

Digital assistance for energy reduction in ADCs using a signal prediction algorithm

D. Avila, E. Alvarez and A. Abusleme

With the adoption of new technologies for analogue circuits, different digital techniques have been designed to enhance their performance. Among the existing techniques, a promising approach is to adapt the circuit operation dynamically considering the application characteristics. In the field of analogue-to-digital converters (ADCs), typically this approach is carried out by taking advantage of application-dependent signal properties, hence their use is limited. In this work, a digital assistance technique for power reduction in ADCs is presented. By defining a range for the next sample based upon the maximum possible change of the input signal between samples, the proposed algorithm reduces the mean energy consumption per conversion in a variety of ADC architectures, regardless of the application.

Introduction: Digital assistance is used to expand the performance envelope of analogue circuits by taking advantage of the fast growing performance of digital circuits resulting from the technology scaling [1]. Among the existing digital assistance techniques, a promising approach is to adapt the circuit operation dynamically according to the characteristics of its input signal, for example, in order to reduce the overall energy consumption. Several applications using this approach can be found in the design of ADCs for high-efficiency, low-power applications [2, 3, 4, 5]. Typically, this approach is carried out by using application-dependent signal characteristics, such as periodical changes in the input signal rate or the input signal expected shape. For this reason, these techniques lack generality and can only be used in specific applications. In this work, a digital assistance technique for energy reduction in general-purpose ADCs is presented.

When designing an ADC for a custom IC, the maximum rate of change of the ADC input signal can be determined from the frequency bandwidth of the stage preceding to the ADC, typically a signal-conditioning stage. Using this parameter and the ADC sampling frequency, an expression for the maximum variation between samples of the ADC output can be computed. If the maximum possible change of the input signal between samples is limited, some of the most significant bits (MSBs) of the ADC output will remain constant between two consecutive samples, and for the next conversion, a part of the resulting bits are known already. Skipping the MSBs that remain constant, the ADC only needs to compute the bits that are subject to change, thus reducing the mean energy consumption per conversion. For instance, in flash ADCs, for each MSB that remains constant this technique allows to turn off half of the comparators, whereas in bit-at-a-time ADCs, this technique allows a higher sampling rate when operating asynchronously, resulting in an energy saving proportional to the mean number of bits that remain constant between two conversions.

Maximum variation of the ADC output: The maximum variation between consecutive samples of the ADC input voltage, $\Delta v_{in,max}$, can be calculated considering that v_{in} is maintained at its maximum rate of change for a complete sampling period, as illustrated in Fig. 1. Mathematically, $\Delta v_{in,max}$ can be written as

$$\Delta v_{in,max} = \left| \frac{dv_{in}}{dt}(t) \right|_{max} \times \frac{1}{f_s} \quad (1)$$

where $|dv_{in}/dt(t)|_{max}$ is the maximum rate of change of v_{in} and f_s is the ADC sampling frequency. The maximum variation of the ADC output between consecutive samples, N , can be written as a function of $\Delta v_{in,max}$ as follows

$$N = \left\lceil \frac{\Delta v_{in,max}}{LSB} \right\rceil \quad (2)$$

where $LSB = FSR/2^B$ is the code length, FSR is the ADC full scale range, B is the ADC stated resolution in bits, and $\lceil \cdot \rceil$ is the ceiling function.

For illustration purposes, let us suppose that the bandwidth of v_{in} is limited by a brick-wall filter with a cut-off frequency of $f = BW$. Considering this, the maximum rate of change of v_{in} is given by the maximum slope of a sine wave of frequency BW and peak amplitude $FSR/2$. Therefore, (2) can be written as a function of the ADC input

graph-eps-converted-to.pdf

Fig. 1. Illustration of the maximum voltage variation at the ADC input.

voltage bandwidth as

$$N = \left\lceil \frac{\pi \times BW \times FSR}{LSB} \times \frac{1}{f_s} \right\rceil = \left\lceil \frac{2^{B-1} \times \pi}{OSR} \right\rceil \quad (3)$$

where $OSR = f_s / (2 \times BW)$ is the oversampling ratio. In a practical design, N must be calculated considering the actual frequency response of the stage prior to the ADC.

Algorithm: Considering that the maximum variation of the ADC output between two consecutive samples is $\pm N$, the proposed algorithm works as follows: once the current digital output L is computed, a variation of $\pm N$ is applied to L and the results given by $\min(L + N, 2^B - 1)$ and $\max(L - N, 0)$ are bitwise compared to L in order to determine B_c , which is the number of MSBs that will remain constant in the following conversion. Fig. 2 shows a pseudo code implementation for the proposed algorithm, which has been formulated to be written in a hardware description language to synthesize a combinational digital circuit.

```
a = max(L-N, 0) ⊕ L
b = min(L+N, 2^B-1) ⊖ L
c = a ⊕ b
Bc = clz(c)
```

Fig. 2 Pseudo code for the proposed algorithm. The operand \oplus is the bitwise exclusive OR operator and clz is the count leading zeroes function.

Using this algorithm, the operation of an ADC can be adapted considering that the B_c MSBs of the following conversion are already known, so there is no need to compute them again and the mean energy consumption per conversion can be reduced.

