DAC nonlinearity and residue gain error correction in a pipelined ADC using a split-ADC architecture

Imran Ahmed
Department of Electrical and Computer Engineering
University of Toronto
Toronto, Canada
imran@eecg.toronto.edu

David A. Johns
Department of Electrical and Computer Engineering
University of Toronto
Toronto, Canada
johns@eecg.toronto.edu

***Abstract*—A split-ADC architecture is used to calibrate both the non-linearity errors introduced by capacitor mismatches in the DAC, and gain errors in the residue amplifier of the first stage in a pipelined ADC. The background scheme only requires 10^5 clock cycles to perform the calibration to more than 12b accuracy. Simulated in Simulink and Spice, the digital calibration scheme improves the ADC's SNDR/SFDR from 54dB/58dB before calibration to 78dB/85dB after calibration.**

I. Introduction

The thrust of pipelined ADC research in recent years has been focused on calibration schemes which compensate for technology limitations that limit ADC accuracy. Low intrinsic gain in CMOS transistors cause deviations in residue amplifier gain, and insufficient matching between capacitors result in both deviations in residue amplifier gain and DAC nonlinearity in a pipelined ADC.

To minimize the distortion of a thermometer coded DAC in a pipeline stage, DAC capacitors are typically made larger than thermal noise requirements, resulting in increased power consumption. Additional layers can also be used to implement well matched MiM capacitors, although extra layers increase fabrication costs. Prior publications of ADCs which digitally correct DAC nonlinearity are categorized into foreground or background schemes. A background scheme is preferred as the operation of the ADC is not required to be interrupted to perform calibration and no special calibration signals are required to be generated. In [1] a DAC Noise Cancellation (DNC) scheme based dynamic element matching is used to cancel errors of the DAC in the background. Although effective the DNC scheme requires many clock cycles (2^{25} for 14b precision) to perform the calibration as it is statistical in nature. Other schemes [2] include swapping each DAC capacitor with an extra capacitor where capacitor mismatch is measured by modulating the switching scheme. Such schemes however also require long calibration times when operating in the background [2].

In [3], [4] it is shown that a fast background calibration of residue amplifier gain error can be achieved using a 'split-ADC' architecture. In this work we show how the split-ADC architecture can be expanded to also digitally calibrate a pipeline ADC's DAC errors in the background over a short time interval. The architecture is implemented in both Simulink and Spice where both models show the ADC to achieve an SNDR beyond 12b after calibration, contrasted with 8-9b before calibration. As a result the high accuracy ADC of this work can be more cost effective to implement than prior works because: high matching MiM capacitors are not required, stringent layout requirements for high matching can be relaxed (thus saving design time and costs), and a short calibration time can be achieved thus minimizing the time required to test a product before deployment.

II. Split-ADC based DAC calibration scheme

A. *Review of split-ADC architecture*

The split-ADC architecture is based on two ADCs processing the same sampled analog input in parallel, where each ADC has the same resolution, but a different residue transfer characteristic [4] as shown in Fig. 1. Note in [4] the split-ADC was used in a cyclic ADC. The technique naturally lends itself to application in a pipelined ADC as shown in Fig. 1. If no residue amplifier errors are present both ADCs output the same output code. If errors exist in the residue amplifier however (e.g. due to finite opamp gain), both ADCs produce a different output due to each ADC having a different residue transfer characteristic. Thus the difference between the two ADC outputs can be used to adapt a corrective term to compensate for residue amplifier gain errors in the digital domain as shown in Fig. 1.

Since the final ADC output is generated by the sum of the two ADC outputs, each of the split ADCs can be designed with half the capacitance to meet thermal noise requirements, thus each of the two split ADCs have approximately half the power and area required in a single ADC designed for the same resolution.

