi-dimensional coarse grid, resulting in a mesh of hypercubes (*main optimization*). A statistical method is usually applied in this step. Once the optimum hypercube has been obtained, a final optimization – using a deterministic method – is performed inside this hypercube (*local optimization*).

B. Implementation in the MATLAB® environment

The proposed tool has been conceived as a MATLAB® toolbox for the simulation and synthesis of Nyquist-rate data converters, including flash, pipeline ADCs and current-steering DACs. Fig.9 shows some parts of the toolbox comprising a GUI to allow the designer to browse through all steps of the simulation, synthesis and post-processing of results. By using this

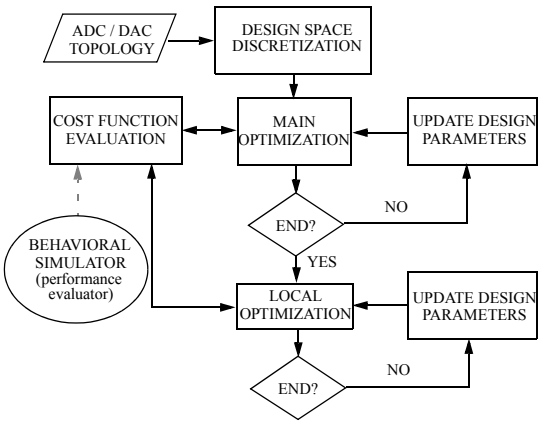


Fig. 8: Operation flow of the optimization core.

GUI, the designer can either open an existing ADC or DAC architecture or create a new one in the SIMULINK® platform by connecting the building-block available in the toolbox. After a simulation is done, different figures such as output spectrum, in-band noise power, INL, harmonic distortion, etc., can be computed through the analysis/data processing menu. High-level synthesis is started from the synthesis menu, where constraints, performance specifications, design parameters, optimization algorithms, etc., can be specified. Then, the optimization core starts the exploration of the design space to find out the optimum solution by using the simulation results for performance evaluation.

IV. CASE STUDY: A 12-bit@80MS/s PIPELINE ADC FOR PLC

In order to illustrate the capabilities of the proposed toolbox for the simulation and synthesis of Nyquist-rate data converters, the high-level design of a 0.13µm CMOS 12-bit@80MS/s A/D interface for PLC will be described. The specifications are shown in Table 1 and the objective consists on achieving those specifications with the minimum requirements in terms

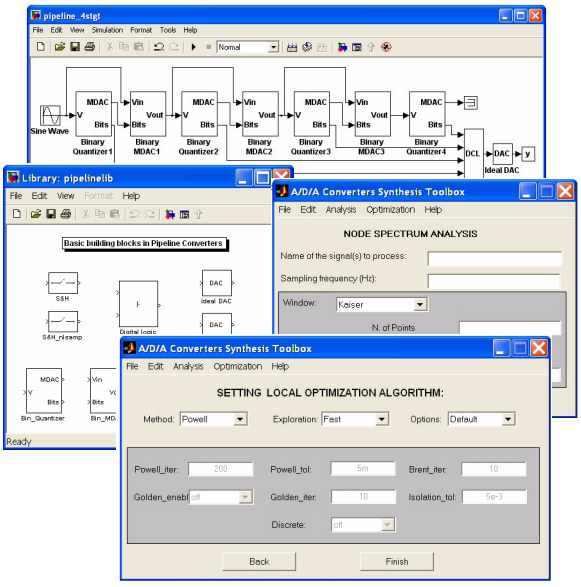


Fig. 9: Some parts of the data converter synthesis toolbox.

of power consumption and area. In addition, this design is planned to be implemented without using calibration. For that reason, capacitor mismatch is a critical issue. In fact, this limitation forces us to optimize the capacitor sizes not only in terms of thermal noise and dynamics considerations but also in terms of minimum capacitance area needed to achieve the required mismatch. Another critical parameter considered in the optimization is the resolution per stage. Taking into account these factors, a wide exploration of several architectures has been carried out with the proposed synthesis toolbox.

