Closed-loop Simulation Method for Evaluation of Static Offset in Discrete-Time Comparators

A. J. Ginés, E. Peralías, G. Leger, A. Rueda, *Senior Member, IEEE*
Instituto de Microelectrónica de Sevilla IMSE-CNM, (CSIC and Universidad de Sevilla), Sevilla, Spain
E-mails: {gines,peralias,leger,rueda}@imse-cnm.csic.es

***Abstract*—This paper presents a simulation-based method for evaluating the static offset in discrete-time comparators. The proposed procedure is based on a closed-loop algorithm which forces the input signal of the comparator to quickly converge to its effective threshold. From this value, the final offset is computed by subtracting the ideal reference. The proposal was validated using realistic behavioral models and transistor-level simulations in a 0.18µm CMOS technology. The application of the method reduces by several orders of magnitude the number of cycles needed to characterize the offset during design, drastically improving productivity.**

Keywords—comparator offset evaluation; discrete-time; Flash-ADC; simulation-based techniques.

I. INTRODUCTION

Discrete-time comparators are basic building blocks in mixed-signal systems, almost ubiquitous in all analog-to-digital (ADC) architectures, such as Flash, Pipeline, SAR (Successive-Approximation-Register) or Sigma-Delta. During the design phase of this kind of comparators, the determination of their static offsets at simulation level is a fundamental but tedious task, especially when mismatch and PVT (process, voltage and temperature) variations must be analyzed. Different aspects such as the effective input level sampled by the comparator, different clock phases used in the structure and variability range in the nominal threshold should be properly established in the simulation test set-up. Usually, a slow ramp or triangular stimulus is considered. Their amplitudes have to span (with enough security margin) the maximum and minimum expected threshold shift, which implies long transient simulations during verification, affecting to the design productivity. For instance, dynamic latch comparators could suffer offset errors up to 200mV in CMOS technologies [1]-[2], and hence, a range of 400mV would be required (*span* = 400mV). In these conditions, reaching a 1mV precision during PVT variation characterization (*accuracy* = 1mV) would need 400 cycles (=*span/accuracy*) per simulation run.

This work presents a closed-loop simulation technique for evaluating the offset during design phase. The method is based on the inclusion of the comparator in a binary search algorithm which forces its input signal to achieve its effective threshold with fast convergence. Given the wanted ideal threshold, the offset is obtained by a simple subtraction. The search algorithm does not need any estimation of the threshold span, or knowing its ideal location to provide a fast convergence to the actual value. Considering that we approximately know the range of the mismatch and PVT variability of the comparator threshold, the number of clock cycles is reduced to $N_c \approx \log_2(span/accuracy)$, drastically relaxing the simulation time with respect to the classical ramp-stimulus analysis. This means for the previous example, that only 9 cycles would be required (2 orders of magnitude lower).

Contents in this paper are distributed as follows. Section II shows the basic concepts and followed notation. It also makes revision of the standard test ADC methods for offset characterization [3]-[4], since comparators can be understood as 1-bit ADC. Section III presents the proposed method. In Section IV, the behavioral and transistor levels simulations results are presented. Finally, conclusions are drawn.

II. OFFSET IN DISCRETE-TIME COMPARATORS

The basic functionality of a discrete-time comparator consists in discriminating if the sampled analog input signal, $x[n] = x(nT)$, where n and T are the sampling index and clock period respectively, is above or below a given threshold l. This threshold is defined as the analog input level at which the output bit commutes from low to high logical values. When $x[n] \geq l$, the output code b[n] takes the logical value 1. If $x[n] < l$, b[n] has the logical value 0. In a compact manner, b[n] is expressed as,

$$b[n] = (x[n] \geq l) \quad \text{with} \quad x[n] = x(nT) \quad ; \quad l = l_{ideal} + off \qquad (1)$$

where the offset is the threshold shift with respect to its ideal value l_{ideal}, i.e. $off = l - l_{\text{ideal}}$.

