Statistical ADC Enhanced by Pipelining and Subranging

Sen Tao, *Student Member, IEEE*, Emmanuel Abbe, *Member, IEEE*, and Naveen Verma, *Member, IEEE*

***Abstract*—This brief explores the potential for enhancing both the achievable dynamic range and energy efficiency of statistical analog-to-digital converters (ADCs). To address dynamic range, the focus is on 1) the use of a strong kernel for statistical estimation (maximum-likelihood estimator) and 2) enabling the use of a large number of statistical observations. However, such a kernel, when used with a large number of observations, imposes high complexity and energy cost. To address energy, a pipelined front-end estimator is employed for coarse subrange estimation, and a back-end estimator is employed for fine statistical estimation. Architectural optimization of the back-end and front-end estimators is presented. For an ADC with nominal resolution of 10 bits, the approach achieves $<$ 1.3 LSB root-mean-square error, while reducing computations by $15\times$ compared to full statistical estimation.**

***Index Terms*—Analog-to-digital converter (ADC), comparators, estimation, statistical architecture.**

I. Introduction

STATISTICAL analog-to-digital converters (ADCs) refer to a class of ADCs that exploit stochastic behaviors at the circuit and device levels (electrical variations, geometric variations, etc.) as mechanisms for data conversion. With the increasing prominence of such behaviors, the appeal of statistical ADCs is that they can potentially circumvent the need for resource-intensive strategies to counteract these behaviors, possibly assuming an advantage over traditional ADCs [1].

A few examples of statistical ADCs have recently emerged, employing flash-type architectures [2]–[4]. These have been limited to precision less than 6 bits. Both Daly and Chandrakasan [2] and Flynn *et al.* [4] exploit stochastic device-level behaviors by employing a deterministic foreground calibration. On the other hand, Weaver *et al.* [3] exploit stochastic behaviors without calibration through comparators with thresholds designed to approximate a uniform probability density function (PDF) in order to simplify estimation. Analysis shows that increasing the number of comparators improves precision, but that precision is ultimately limited by the PDF approximation error of the comparator thresholds.

Thus far traditional ADCs have shown substantially higher efficiency and precision than statistical ADCs. One reason why it has been difficult to evaluate the full potential of statistical ADCs is that they have previously employed simple flash-type architectures. The aim of this work is to introduce architectural enhancements on top of this, which have traditionally yielded substantial benefits in ADCs, namely, pipelining and subranging. We show that these can affect the energy and precision of statistical ADCs in two ways. First, they enable the use of only a selected subset of the available statistical observations that is most informative for estimation. Second, by reducing the number of observations, they enable the use of strong estimation kernels as well as the opportunity to apply simplifying approximations. Kernels, such as maximum-likelihood estimators (MLE), can accommodate arbitrary PDFs, overcoming limitations due to PDF approximation. Generally, the PDFs can be "calibrated", but calibration is not applied to each instance of hardware, rather over aggregated hardware for representing the statistics. The calibration coefficients can then be preloaded in the hardware. Finally, this work presents an optimization, which guides how far pipelining and subranging should be taken within the architecture for a targeted precision.

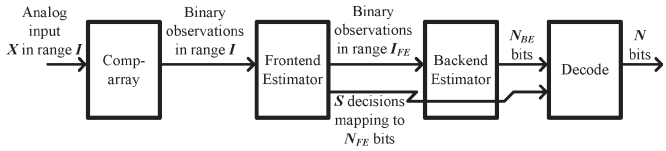

Fig. 1. Statistical ADC block diagram.

II. System Description

Fig. 1 shows the ADC block diagram. The analog input X is assumed to be uniformly distributed over a range I. The aim is to estimate the value x of this random variable X to a desired precision of N bits from binary statistical observations of x generated by the comparator-array block. This work targets a resolution of $N = 10$ bits. Generally, increasing precision involves increasing the number of informative observations and, thus, the complexity of estimation. To address this, the front-end estimator performs a subranging estimation to identify a reduced range $I_{\rm FE}$, in which x lies. The back-end estimator then performs statistical estimation over $I_{\rm FE}$ to finely estimate x. Thus, the front-end estimator effectively resolves the input to $N_{\rm FE}$ bits, with $N_{\rm FE} = \log_2(I/I_{\rm FE})$; however, as described below, this is done through S binary decisions, with $S > N_{\rm FE}$. The back-end estimator then resolves the remaining bits $N_{\rm BE} = N - N_{\rm FE}$, in the range $I_{\rm FE}$. Reduction of the range in this way has two implications for the back-end estimator. First, it reduces the number of observations by concentrating observations that are most informative.