In order to compute the mean number of bits that remain constant between two consecutive conversions, $\overline{B_c}$, an expression for the number of codes whose i MSBs remain constant after a variation of $\pm N$ LSBs, C_i , should be computed first as follows

$$C_i = \begin{cases} 2(2^x - N) & i = B - x \\ 2(2^{B-i} - N) + (2^i - 2)(2^{B-i} - 2N) - \sum_{j=i+1}^{B-x} C_j & i = 2 \dots B - x - 1 \\ 2^B - 2N - \sum_{j=2}^{B-x} C_j & i = 1 \end{cases} \quad (4)$$

where $x = \lceil \log_2(N + 1) \rceil$.

Assuming an uniformly distributed input, $\overline{B_c}$ can be calculated as

$$\overline{B_c} = \frac{1}{2^B} \sum_{i=1}^{B-x} i \times C_i \quad (5)$$

Replacing (4) in (5), it can be seen that $\overline{B_c}$ only depends on the OSR , and not on the stated resolution of the ADC.

A variation of the proposed algorithm can be formulated for asynchronous bit-at-a-time ADCs. In these ADCs, the conversion time depends on B_c . Therefore, N is not constant and can be recalculated at the end of each conversion. Assuming that the reset of the ADC takes one clock period, at the i -th conversion, N_i can be computed dynamically as

$$N_i = \left\lceil \frac{2^{B-1} \times \pi}{OSR} \times \frac{B - B_{c,i-1} + 1}{B + 1} \right\rceil. \quad (6)$$

It can be seen that under this variation, the number of bits that remain constant between two consecutive conversions depends on the signal shape and not on the signal statistics, thus, $\overline{B_c}$ cannot be calculated for asynchronous operation.

Behavioural simulation results: The proposed algorithm and its variation with dynamic calculation of N for asynchronous operation were simulated for a 10-bit ADC. Both algorithms were tested using an input ramp spanning the full scale range, and using an input sine wave with a peak amplitude of $FSR/2$ and frequency $f = BW$.

Table 1 shows $\overline{B_c}$ for different values of OSR . As mentioned earlier, $\overline{B_c}$ does not depend on B , so the results shown in Table 1 are also valid for a generic B -bit converter. Using these results and considering the ADC architecture, the mean saved energy consumption per conversion can be estimated. For instance, in a Flash ADC each bit that remains constant between two consecutive conversions allows to turn off 50 % of the comparators, thus, a mean energy reduction of 18.37 % can be reached for $OSR = 4$, whereas a mean energy reduction of 45.32 % can be reached for $OSR = 8$.

OSR	2	4	8	16
FSR ramp, static N	0	0.21	0.71	1.27
FSR ramp, dynamic N	0	0.27	0.85	1.50
$FSR \sin(2 \times \pi \times BW)/2$, static N	0	0.73	1.59	2.42
$FSR \sin(2 \times \pi \times BW)/2$, dynamic N	0	0.77	1.67	2.56

Table 1: Simulated $\overline{B_c}$ as a function of OSR .

Conclusion: A technique to reduce the mean number of bits processed per conversion is presented. The technique is based on estimating a range for the next ADC output and adapt the operation of the circuit accordingly. The implementation of the technique is based on an algorithm that can be synthesized in a simple combinational digital circuit. Additionally, the algorithm allows an speed improvement in asynchronous bit-at-a-time ADCs.

Acknowledgment: The authors thank the National Commission for Scientific and Technological Research (CONICYT) of Chile, through project FONDECYT 11110165, for the funding provided for this research.

D. Avila, E. Alvarez and A. Abusleme (*Department of Electrical Engineering Pontificia Universidad Católica de Chile, PO Box 306 - Correo 22, Santiago 7820436, Chile*)

E-mail: dlavila@uc.cl

References

- 1 Murmann, B.: 'Digitally Assisted Analog Circuits', *Micro. IEEE*, 2006, **26**, no.2, pp. 38-47.
- 2 Trakimas, M.; Sonkusale, S.R., 'An Adaptive Resolution Asynchronous ADC Architecture for Data Compression in Energy Constrained Sensing Applications', *Tran. on Circ. and Sys I*, 2011, **58**, no.5, pp. 921-934.
- 3 O'Driscoll, S.; Shenoy, K.V.; Meng, T.H., 'Adaptive Resolution ADC Array for an Implantable Neural Sensor', *IEEE Trans. on Biom. Cir. and Sys.*, 2005, **5**, no.2, pp. 120-130.
- 4 Zaare, M.; Sepehrian, H.; Maymandi-Nejad, M.; 'A New Non-Uniform Adaptive-Sampling Successive Approximation ADC for Biomedical Sparse Signals', *Analog Integrated Circuits and Signal Processing*, 2013, **74**, no.2, pp. 317-330.
- 5 Harpe, P.; Cantatore, E.; van Roermund, A., 'A 2.2/2.7fJ/conversion-step 10/12b 40kS/s SAR ADC with Data-Driven Noise Reduction', *Tech. Dig. IEEE Int. Solid-State Circ. Conf.*, 2013, pp. 270-271.