The dependence of the output code $b[n]$ on the clock introduces additional difficulties in determining the offset during design, with respect to the continuous-time comparator counterpart, since a) the DC sweep simulations are not any more valid, b) offset evaluation must be performed by relatively long transient simulations, and c) the accuracy in the estimation depends on the relationship between the input stimulus (typically, ramps or triangular signals) and the sampling clock edge. Fig. 1 shows this effect in a simple example of a triangular input signal considering that the sampling instant coincides with the clock rising edge. Note that, the uncertainly Δ in the determination of the comparator offset is directly related to the ramp slope, $s_{\text{ramp}} = \Delta / T$. This dependence produces that given a threshold uncertainly $|off|_{\max}$, the minimum number of clock cycles to achieve a wanted accuracy Δ becomes,

$$N_c = span / \Delta = 2\left|off\right|_{\max} / \Delta \quad ; \quad |off|_{\max} = \max|l - l_{ideal}| \qquad (2)$$

where the factor 2 arises from allowing positive and negative values in the offset, $off = l - l_{\text{ideal}}$. From this expression, the minimum time per simulation is $t_{sim} \geq N_c T$, making evident

This work was supported in part by the Andalusian Government project P09-TIC-5386 and the Spanish Government project TEC2011-28302, both of them cofounded by the European FEDER program.

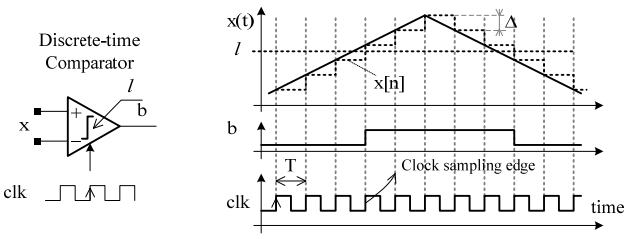

Fig. 1. Effect of apparent offset in Discrete-time Comparators.

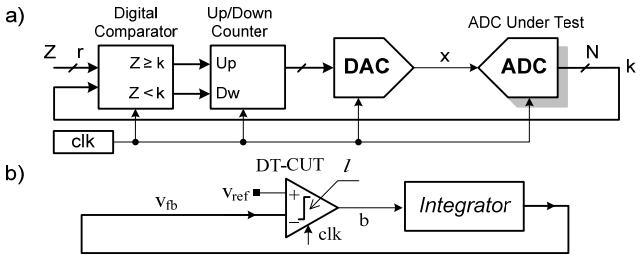

Fig. 2. a) Offset evaluation based on the mixed-signal implementation of the servo-loop method; b) application for offset estimation in [6].

that the classical analysis of the offset is not computationally efficient, and different test strategies need to be developed.

The following sub-sections summarize the existing simulation-based methods using both open-loop [5] and close-loop [6] configurations. These methods are directly based on the standard test methods for ADC static experimental characterization [3]-[4], taking into account that a comparator is by definition a 1-bit ADC. Although [3]-[4] have indubitable advantages due to transient random-noise immunity, they result inefficient in terms of computation time in a well-controlled simulation environment where thermal and flicker noise are disabled in transient simulations. This drawback is overcome in our technique by embedding the comparator under test in a binary search loop which shows fast convergence to its actual offset error. Simultaneously to the publication of our proposal, a close-loop technique to calibrate the offset in the decision comparator of a SAR-ADC has been presented in [7].

A. Statistical Evaluation of Offset (*Open-loop*)

Static characterization of ADCs is usually performed by histogram-based methods [3]. These techniques traditionally use an analog signal with known amplitude probability function (such as sinusoidal or ramp signals) and with greater linearity than the ADC under test. Based on the difference between the ideal and measured output code probabilities in an open-loop configuration, the method obtains a statistical measurement of the ADC representative code transitions l_k in the input-output characteristic.