Manuscript received November 18, 2014; accepted January 20, 2015. Date of publication February 24, 2015; date of current version May 29, 2015. This work was supported by Systems on Nanoscale Information fabriCs (SONIC), one of the six SRC STARnet Centers, sponsored by MARCO and DARPA. This brief was recommended by Associate Editor L. H. Corporales.

The authors are with the Department of Electrical Engineering, Princeton University, Princeton, NJ 08544 USA (e-mail: stao@princeton.edu).

Color versions of one or more of the figures in this brief are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSII.2015.2407231

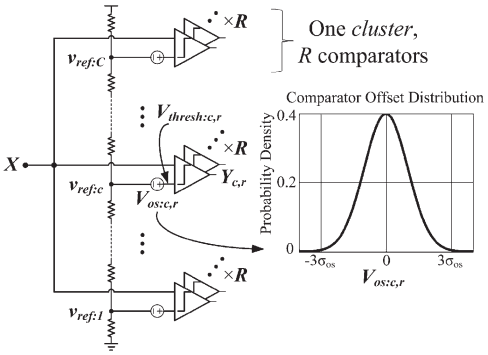

Fig. 2. Comparator-array and offset distribution.

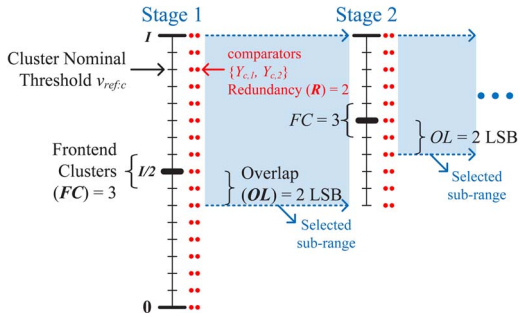

Fig. 3. Functional structure of front-end estimator.

Second, it enables a mathematical simplification made possible by treating all observations as being equally informative.

A. Comparator Array

Fig. 2 shows the structure of the comparator array. A group of comparators, referred to as a *cluster*, shares each reference voltage $v_{\mathrm{ref}:c}$. The number of comparators in each cluster, referred to as the *redundancy*, is designated by R. It is assumed that each comparator has an offset voltage $V_{\mathrm{os}:c,r}$. The random variables $V_{\mathrm{os}:c,r}$ are assumed to be independent and identically distributed (i.i.d.) with zero mean and standard deviation σ_{os}. The effective comparator thresholds are thus $V_{\mathrm{thresh}:c,r} = v_{\mathrm{ref}:c} + V_{\mathrm{os}:c,r}$, which are also random variables, having mean $v_{\mathrm{ref}:c}$ and standard deviation σ_{os}. For the designs considered, C reference voltages will be used, chosen to be uniformly spaced with separation less than σ_{os}. For example, assuming analog input range $I = 1$ V and $\sigma_{\mathrm{os}} = 20$ mV (which is typical for aggressively sized comparators in current technologies), $C = 2^{10} - 1$ reference voltages are considered.

Many factors contribute to comparator offset (transistor threshold variations, geometric variations, differential coupling, comparator kick-back noise, etc.). While some of these result in offset voltage with Gaussian distribution, this is not generally true for all sources. Nonetheless, a Gaussian distribution is assumed, but the precise distribution is not critical for the analysis in this work. We define the observation from each comparator to be a binary random variable $Y_{c,r}$. For a given x, $Y_{c,r}$ has a conditional probability mass function (PMF)

$$p_{Y_{c,r}|X}(y_{c,r}|x) = \begin{cases} \Pr\{x \geq V_{\mathrm{thresh}:c,r}\} = \Phi(x - v_{\mathrm{ref}:c}) & \text{if } y_{c,r} = 1, \\ \Pr\{x < V_{\mathrm{thresh}:c,r}\} = 1 - \Phi(x - v_{\mathrm{ref}:c}) & \text{if } y_{c,r} = 0. \end{cases} \quad (1)$$

B. Front-End Estimator

The front-end estimator is motivated by two considerations.