The optimum architecture was a 7-stage pipeline with the next resolution-per-stage: 3-2-3-2-3-3-2. Redundant sign digit coding was used in order to relax the requirements for the comparators in the flash quantizers.

The results of the high-level synthesis for the first stages as well as the requirements for the opamps are summarized in Table 2. The optimization procedure for a given architecture required about 3000 iterations of 16384-clock cycles taking about 2 hours of CPU-time.

Table 3 shows a summary of the converter performance from where it can be deduced that the specifications are fully satisfied. MonteCarlo analysis has been carried out taking into count both capacitor and resistor mismatch in order to characterize typical and worst cases of different figures. As an illustration, Fig.10(a) shows an histogram of a MonteCarlo analysis comprising 1000 iterations and Fig.10(b) shows the output spectrum for the MTPR (case (a) in Table 1). Finally, the effective resolution is illustrated in Fig.11, by plotting an histogram of the MonteCarlo analysis (Fig.11(a)) and an output spectrum corresponding to a full-scale input tone at f_i = 34MHz (Fig.11(b)).

CONCLUSIONS

A complete MATLAB® toolbox for the high-level synthesis and verification of pipeline ADCs has been described. The combination of an efficient SIMULINK®-based time-domain behavioral simulator and an advanced statistical optimizer allows to efficiently map system-level specifications into building-block specifications in reasonable computation times. Critical design issues such as the resolution per

TABLE 1. Specifications for the A/D converter

Specifications	
12bits@80MS/s	
Multi-Tone Power Ratio (MTPR) (a) 15 tones on-1 tone off (b) 120 tones on - 8 tones off (c) 240 tones on - 16 tones off	≥ 56dB
Effective Number Of Bits (ENOB)	≥ 9.2 bits
Differential Input Range	2 Vp-p
Power Supply	3.3V

TABLE 2. High-level synthesis for the A/D converter

Block		Parameter	Requirement
S/H		Sampling capacitor (pF)	<5.5
		Switch on-resistance (Ω)	<60
	Opamp	Eq. load (pF)	11.8
		Slew-Rate (V/μs)	>932
		GB (MHz)	>135
		DC-gain (dB)	>60
		Noise PSD (nV/$\sqrt{\text{Hz}}$)	<6
1st. stage (3 bits)	Flash quant.	Comparators Offset (mV)	<100
		Comparators Hysteresis (mV)	<100
	MDAC	Switch on-resistance (Ω	<40
	Opamp	Eq. load (pF)	32.9
		Slew-Rate (V/μs)	>107
		GB (MHz)	>218
		DC-gain (dB)	>81
		Noise PSD (nV/$\sqrt{\text{Hz}}$)	<3.3
2nd. stage (2 bits)	Flash quant.	Comparators Offset (mV)	<100
		Comparators Hysteresis (mV)	<100
	MDAC	Switch on-resistance (Ω)	<40
	Opamp	Eq. load (pF)	22.2
		Slew-Rate (V/μs)	>180
		GB (MHz)	>201
		DC-gain (dB)	>72
		Noise PSD (nV/$\sqrt{\text{Hz}}$)	<8
3rd. stage (3 bits)	Flash quant.	Comparators Offset (mV)	<100
		Comparators Hysteresis (mV)	<100
	MDAC	Switch on-resistance (Ω)	<60
	Opamp	Eq. load (pF)	15
		Slew-Rate (V/μs)	>107
		GB (MHz)	>146
		DC-gain (dB)	>70
		Noise PSD (nV/$\sqrt{\text{Hz}}$)	<14
4th. stage (2 bits)	Flash quant.	Comparators Offset (mV)	<100
		Comparators Hysteresis (mV)	<100
	MDAC	Switch on-resistance (Ω)	<200
	Opamp	Eq. load (pF)	13.4
		Slew-Rate (V/μs)	>112
		GB (MHz)	>143
		DC-gain (dB)	>66
		Noise PSD (nV/$\sqrt{\text{Hz}}$)	<63

TABLE 3. Simulation results.