1-4244-0157-7/06/$20.00 ©2006 IEEE

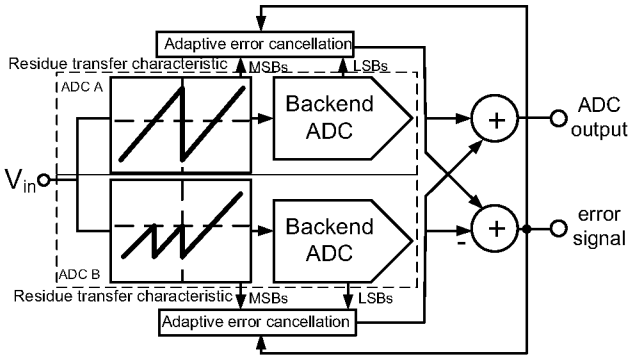

Fig. 1: Split-ADC generalized to a pipeline ADC

In a split-ADC approach each sampled analog input produces an error which can be immediately used to drive the adaptive correction, and therefore achieve fast calibration. This is contrasted with other statistical approaches (e.g.) [5] which require thousands of ADC outputs to be correlated before a valid sample of the error signal can be used for adaptation. Split-ADC calibration is a background approach as most practical inputs to the ADC produce an error signal which can be used to drive the corrective adaptation.

B. Architecture of this work

Each split-ADC in this work consists of a 4b pipeline stage followed by 10b backend ADC. Calibration is only performed on the first stage of each split-ADC, as prior publications show a 10b ADC can be implemented without calibration. For simplicity in the simulation model we model the backend as a single 10b flash ADC, thus effectively the model is of a two-stage ADC, however the abstraction is also applicable to a pipelined ADC with more than two stages where only the first stage is calibrated. Note that we aim for a 12b ADC - two extra bits of resolution are included in the pipeline to improve the accuracy of the correction to less than 1LSB at the 12b level.

The key to the split-ADC is to have a different residue transfer characteristic in each of the two split-ADCs. In this work different residue transfer characteristics are achieved by offsetting one curve with respect to the other [5] as shown in Fig. 2. Fig. 3 shows the output and error signal of each split ADC (only 1b of MSBs are shown to simplify the illustration; the MSB is 4b in this work) under ideal conditions, i.e. no gain or capacitor mismatch errors.

Errors in the residue amplifiers cause each backend code to have a different slope. As shown in Fig. 4 the dissimilar residue transfer characteristics of channel A and B lead to the difference between channel A and B to be nonzero, hence the difference can be used to adapt a corrective term for gain errors in the residue amplifier [4].

Random mismatch between capacitors in the DAC of the first pipeline stage however result in a residue transfer characteristic that has random static offset as a function of the MSB code as shown in Fig. 5.

Note however that if no residue amplifier gain *and* no DAC errors are present the difference between channel A and B is zero. Thus the difference between channel A and B can *also* be used to calibrate the DAC nonlinearity in addition to the residue gain error.

C. Residue amplifier gain correction

Errors in the residue gain amplifier can be corrected by scaling the effective radix of the ADC. Alternatively each portion of the residue transfer characteristic associated with each MSB code can be shifted by a constant that uniformly changes with the MSB as shown in Fig. 6.

D. DAC nonlinearity correction

In this work, to correct DAC errors, the corrective shift described in section C is made a function of the MSB code, where an array of offsets indexed by the MSB in the digital domain is LMS adapted to shift portions of the residue transfer characteristic by an amount related to both the residue gain error and the mismatch in the DAC to drive the error signal to zero. The architecture of this work is shown in Fig. 7 (only ADC A is shown; ADC B is identical).