In the standard histogram method, the estimation of the input signal voltage parameters —amplitude (A) and offset v_{os} — is often performed using exclusively the information of the ADC output code k with resolution N >1. Obviously, for a comparator, i.e. N = 1, since this digital approach is not any more valid, the parameters of the input stimulus must be extracted in the analog domain. This could be limiting in the laboratory, but in simulation set-up a well-known stimulus can be considered. Taking into account this information, the comparator offset is obtain using a simple statistical processing of the comparator output bit b. For a triangular input signal [3], the method estimates the output code probability, p_1 when b = 1, by the relative frequency of the output occurrence. Any deviation from its ideal ½ value is directly caused by an offset, which can be statistically evaluated as follows,

$$off = 2A \cdot (1/2 - p_1) + v_{os} - l_{ideal} \qquad (3)$$

The main disadvantages for this approach in a simulation environment [5] are on one hand, that special care should be paid to the analogue input stimulus to assure its statistical properties (uniformity for ramp and triangular), and other hand, that due to statistical processing a big number of samples are needed. Actually, given an accuracy Δ, the number of samples could exceed those needed in (2) for the ramp input stimulus, and hence, it is not recommended for simulation set-up.

B. Servo-loop Evaluation of Offset (*Closed-loop*)

A basic configuration of the servo-loop test method is illustrated in Fig. 2a. A feedback loop is created that forces the N-bit ADC to any desired code transition edge. The ADC input test stimulus may be generated by an M-bit DAC (digital-to-analogue converter) with $M > N$, as illustrated, and additional low-pass filtering may be incorporated. Alternatively, the feedback signal can be generated by an analogue integrator which integrates either a positive or negative reference voltage depending on the comparator output [4]. In any case, the ADC output code k is compared to some value Z. When the ADC output is smaller than the value Z, the DAC input will be incremented by the Up/Down counter. When the ADC output reaches the value Z, it will decrement the DAC input. In the quiescent state, the ADC input will oscillate around the particular code transition edge, and the value can be measured as the average voltage on the ADC input. All code transition edges can be tested by incrementing the value Z from 1 to 2^{N-1}.

In the case of a comparator, i.e. a 1-bit ADC, the scheme is simplified since just a single transition (the comparator threshold) is defined. Using this idea, a simulation-based approach was presented in [6]. The method is a direct translation of the servo-loop with a continuous-time integrator in the feedback loop, as shown in Fig. 2b. A disadvantage of this proposal is that extra analog building blocks are introduced in the simulation set-up (resistors, gm-C integrator) with the consequent penalty in terms of simulation time. Independently of the integrator architecture, the servo-loop approach shows a slow convergence to the actual threshold value when mismatch and PVT variations are considered. Actually, the number of samples needed to achieve convergence in a simulation environment could be sensibly greater than those predicted by (2), since the updating step is in the order of Δ, and extra time should be allocated to let the analog averaging filter to settle. In conclusion, this method could be ineffective in terms of computation resources in a simulation environment.

III. PROPOSED CLOSED-LOOP METHOD

To reduce the simulation time during the offset evaluation, we have developed a closed-loop method based on a binary search algorithm which shows a fast convergence to the actual comparator threshold. The search algorithm does not need any

estimation of the threshold span, or knowing the ideal location of the threshold to drastically reduce the number of clock cycles. The key idea of the method is the inclusion of the discrete-time comparator under test (DT-CUT) as the decision block in a SAR-loop.

In the SAR ADC architecture (Fig. 3), the analog input digitalization is iteratively performed by comparing the input signal with a reference, which is generated by an internal DAC, according to a digital SAR control unit. Conceptually, the SAR ADC is formed by a *SH*, an internal DAC and a control unit which performs the binary search. If a binary weighted DAC is considered, the number of cycles to achieve convergence is in the order of the resolution *N* of the converter. In this topology, the feedback voltage v_{fb} becomes an analog estimation of the sampled analog input x_s. Taking into account the sign criterion in the figure, the comparator output is,

$$out[n] = (v_{fb}[n] - x_s) \geq l \tag{4}$$

from which the final value of v_{fb} is derived in the form,

$$v_{fb}[N] = x_s + l + \varepsilon_\Delta \tag{5}$$

where ε_Δ is the quantization error, evaluated from the full-scale input signal *FS* and number of bits *N*, as follows,

$$|\varepsilon_\Delta| \leq \frac{\Delta}{2} \quad ; \quad \Delta = \frac{FS}{2^N} = \frac{\gamma(Rp - Rn)}{2^N} \quad \text{with} \quad \gamma = \begin{cases} 1: & \text{Single} \\ 2: & \text{Differential} \end{cases} \tag{6}$$

where the full scale *FS* is given by the DAC references (R_p,R_n) with a factor γ depending on single or differential mode.