1. The complexity of the back-end estimator scales strongly with the number of observations it uses.
2. Some comparator observations $Y_{c,r}$ are more informative than others. If the PDF of the offset $V_{\mathrm{os}:c,r}$ falls substantially away from the mean, comparators with reference voltage $v_{\mathrm{ref}:c}$ close to the input x are most informative.

To reduce the number of observations required by the back-end estimator, the front-end estimator finds an informative subset of observations by employing subranging in multiple pipeline stages. Fig. 3 illustrates the structure by showing two such pipeline stages. A binary search produces a subrange, reducing ideally by half the range of consideration for estimating x at each stage. Since $Y_{c,r}$ are one-bit binary variables, pipelining is easily achieved by passing the observations through the stages using flip-flops.

For ideal binary search, the range is reduced by half at each stage. However, this implies the need to make high-precision decisions in the front-end estimator stages. In particular, analog inputs that fall near the midpoint of a subrange must be resolved to full N-bit precision as falling in the upper or lower half. This would require a large number of statistical observations at each stage. Instead, the approach and parameters shown in Fig. 3 are employed. The decision in each stage is made by majority voting over a number of front-end clusters FC, centered at the subrange midpoint. Therefore, only $R \times FC$ comparators are employed. This gives lower precision, such that the subrange cannot be reduced by half. Thus, an overlap region that extends from the midpoint into the upper and lower halves by an amount OL is included. This implies that each stage makes slightly less than a one-bit decision, and the number of stages needed is denoted by S. The rationale behind this approach is that the probability of making an error in the subrange decision goes down rapidly with even small OL, provided that the PDF of the comparator offset goes down rapidly for values away from its mean. Such a scenario holds for the assumed Gaussian PDFs and is also true for other PDFs that might be expected. However, with overlap region, the front-end estimator cannot reduce I_{FE} below $2 \times OL$. This leads to an efficiency optimization, wherein a large OL is preferred in the front-end estimator to reduce $R \times FC$ and/or the probability of error, but a small I_{FE} is preferred in the back-end estimator to reduce the range for statistical estimation.

Impact of design parameters on performance. The front-end estimator performance is affected by the design parameters, namely, R, FC, and OL (all assumed to be the same for all front-end stages). To evaluate performance, we consider the expected probability of making a decision error in a given stage, i.e., stg: $P_{\mathrm{ERR}:\mathrm{stg}}$ ($stg = 1, 2, \ldots, S$). This probability is defined as follows. Stage decisions are based on majority voting by $R \times FC$ comparators; however, deviated thresholds due to offsets can cause comparator-decision errors. From (1), the probability that a comparator with reference voltage $v_{\mathrm{ref}:c}$ gives an error when its output observation $Y_{c,r}$ is 1 (or 0) is $p_{Y_{c,r}|X}(1|x)$ (or $p_{Y_{c,r}|X}(0|x)$) for $x < v_{\mathrm{ref}:c} - OL$ (or $x > v_{\mathrm{ref}:c} + OL$). A stage error results when more than half of the $R \times FC$ comparators give such an error (note that, by our definition, when x is in the overlap region $[v_{\mathrm{ref}:c} - OL, v_{\mathrm{ref}:c} +$

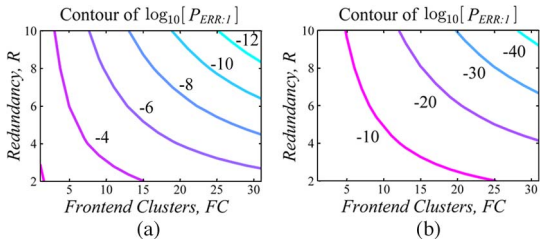

Fig. 4. Contours of $\log_{10}[P_{\mathrm{ERR:1}}]$ in terms of R, FC, and OL. (a) $OL = 10$ LSB. (b) $OL = 20$ LSB.

OL] no comparator error can occur.). As can be seen, the probability of stage error depends on x. Thus, $P_{\mathrm{ERR:stg}}$ corresponds to the expected (average) value of the stage error over the possible values of x. In particular, we assume x to be from a uniform distribution spanning the entire subrange under consideration by a stage in question; note that, because this subrange is different for each stage, $P_{\mathrm{ERR:stg}}$ is also slightly different.