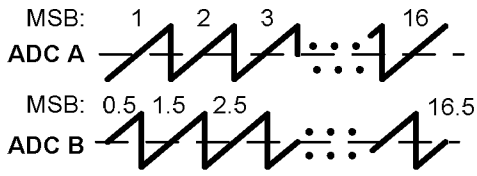

Fig. 2: Residue transfer characteristic of this work

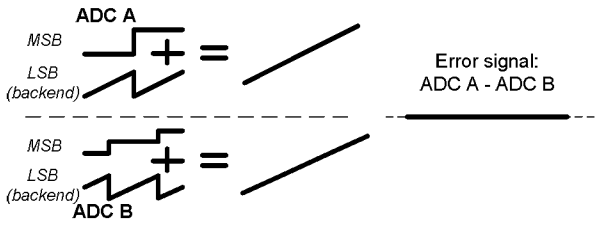

Fig. 3: Error signal under ideal conditions

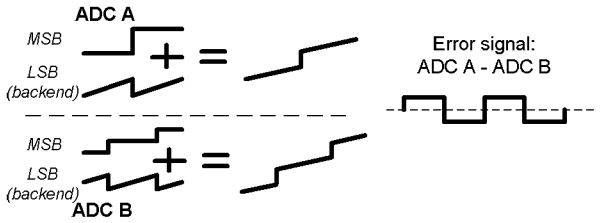

Fig. 4: Error signal with gain errors in ADC A, B

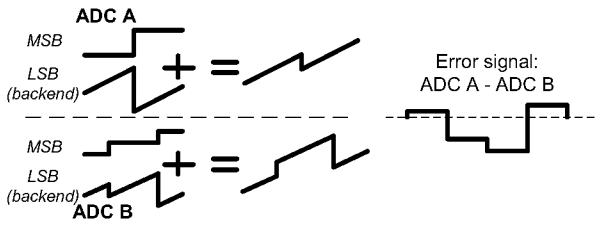

Fig. 5: Error signal w/DAC nonlinearity in ADC A, B

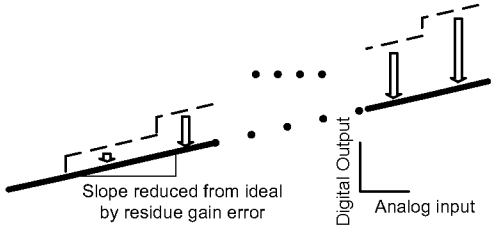

Fig. 6: Residue gain correction with successive offsets

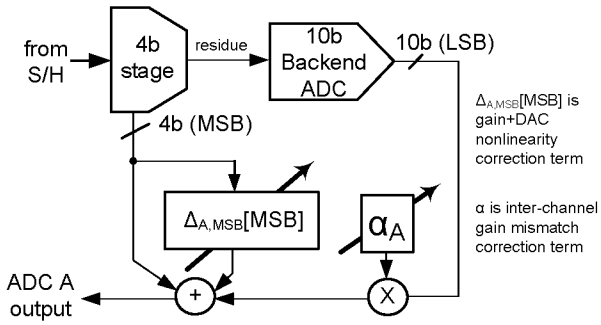

Fig. 7: Calibration architecture

E. Accounting for interchannel mismatch

Due to random mismatches, ADC A of Fig. 1 will have a slightly different overall gain and offset when compared to ADC B. Inter-channel offsets are already accounted for by the adaptive offset terms. To account for gain mismatch between the two ADCs an LMS adaptive gain term (α in Fig. 7) is also included which scales each backend code, which when combined with the adaptive offset term can be used to effectively scale the overall gain of each ADC, such that inter-channel gain mismatch errors are minimized.

F. Calibration algorithm

Consider the ADC input/output transfer characteristic of channels A and B as shown in Fig. 8 (1b MSB example only shown to simplify illustration). The transfer characteristics are separated into different regions, which correspond to different combinations of the MSB codes of each ADC. If for example three consecutive ADC inputs lie in regions 1, 2, then 1 respectively, the adaptation will not converge as driving the error to zero in a particular region inadvertently introduces an offset in adjacent regions as shown in Fig. 8 undoing the correction of those regions.

To avoid undoing calibration in adjacent regions, adjacent regions to one side are also shifted by the same amount as the region under calibration to preserve the code separation between ADCs A and B. I.e. when in region 1 the adaptive offset for ADC A is adjusted to force the output of ADC A to equal ADC B. To preserve the error of adjacent regions, the adaptive offsets of ADC A and B in regions 2, 3, and 4 are also shifted by the same amount.

When in region 2, to ensure that the error in region 1 is unaffected, the adaptive offset for ADC A is not modified, rather the error is forced to zero by adapting the offset term of ADC B in region 2 such that the outputs of the two channels are equal. To ensure that the separation of output codes between ADC A and B are preserved, the same offset is applied to the adaptive terms in ADC A and B for regions 3, and 4 as described in Table I. By induction one can show that a similar approach can be used for regions 3 and 4.