Since we are interested in the location of threshold *l*, the external input signal x_s in (5) is not need, i.e. $x_s = 0$, and neither the SH. Taking into account this modification, Figure 4 shows the conceptual scheme for our proposed simulation set-up. It comprises the comparator under simulation and a *VIS* (virtual instrument for simulation) block that resembles the functionality of the SAR and DAC blocks in the closed-loop approach, but it is implemented in any high-level description language. This *VIS*: a) has a fully functional description, b) does not contain any circuit-level device, and hence, c) has a negligible contribution to the total simulation time. As highlighted at the introduction, the search algorithm does not need any estimation of the threshold span, or knowing the ideal location of the threshold to provide a fast convergence to the actual value. Considering we approximately know the range of the mismatch and PVT variability of the comparator threshold, the initial algorithm seed ($v_{fb}[0] = l_{ideal}$, the center of the search span) and range (*FS* = *span*) can be defined. In this case, the number N_c of clock cycle coincides with the resolution *N* of the SAR-ADC counterpart. Taking into account the second term in (6), it reduces to:

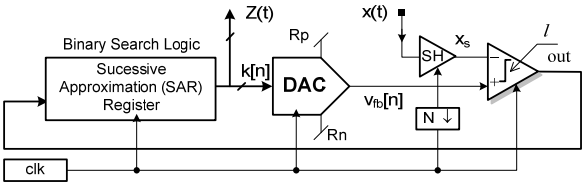

Fig. 3. Simplified block diagram of SAR-ADC.

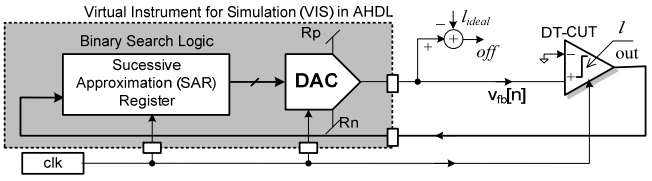

Fig. 4. Conceptual scheme of the proposed closed-loop method with fast convergence speed.

$$N_c = N = \lceil \log_2(FS/\Delta) \rceil = \lceil \log_2(span/accuracy) \rceil \tag{7}$$

where ⌈ ⌉ is the ceiling operator, drastically relaxing the simulation time with respect to the previous approaches. If the variability information is not available, only a few extra cycles are needed (typically, 3 or 4) to resolve the actual estimation with enough accuracy. For instance, if a comparator has a 200mV offset uncertainly (span = 400mV), but we do not have this information in advance, we can always use technological limit to establish the *VIS* span ($FS = FS_{tech}$). For a technological node with V_{DD} =1.8V, a maximum range in differential mode would be $FS_{tech} = 2\alpha \cdot V_{DD}$, where α is an architectural-dependent factor, typically ranging from 0.5 to 1. In a 1.8V 0.18µm CMOS process with FS_{tech} = 2V (α = 0.556), the extra number of cycles needed to measure the offset becomes Δ= ⌈log2(FS_{tech} /span)⌉, which particularized for a 1.8V 0.18µm CMOS process with FS_{tech} = 2V, only supposes 3 additional cycles.

The method can be easily extended for offset evaluation in comparators of Flash ADCs considering a multiplexer at the output of the bank of comparators which selects one of the thermometer bits in the output code as input for the *VIS*. Proceeding like this, the different offsets for all comparators can be sequentially evaluated (comparator by comparator). When the comparator under test is embedded in an ADC as in the Flash topology, the complete output code *k* with resolution *N* can be considered to accelerate the convergence speed.

IV. Validation Results

The validation of the proposed method was performed in multiple cases of study in different CMOS technologies. In this section, some simulation examples developed within the Cadence DesignFramework II environment are presented. In all cases, the *VIS* was described in a compact block with VerilogA and a fully differential comparator was considered.