For quantitative illustration, we consider the first stage, with $I = 1$ V and $C = 2^{10} - 1$ uniformly spaced comparator reference voltages. We use an offset standard deviation of $\sigma_{\mathrm{os}} = 20$ mV (=20 LSB, assuming a target resolution of $N = 10$ bits over $I = 1$ V). Fig. 4 shows contours of $\log_{10}[P_{\mathrm{ERR:1}}]$, with respect to R and FC, plotted for two different OL (10 and 20 LSB). As shown, $P_{\mathrm{ERR:1}}$ rapidly decreases with increasing OL. Similar values of $P_{\mathrm{ERR:stg}}$ are obtained for later stages.

C. Back-End Estimator

The back-end estimator employs comparators over BC clusters (i.e., back-end clusters) to arrive at an estimate $\hat{x}$ of the analog input, within the range I_{FE}. MLE is a widely used approach to statistical estimation, having the following objective function

$$\hat{x} = \arg\max_{x} \prod_{c,r} p_{Y_{c,r}|X}(y_{c,r}|x) \tag{2}$$

where c, r correspond to the $R \times BC$ comparators employed for estimation.

Two aspects pose challenges: (i) each of the terms in (2) must be known $p_{Y_{c,r}|X}(y_{c,r}|x)$, as defined in (1); (ii) a large number of informative observations $R \times BC$ should be used for low estimation error. With regards to (i), previous approaches have attempted to approximate $p_{Y_{c,r}|X}(y_{c,r}|x)$ by emulating a uniform PDF for the comparator threshold; this simplifies the objective function, but has limited precision to the 6-bit level [3]. With regards to (ii), the front-end estimator identifies the most informative subset of observations. However, for even a small subset, the number of terms in the objective function becomes intractable. The back-end estimator overcomes these challenges in two ways. First, reduction to the subset enables a mathematical simplification. Second, $p_{Y_{c,r}|X}(y_{c,r}|x)$ is not derived from a presumed uniform PDF for the comparator thresholds. Rather, the simplification enables arbitrary distributions to be approximated.

The simplification arises by using only comparator observations in a range I_{BE}, which spans BC clusters. I_{BE} is nearly the same size as I_{FE} (slightly larger) and is centered around I_{FE}. The benefit is that all comparator thresholds in this range can be treated as samples of a single random variable V_{thresh},

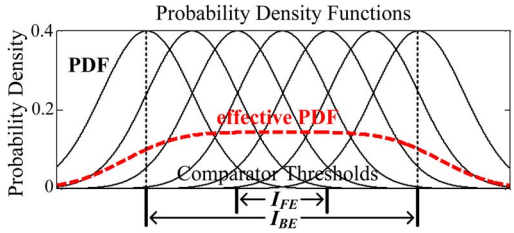

Fig. 5. Effective PDF employed for comparator thresholds.

which has the effective PDF $p_{V_{\mathrm{thresh}}}(v_{\mathrm{thresh}})$, as illustrated in Fig. 5. In particular, this PDF comes about both due to the equally spaced comparator reference voltages $v_{\mathrm{ref:}c}$ and the individual comparator offsets $V_{\mathrm{os:}c,r}$. To understand the impact of this simplification on statistical estimation, we derive $p_{V_{\mathrm{thresh}}}(v_{\mathrm{thresh}})$. To do so, we first treat the equally spaced comparator reference voltages $v_{\mathrm{ref:}c}$ as samples of a random variable V_{ref} with uniform distribution, thus having PDF

$$p_{V_{\mathrm{ref}}}(v_{\mathrm{ref}}) = \begin{cases} 1/I_{\mathrm{BE}} & v_{\mathrm{ref}} \in I_{\mathrm{BE}}, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

Next, we define a distribution for V_{thresh} by conditioning a nominally Gaussian offset-voltage distribution on the reference-voltage distribution earlier, as follows:

$$p_{V_{\mathrm{thresh}}|V_{\mathrm{ref}}}(v_{\mathrm{thresh}}|v_{\mathrm{ref}}) = \frac{1}{\sigma_{\mathrm{os}}\sqrt{2\pi}} e^{-\frac{(v_{\mathrm{thresh}} - v_{\mathrm{ref}})^2}{2\sigma_{\mathrm{os}}^2}} \tag{4}$$