Results	Typical case	Worst case
ENOB (bits) (f_i@34MHz)	10.13	9.7
INL(12bits) (lsb)	-2.60/2.83	-
DNL(12bits) (lsb)	-0.59/0.66	-
MTPR (case a)	59.62	55.96
MTPR (case b)	61.48	57.95
MTPR (case c)	62.15	60.16
Power Consumption (mW)	230	

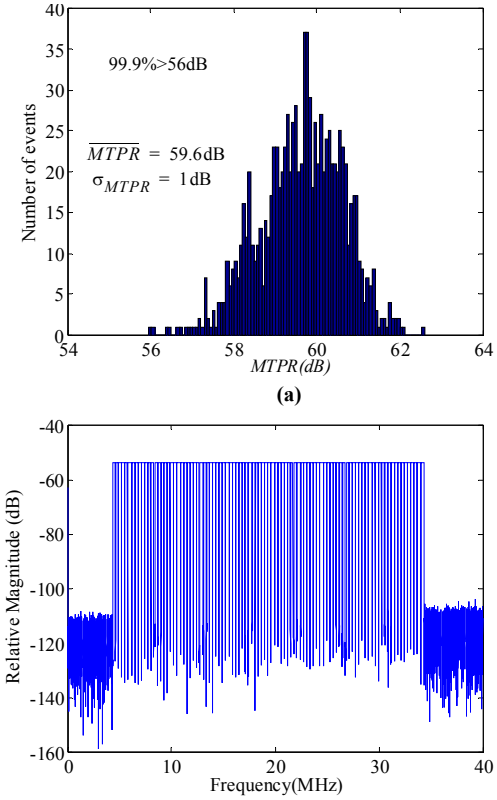


Fig. 10: (a) MTPR (case (a) in Table 1): (a) histogram of a Monte-Carlo analysis; (b) output spectrum.

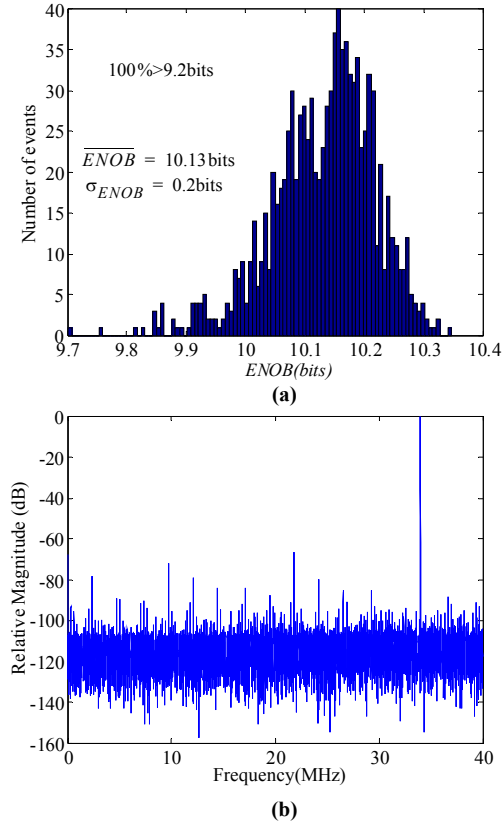


Fig. 11: *ENOB*: (a) histogram of a MonteCarlo analysis; (b) output spectrum for a full-scale amplitude input tone at 34MHz.

stage are optimized in terms of power consumption and silicon area. As a case study, a 0.13µm CMOS 12bit@80MS/s ADC for PLC front-end has been designed and analysed using the proposed toolbox.