A. Behavioral Simulations

The first example considers a behavioral model of the comparator with offset also in VerilogA. The aim of these simulations is to illustrate the accuracy and convergence of the method to the actual comparator threshold, even when its ideal value is not known a priori (worst-case conditions). Fig. 5 shows the behavioral simulations of transient convergence process for different comparator thresholds values {-505.7, 111.2, 477.3}mV with FS_{tech} = 2V and *T* = 10ns. The curves with markers, which correspond to the offset estimations, are superposed with actual offset values using the same color criteria to easily identify correspondence. The number of steps (clock cycles) in the *VIS* is *N* = 12, which provides according to (6) an accuracy in the threshold estimation of Δ ≈ 0.49mV. In all the cases, this theoretical limit was satisfactorily verified.

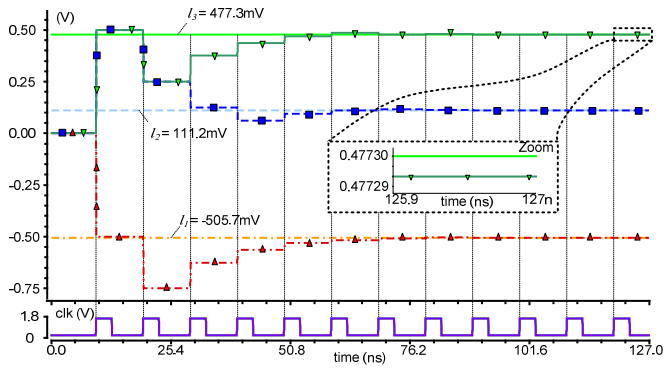

Fig. 5. Behavioral simulations of transient convergence process for different comparator threshold values (discrete-time comparator in VerilogA).

B. *Transistor-level Simulations*

This section presents results for the offset evaluation at full transistor level with mismatch and process variations (PVT). Two fully differential cases of study are considered in a 0.18µm CMOS process, labeled SCPC and SADL in Fig. 6a-b, respectively, which are comprised by: 1) a high-precision comparator based on a switched capacitor front-end with preamplifer, a dynamic latch followed by a RS latch and buffers, and 2) a relatively low-accuracy comparator without active front-end, the input of which is directly a stand-alone-alone dynamic latch (SADL), respectively. In the first stage, threshold voltages (l_p,l_n) are generated from a resistive ladder (omitted in the figure), while in the second topology, its generation is embedded in the circuit considering a size imbalance between input transistors $M_{R1\text{-}R2}$ [2] (in both examples, zero ideal thresholds, $l_{ideal} = 0$, were implemented).

Fig. 6c shows histogram distribution of the input referred offset estimations using Monte-Carlo analysis for the SCPC topology. Fig. 6d shows the equivalent results for the SADL comparator in the same conditions. As expected the input referred offset is much more sensitive to mismatch in this second case with a standard deviation std = 41mV_{rms} and maximum offset $|off|_{max}$=130mV, in contrast with the std=12mV_{rms} and $|off|_{max}$= 33mV for the SCPC. In each run, the offset is measured using a *VIS* with $N = 12$ and worst case span $FS_{tech} = 2V_{pp}$. The total CPU time (#runs: 300) was 17 minutes. For the same accuracy, the equivalent CPU time using the classical ramp method would be 91 days (17 days with a reduced span of 400mV).

CONCLUSIONS

This work presents a closed-loop simulation technique for evaluating the offset of discrete-time comparators during their design phase. The method, which uses a binary search algorithm, is based on the inclusion of the comparator under test in a SAR-loop that forces the input signal of the comparator to converge to its effective threshold with fast convergence (two orders of magnitude lower than existing approaches).

The search algorithm does not need any estimation of the threshold span, or knowing the ideal location of the threshold to drastically reduce the simulation time during comparator design phase. If the variability information of mismatch and PVT in the comparator threshold is known a priori, the number N_c of clock cycle to achieve convergence is reduced to $N_c \approx \log_2(span/accuracy)$. If this information is not available, only a few extra cycles are needed (typically, 3 or 4).