From (3) and (4), the marginal distribution $p_{V_{\mathrm{thresh}}}(v_{\mathrm{thresh}})$ is

$$p_{V_{\mathrm{thresh}}}(v_{\mathrm{thresh}}) = \int_{-\infty}^{+\infty} p_{V_{\mathrm{thresh}}|V_{\mathrm{ref}}}(v_{\mathrm{thresh}}|v_{\mathrm{ref}}) p_{V_{\mathrm{ref}}}(v_{\mathrm{ref}})\, d\, v_{\mathrm{ref}} \tag{5}$$

With the back-end estimator treating all comparators as having this threshold distribution, we associate the comparator observations with a new single random variable Y_{BE}. For a given input x, we now derive the corresponding conditional PMF

$$\begin{aligned} &p_{Y_{\mathrm{BE}}|X}(y_{\mathrm{BE}} = 1|x) = \Pr\{x \geq V_{\mathrm{thresh}}\} \\ &= \int_{-\infty}^{x} p_{V_{\mathrm{thresh}}}(v_{\mathrm{thresh}}) d\, v_{\mathrm{thresh}} \\ &= \int_{-\infty}^{x} \left[\int_{-\infty}^{+\infty} p_{V_{\mathrm{thresh}}|V_{\mathrm{ref}}}(v_{\mathrm{thresh}}|v_{\mathrm{ref}}) p_{V_{\mathrm{ref}}}(v_{\mathrm{ref}})\, d\, v_{\mathrm{ref}} \right] d\, v_{\mathrm{thresh}} \\ &= \int_{-\infty}^{+\infty} p_{V_{\mathrm{ref}}}(v_{\mathrm{ref}}) \left[\int_{-\infty}^{x} p_{V_{\mathrm{thresh}}|V_{\mathrm{ref}}}(v_{\mathrm{thresh}}|v_{\mathrm{ref}}) d\, v_{\mathrm{thresh}} \right] d\, v_{\mathrm{ref}} \\ &= \int_{-\infty}^{+\infty} p_{V_{\mathrm{ref}}}(v_{\mathrm{ref}}) \Phi(x - v_{\mathrm{ref}}) d\, v_{\mathrm{ref}} \\ &= E_{V_{\mathrm{ref}}}[\Phi(x - v_{\mathrm{ref}})] \end{aligned} \tag{6}$$

Next, since Y_{BE} is a binary random variable, values of its conditional PMF can be denoted by an effective cumulative distribution function $\mathrm{CDF}_{\mathrm{eff}}$

$$p_{Y_{\mathrm{BE}}|X}(y_{\mathrm{BE}}|x) = \begin{cases} \Pr\{x \geq V_{\mathrm{thresh}}\} = \mathrm{CDF}_{\mathrm{eff}}(x) & \text{if } y_{\mathrm{BE}} = 1, \\ \Pr\{x < V_{\mathrm{thresh}}\} = 1 - \mathrm{CDF}_{\mathrm{eff}}(x) & \text{if } y_{\mathrm{BE}} = 0. \end{cases}$$

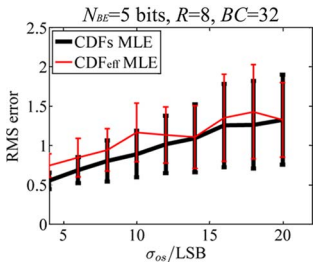

Fig. 6. Performance comparison between $\mathrm{CDF}_{\mathrm{eff}}$ and CDFs.

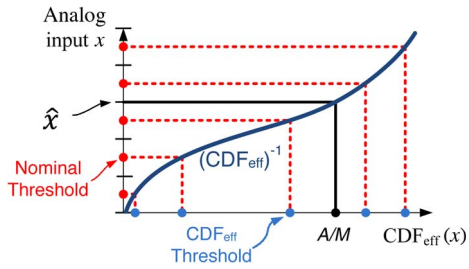

Fig. 7. Lookup table search approach.

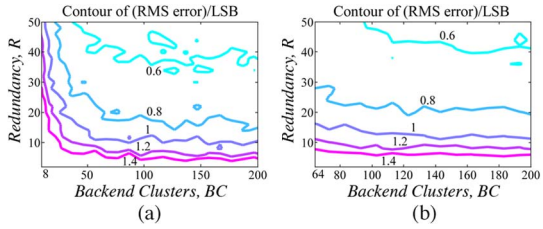

Fig. 8. Contours of (RMS error)/LSB in terms of R, BC, and I_{FE}. (a) $I_{FE} = 8$ LSB. (b) $I_{FE} = 64$ LSB.