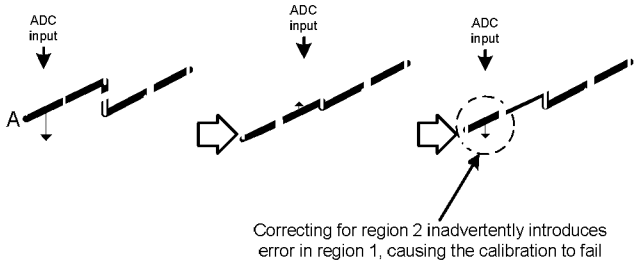

Fig. 8: Effect random input on calibration

TABLE I: Offset update rules (1b example, μ a constant)
(Δ subscripts are based on MSB labeling from Fig. 2)

Region	Update for ADC A	Update for ADC B
1	$\Delta_{A1} = \Delta_{A1} - \mu e$ $\Delta_{A2} = \Delta_{A2} - \mu e$	$\Delta_{B1.5} = \Delta_{B1.5} - \mu e$ $\Delta_{B2.5} = \Delta_{B2.5} - \mu e$
2	$\Delta_{A2} = \Delta_{A2} + \mu e$	$\Delta_{B1.5} = \Delta_{B1.5} + \mu e$ $\Delta_{B2.5} = \Delta_{B2.5} + \mu e$
3	$\Delta_{A2} = \Delta_{A2} - \mu e$	$\Delta_{B2.5} = \Delta_{B2.5} - \mu e$
4	no update	$\Delta_{B2.5} = \Delta_{B2.5} + \mu e$

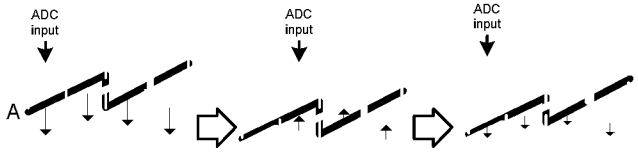

Fig. 9: Modified calibration with random input

III. SIMULATION RESULTS

A. Simulink verification

A model of the architecture in Fig. 7 was implemented in Simulink for the split-ADC configuration. The model was simulated with a 3% mismatch in the relative overall gain between ADC A and B. Nonlinearity in the DACs were modeled by adding random static offsets as a function of the DAC inputs, to the DAC outputs. Residue gain errors were modeled by scaling the ideal residue amplifier gain by a factor less than 1. The adaptive offset and gain terms of Fig. 7 were preset to the values required for an ideal ADC, thus the calibration time is measured as the time required for the ADC to adapt from the state where no corrections are required. To verify the background nature of the calibration, a full scale uniform random input was applied to the ADC for 10^5 clock cycles (the system was also verified to converge with other inputs such as sine waves, ramps, etc.). Thereafter the adaptation was frozen and a sinusoid

subsequently applied to the ADC input for 2^{14} clock cycles, and a 16,384 point FFT taken of the digitized sinusoid.

Fig. 10 shows the evolution of the error signal with the random input to the ADC. When simulating with a capacitor matching of only ~8 bits and an error in the residue amplifier on the order of 10% from the ideal (where each ADC had slightly different residue gain and capacitor mismatch), the SNDR/SFDR of the uncalibrated ADC was 50/61dB. After calibration the SNDR/SFDR of the ADC was 83/96dB. The calibration time of 10^5 (~2^{17}) clock cycles is much lower than the 2^{2N} (N=calibration accuracy) clock cycles [4] typically required to calibrate using statistical methods, and the 2^{25} clock cycles required in [1] to calibrate DAC errors.

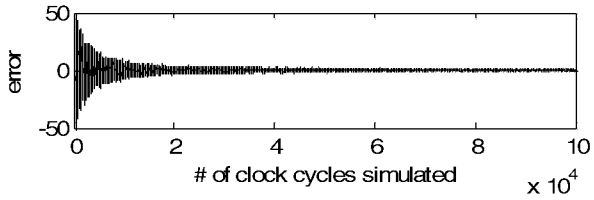

Fig. 10: evolution of error signal over time

B. Verification in Spice

As circuit non-idealities such as switch distortion, opamp nonlinearity, charge injection etc. are difficult to accurately model in Simulink, a model of the front end stage was implemented in Spice. The front end pipeline stage was implemented using circuit blocks from a prior fabricated pipelined ADC by the authors [6], and modified from a 1.5b to a 4b pipeline stage. To simplify the model, ideal comparators and an ideal buffer in the sample and hold were used in the Spice simulation as shown in Fig. 11.