REFERENCES

[1] Jun He, Sanyi Zhan, Degang Chen, R.L. Geiger, "Analyses of Static and Dynamic Random Offset Voltages in Dynamic Comparators," *IEEE Tran. on Circuits and Systems I*, vol.56, no.5, pp.911,919, May 2009.

[2] A.J. Ginés, E. J. Peralias, A. Rueda, "Background Digital Calibration of Comparator Offsets in Pipeline ADCs", accepted for publication in *IEEE Transactions on Very Large Scale* Integration *Systems*.

[3] IEEE Standards, "IEEE standard for terminology and test methods for analog-to-digital converters," *IEEE Std* 1241-2000, 2001.

[4] J. J. Corcoran, T. Hornak & P. B. Skov: A high resolution error plotter for analog-to-digital converters, IEEE Transactions on Instrumentation and Measurement, 1975, vol. IM-24, no. 4, pp. 370-374

[5] A. Graupner, "A Methodology for the Offset-Simulation of Comparators". Available at: http://www.designers-guide.org.

[6] T.W. Matthews, P.L. Heedley, "A simulation method for accurately determining DC and dynamic offsets in comparators," *Midwest Sym. on Circuits and Systems,* pp.1815,1818 Vol. 2, 7-10 Aug. 2005.

[7] T. Rabuske et al., "An 8-bit 0.35-V 5.04-fJ/Conversion-Step SAR ADC With Background Self-Calibration of Comparator Offset," *IEEE Trans. on Very Large Scale Integration (VLSI) Sys.*, Early Access Online, 2014.

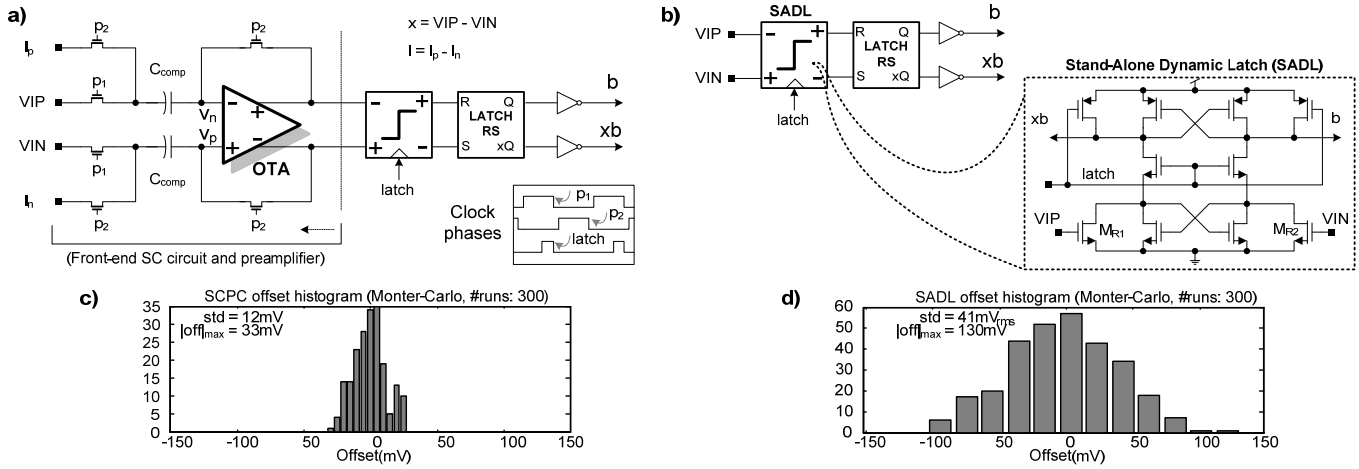

Fig. 6. Simplified schematics of: a) a SC front-end preamplified comparators (SCPC); b) stand-alone dynamic latch based comparator (SADL). Evaluation of the input referred offset based on Monte-carlo analysis at transistor level (#runs: 300) in a 0.18µm CMOS process: c) SCPC results; d) SADL results.