With all comparator observations $y_{\mathrm{BE},i}$ now treated as samples of the same random variable Y_{BE}, the MLE objective function is simplified as follows:

$$\hat{x} = \arg\max_{x} \prod_{i} [\mathrm{CDF}_{\mathrm{eff}}(x)]^{y_{\mathrm{BE},i}} [1 - \mathrm{CDF}_{\mathrm{eff}}(x)]^{1-y_{\mathrm{BE},i}}. \quad (7)$$

We can see from (6) that what is lost in this simplification is that all observations within the range I_{BE} are treated as being derived from the *same averaged* $\mathrm{CDF}_{\mathrm{eff}}$, i.e., $E_{V_{\mathrm{ref}}}[\Phi(x - v_{\mathrm{ref}})]$. This has two implications. First, all observations now scale the objective function equally, when, in reality, observations with different $v_{\mathrm{ref}:c}$ should scale the objective function differently (i.e., those with $v_{\mathrm{ref}:c}$ closer to x should have heavier weight through their individual CDFs, since they are somewhat more informative, while those with $v_{\mathrm{ref}:c}$ farther from x should have lower weight). Second, $\mathrm{CDF}_{\mathrm{eff}}$ is derived by averaging the individual CDFs of the comparators in the range I_{BE}. Hence, we see that the diversity of the individual CDFs *is* considered through this averaging, but this diversity is *not* associated to the specific observations. Reduction to I_{FE} of the range under consideration is critical for this simplification, since this diversity is reduced within this range. To show the validity of the simplification, Fig. 6 considers a range of $I_{\mathrm{FE}} = 32$ LSB (corresponding to $N_{\mathrm{BE}} = 5$ bits), showing the root mean square (RMS) of the error from estimation (normalized to a LSB at the 10-bit level, over $I = 1$ V) for a full MLE and for the simplified MLE (in (7)). As shown, at the comparator-offset standard deviation of interest ($\sigma_{\mathrm{os}} = 20$ mV $= 20$ LSB), the two estimates converge.

MLE. Use *log-likelihood* function to solve (7). Then, MLE can be achieved easily by setting the derivative of the log-likelihood function to zero, to arrive at the desired $\hat{x}$, as follows:

$$\frac{d[\log(\prod_i [\mathrm{CDF}_{\mathrm{eff}}(x)]^{y_{\mathrm{BE},i}}[1 - \mathrm{CDF}_{\mathrm{eff}}(x)]^{1-y_{\mathrm{BE},i}})]}{d\,x} = 0$$

$$\mathrm{CDF}_{\mathrm{eff}}(x) = \frac{\sum_i y_{\mathrm{BE},i}}{\sum_i 1} = \frac{A}{M}. \quad (8)$$

To perform statistical estimation from *A/M*, we propose an implementation based on a lookup table that stores $(\mathrm{CDF}_{\mathrm{eff}})^{-1}$. Since only N_{BE} bits must be estimated, the lookup table must store only $2^{N_{\mathrm{BE}}}$ entries. Furthermore, since $\mathrm{CDF}_{\mathrm{eff}}$ (or $(\mathrm{CDF}_{\mathrm{eff}})^{-1}$) is a monotonically increasing function with respect to x (or *A/M*), fast lookup table search approaches can be employed (such as binary search) to arrive at a value for $\hat{x}$, where $\mathrm{CDF}_{\mathrm{eff}}$ is closest to *A/M*. Fig. 7 illustrates the lookup table approach. We note that, while the derivation arriving at $\mathrm{CDF}_{\mathrm{eff}}$ (in (6)) assumes a Gaussian distribution for comparator offset, the actual $\mathrm{CDF}_{\mathrm{eff}}$ can be derived for arbitrary distributions. In fact, in practice, $\mathrm{CDF}_{\mathrm{eff}}$ can be derived from hardware measurement and then preloaded in the lookup table. In general, comparator offset will depend on dynamic conditions (e.g., supply, temperature variations). While differential comparator circuits provide some first-order cancelation, $\mathrm{CDF}_{\mathrm{eff}}$ can be characterized across these conditions to accommodate second-order effects. Although this will result in better estimation performance *across* the conditions, the standard deviation of comparator offset is likely to increase, leading to worse estimation performance compared to a *specific* set of conditions. Alternate approaches wherein different $\mathrm{CDF}_{\mathrm{eff}}$ values are characterized for specific conditions and then selected (based on sensors) can also be considered.