The opamp used in the MDAC of the Spice model had a nominal DC gain of only ~50dB, noting that without calibration a DC gain of >80dB would be required to realize better than 10b settling accuracy and a closed loop gain of 16. The Spice model was simulated with a sinusoidal input for 256 clock cycles where the 256 output samples formed one period of a sinusoid. The residue amplifier output and MSB outputs of each ADC were streamed out to a text file and imported in Simulink where the same model of the backend ADC and digital calibration as used in section III.A operated on the results of the Spice simulation. To allow more calibration cycles than were simulated for in Spice, the Spice outputs were looped several times to emulate the sampling of a periodic input. To include random mismatch the circuit netlist was flattened and random mismatches applied to each transistor according to device and foundry information. To highlight the corrective ability of this work, capacitors were matched to only ~9b. Simulations with the front end modeled in Spice show the ADC without calibration to have an SNDR/SFDR of 53.6/58dB, and with calibration to have an SNDR/SFDR of 77.9/84.5dB. The linearity of the Spice/Simulink simulation is limited by the linearity of the CMOS switches used in the circuit. The calibration convergence time was ~10^5 clock cycles. FFTs of un-calibrated and calibrated outputs are shown in Fig. 12

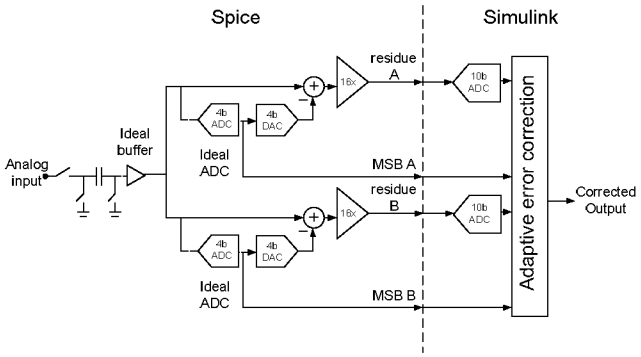

Fig. 11: Spice-Simulink hybrid model

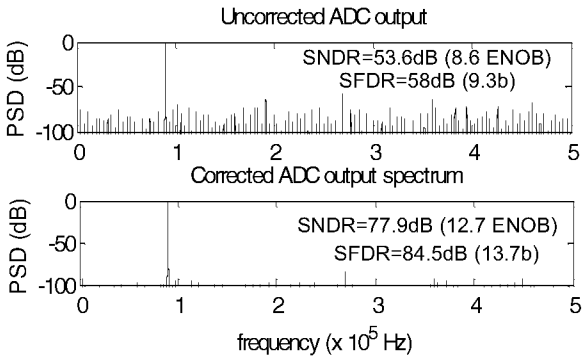

Fig. 12: FFT before and after calibration of Spice/Simulink hybrid model

As the SNDR is beyond 12b after calibration, thermal noise (not included in simulations) will dominate the accuracy of the ADC (assuming capacitors sized for 12b output accuracy, and sufficient linearity in each pipeline stage), which from a power perspective is optimal for a Nyquist rate ADC.

IV. Conclusions

An architecture which calibrates both DAC and residue amplifier errors in the background over short time intervals was described. Simulations in Simulink and Spice verify the ADC architecture to achieve an SNDR beyond the 12b level (78dB), with linearity on the order 14b after calibration.

References

[1] I. Galton, "Digital Cancellation of D/A Converter Noise in Pipelined A/D Converters," IEEE TCAS-II. vol. 47, pp. 185–196, March 2000.

[2] M. Kinyua, F.Maloberti, W. Gosney, "Digital Background Auto-Calibration of DAC Non-Lineairy in Pipelined ADCs", ISCAS 2004. pp.I-13–16.

[3] J. Li, UK Moon, "Background Calibration Techniques for Multistage Pipelined ADCs With Digital Redundancy," IEEE TCAS-II, vol. 50, pp. 531-538, September 2003

[4] J. McNeill, J. Coln, B. Larivee, "A split-ADC architecture for deterministic digital background calibration of a 16b 1MS/s ADC," ISSCC 2005, vol. I, Feb. 2005 pp. 276-598

[5] B. Murmann, B.E. Boser, "A 12b 75MS/s Pipelined ADC using Open-Loop Residue Amplification", ISSCC 2003, vol 1, pp.328-497

[6] I. Ahmed, D. Johns, "A 50MS/s (35mW) to 1kS/s (15μW) power scaleable 10b pipelined ADC with minimal bias current variation", ISSCC 2005, vol I, Feb 2005, pp. 280-598