REFERENCES

[1] A. Rodríguez-Vázquez, F. Medeiro and E. Janssens (Editors): *CMOS Telecom Data Converters*. Kluwer Academic Publishers, 2003.

[2] M. Gutavsson, J. J. Wikner and N. N. Tan: *CMOS Data Converters for Communications*. Kluwer Academic Publishers, 2000.

[3] G. G. E. Gielen and R. A. Rutenbar: "Computer-Aided Design of Analog and Mixed-Signal Integrated Circuits", *Proceedings of the IEEE,* Vol. 88, pp. 1825-1852, December 2000.

[4] A. Shabra and H.-S. Lee: "Oversampled Pipeline A/D Converters With Mismatch Shaping". *IEEE Journal of Solid-State Circuits*, Vol. 37, pp. 566-578, May 2002.

[5] W. Yang, D. Kelly, I. Mehr, M. Sayuk and L. Singer: "A 3-V 340-mW 14-b 75-Msamples/s CMOS ADC with 85-dB SFDR at Nyquist Input". *IEEE Journal of Solid-State Circuits*, Vol. 36, pp. 1931-1936, Dec. 2001.

[6] K. Uyttenhove, J. Vandenbussche, E. Lauwers, G. G. E. Gielen and M. S. J. Steyaert: "Design Techniques and Implementation of an 8-bit 200-MS/s Interpolating/Averaging CMOS A/D Converter". *IEEE Journal of Solid-State Circuits*, Vol.38, pp. 483-494, March 2003.

[7] K. Uyttenhove and M. S. J. Steyaert: "A 1.8-V 6-bit 1.3-GHz Flash ADC in 0.25-µm CMOS". *IEEE Journal of Solid-State Circuits*, Vol. 38, pp. 1115-1122, July 2003.

[8] Y. Geerts, M. Steyaert, and W. Sansen, "A High Performance Multibit ΔΣ CMOS ADC". *IEEE Journal of Solid-State Circuits*, Vol. 35, pp. 1829-1840, December 2000.

[9] A. Di Giandomenico, S. Paton, A. Wiesbauer, L. Hernández, T. Pöster and L. Dörrer: "A 15-MHz Bandwidth Sigma-Delta ADC with 11 bits of Resolution in 0.13µm CMOS". *Proc. of the 2003 European Solid-State Circuits Conference*, pp.233-236.

[10] E. Liu and A. Sangiovanni-Vincentelli: "Verification of Nyquist Data Converters Using Behavioral Simulation". *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 14, pp. 493-502, April 1995.

[11] J. Goes, J. C. Vital and J. E. Franca: *Systematic Design for Optimization of Pipelined ADCs*. Kluwer, 2001.

[12] P. T. F. Kwok and H. C. Leung: "Power Optimization for Pipeline Analog-to-Digital Converters". *IEEE Trans. on Circuits and Systems - Part II*, pp. 549-53, May 1999.

[13] E. Bilhan, P. C. Estrada-Gutierrez, A. Y. Valero-Lopez and F. Maloberti: "Behavioral Model of Pipeline ADC by Using Simulink". *Proc. of 2001 Southwest Symposium on Mixed-Signal Design*, pp. 147-151.

[14] R. Lotfi, M. Taherzadeh-Sani, M. Yaser-Azizi and O. Shoaei: "Systematic Design for Power Minimization of Pipelined Analog-to-Digital Converters". *Proc. of 2003 IEEE Int. Conf. Computer-Aided Design*, pp. 371-374.

[15] The MathWorks Inc.: "MATLAB®. The Language of Technical Computing. V.6.5 R.13", July 2002.

[16] The MathWorks Inc.: "Simulink®. Model-based and System-based Design. V.5 R.13", July 2002.

[17] F. Medeiro, B. Pérez-Verdú, and A. Rodríguez-Vázquez: *Top-Down Design of High-Performance Modulators*, Kluwer Academic Publishers, 1999.