Impact of design parameters on performance. As in the front-end estimator, the precision achievable by the back-end estimator depends on the number of observations employed. However, unlike the front-end estimator, which makes *multiple (less than) single-bit decisions (which map to N_{FE})*, the back-end estimator makes a *single multiple-bit decision (N_{BE})*.

To analyze performance, we use RMS error. The design parameters considered are R, BC, and I_{FE}. Once again, for quantitative illustration, we consider the parameters in Section II-B ($\sigma_{\mathrm{os}} = 20$ mV, $C = 2^{10} - 1$, $I = 1$ V). Fig. 8 shows contours of the (RMS error)/LSB with respect to R and BC, for two values of I_{FE} (8 and 64 LSB, corresponding to $N_{\mathrm{BE}} = 3$ bits and $N_{\mathrm{BE}} = 6$ bits, respectively) by using $\mathrm{CDF}_{\mathrm{eff}}$. As shown, a larger I_{FE} requires higher BC (since $I_{\mathrm{BE}} \geq I_{\mathrm{FE}}$), but the error saturates as BC increases. This occurs because the corresponding comparator reference voltages are far from the target range I_{FE}, making them less informative.

III. System Optimization

This section describes an optimization for the ADC. The front-end and back-end estimators perform different but interdependent functions. Namely, decisions made by the front-end estimator give a precision of I_{FE}, which is limited to a minimum value of $2 \times OL$. I_{FE} then affects the complexity and performance of the back-end estimator (through the number

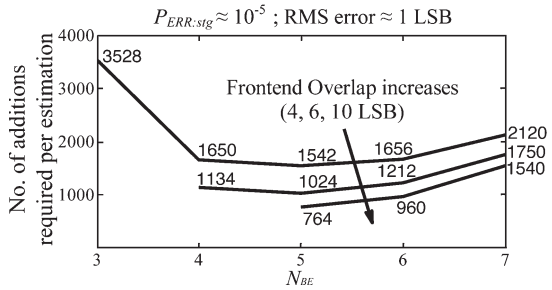

Fig. 9. Total number of additions required per estimation.

of observations that it must employ and the validity of the mathematical simplification that it applies).

The basic computations in both the front-end and back-end estimators are simply additions over the binary observations; in the front-end estimator, additions are used to determine the ratio of 1s over the FC clusters for majority voting, while in the back-end estimator, additions are used to determine the ratio of 1s over BC clusters with respect to $\mathrm{CDF}_{\mathrm{eff}}$. The optimization considered thus seeks to find the fewest number of total additions required.

In addition to being limited by finite precision, as in the case of a traditional ADC, the statistical ADC is also limited by error probability, arising in the front-end subrange decisions. The error probability over S stages $P_{\mathrm{ERR,FE}}$ depends on the error probability of each stage $P_{\mathrm{ERR:stg}}$, i.e., $P_{\mathrm{ERR,FE}} = [1-(1-P_{\mathrm{ERR:1}})(1-P_{\mathrm{ERR:2}})\cdots(1-P_{\mathrm{ERR:}S})]$. Although slightly different mathematically, the stagewise error probabilities have very similar values in practice, giving the approximation $P_{\mathrm{ERR,FE}} \approx 1-(1-P_{\mathrm{ERR:1}})^S$. The optimization then proceeds by defining an acceptable error probability for the front-end estimator and a RMS error for the back-end estimator. For quantitative analysis, we specify an error probability of $P_{\mathrm{ERR,FE}} < 10^{-4}$ and a RMS error of ~ 1 LSB, respectively. We now aim to find the optimal $N_{\mathrm{FE}}[=\log_2(I/I_{\mathrm{FE}})]$ that results in the fewest front-end and back-end estimator additions. For the front-end estimator, the total additions are given by $R \times FC \times S$. The minimum $R \times FC$ (for various OL) can be determined from the analysis in Fig. 4, when given the stagewise error probability $P_{\mathrm{ERR:1}}$. From $P_{\mathrm{ERR,FE}} < 10^{-4}$, $P_{\mathrm{ERR:1}} < 10^{-5}$ is targeted, valid for S between 1 and 10. For the back-end estimator, the total additions are given by $R \times BC$, for which the minimum value (for various I_{FE}) can be determined from the analysis in Fig. 8, when given the RMS error. We can thus vary N_{BE} (for different OL), giving us the front-end stages S and, thus, the total number of front-end additions, as well as the total number of back-end additions. Doing this, Fig. 9 shows that an optimal point emerges near $N_{\mathrm{BE}} = 5$ bits ($I_{\mathrm{FE}} = 32$ LSB). Below this point, additional front-end stages are required; above this, additional back-end clusters are required. Furthermore, as shown, increasing OL reduces the total additions by enabling lower $R \times FC$ in the front-end stages. However, increased OL also limits how small I_{FE} can be made, thus requiring a minimum resolution N_{BE} in the back-end estimator (e.g., $N_{\mathrm{BE}} \geq 5$ bits for $OL = 10$ LSB).

IV. Results

To characterize the performance of the full ADC (front-end and back-end), we apply a uniformly distributed analog input and find the overall RMS error of the resulting digital code. Note that this is larger than the RMS error specified for just the back-end estimator in Section III, due to the impact of front-end error probability.

TABLE I
Overall Performance of the Whole System
($N_{\mathrm{BE}} = 5$ bits, $OL = 14$ LSB, $S = 8$)

R	FC	BC	(RMS error)/LSB	$Adds$	Computational Savings
8	1	37	1.7418	360	23×
		41	1.7472	392	21×
		45	1.7567	424	19×
8	3	37	1.6039	488	17×
		41	1.5190	520	16×
		45	1.4541	552	15×
10	1	37	1.5236	450	23×
		41	1.4240	490	21×
		45	1.4314	530	19×
10	**3**	37	1.4076	610	17×
		41	1.3750	650	16×
		45	**1.2964**	**690**	**15×**
12	1	37	1.3294	540	23×
		41	1.2980	588	21×
		45	1.2774	636	19×
12	3	37	1.3016	732	17×
		41	1.2610	780	16×
		45	1.1964	828	15×

Optimization is performed, as described in Section III ($N_{\mathrm{BE}} = 5$ bits, $I_{\mathrm{FE}} = 32$ LSB), with the parameters in Section II-B ($N = 10$ bits, $\sigma_{\mathrm{os}} = 20$ mV, $C = 2^{10} - 1$, $I = 1$ V). With large OL preferred (see Fig. 9), $OL = 14$ LSB is chosen, near the upper bound for the $N_{\mathrm{BE}} = 5$-bit optimal point. At these values of OL and N_{BE}, $S = 8$.

While, generally, the optimal R might be different for the front-end and back-end estimators, the same R is preferred for simplifying the implementation. The optimization in Section III leads to values for $R \times FC$ and $R \times BC$, but the precise values of R, FC, and BC are not yet constrained. Thus, Table I shows the simulated overall ADC performance, over these parameters, averaged over 50 Monte Carlo runs, wherein all comparator offsets are randomly sampled from a Gaussian distribution. As an example, assuming an overall RMS error target of < 1.3 LSB, the additions needed are as low as 690 (bold line in Table I). In contrast, for full MLE, all comparator observations ($R \times C = 10\,230$) are required, corresponding to $15\times$ reduction due to the pipelining and subranging.

Acknowledgment

The authors thank Prof. A. Singer and R. Corey (UIUC) for their inputs.

References

[1] J. L. Ceballos, I. Galton, and G. C. Temes, "Stochastic analog-to-digital conversion," *Proc. 48th Midwest Symp. Circuits Syst.*, Aug. 2005, vol. 1, pp. 855–858.

[2] D. C. Daly and A. P. Chandrakasan, "A 6-bit, 0.2 V to 0.9 V highly digital flash ADC with comparator redundancy," *IEEE J. Solid-State Circuits*, vol. 44, no. 11, pp. 3030–3038, Nov. 2009.

[3] S. Weaver, B. Hershberg, P. Kurahashi, D. Knierim, and U.-K. Moon, "Stochastic flash analog-to-digital conversion," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 57, no. 11, pp. 2825–2833, Nov. 2010.

[4] M. P. Flynn, C. Donovan, and L. Sattler, "Digital calibration incorporating redundancy of flash ADCs," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 50, no. 5, pp. 205–213, May 2003.