A Generic Multilevel Multiplying D/A Converter for Pipelined ADCs

Vivek Sharma, Un-Ku Moon, Gabor C. Temes
Electrical Engg. & Computer Sci.
Oregon State University
Corvallis, USA
Vivek.Sharma@austriamicrosystems.com, moon@ece.orst.edu, temes@ece.orst.edu

***Abstract*—State-of-art implementations of pipelined ADCs can only realize a multiplying DAC (MDAC) with (2^n –1) levels. However, the number of levels needed to optimize the performance may differ from this number. A novel scheme is proposed allowing for realization of an arbitrary number of MDAC levels, while allowing for 1 bit of digital redundancy and digital error correction without any overhead.**

I. INTRODUCTION

Pipelined ADCs find wide use in a variety of applications requiring high-speed, high-resolution operation at a reasonable cost. These benefits are derived from the design flexibility offered by these ADCs which allows the designer to choose the number of levels resolved in each stage. Previous research has shown the importance of choosing the number of levels *m* and also suggested that this number may assume arbitrary values depending on the specifications [1],[2],[3],[5]. However, the practical implementation of these ADCs is possible only if digital redundancy and error correction are included in the design. Digital redundancy allows for more robust design at the cost of a few extra comparators. The use of digital redundancy along with digital error correction makes the overall system tolerant to large comparator offsets and simplifies the design to such an extent that the use of extra comparators is not a cost-limiting factor. In the absence of these desirable features, the advantages of a pipelined ADC are lost. State-of-art implementations only allow for digital redundancy and error correction when the MDAC resolves (2^n –1) levels. Consequently, designers have had to work with a sub-optimal number of MDAC levels and incur a loss in the power vs. speed tradeoff offered by pipelined ADCs. This paper presents a technique to overcome this problem, thereby allowing designers greater freedom in choosing the number of MDAC levels.

A novel scheme for implementing 1 bit of digital redundancy in an *m*-level MDAC is presented in Section II and illustrated with the example of a 5-level MDAC. The relevance of this configuration has been discussed in [4]. Section III covers the technique for implementing digital error correction and presents a simple overlap-and-add scheme for the 5-level MDAC. Finally, the implementation details are briefly reviewed in Section IV and the discussion is concluded in Section V.

II. *m*-LEVEL MDAC WITH 1 BIT OF DIGITAL REDUNDANCY

A. Basic Definitions

Fig. 1 shows the block diagram of a typical MDAC used in a pipelined ADC stage. The ADC in each pipeline stage performs a coarse quantization of its input, and the error resulting from this quantization is then computed and scaled by the DAC and sample-and-hold amplifier (SHA), which are collectively known as the MDAC. This ADC is realized using a few comparators, while the MDAC is typically implemented using a switched-capacitor sample-and-hold amplifier. 1 bit of digital redundancy is usually enough and is included in each pipeline stage. The gain *G* of the SHA can then be written in terms of the number of MDAC levels *m* as

$$2G = m + 1. \qquad (1)$$

B. Example: 5-Level MDAC

The design methodology is now illustrated with the example of a 5-level MDAC with 1 bit of digital redundancy. The signal range of the ADC is assumed to be $-V_{REF}$ to $+V_{REF}$. We use the input-output transfer characteristic of the MDAC, also known as the *residue transfer curve* for our discussion. The scheme presented here is similar to that discussed in [6],[7] for the design of a 3-level MDAC with 1 bit of digital redundancy. The scheme involves halving the SHA gain *G* and leaving out 1 comparator threshold. Thus, we begin with an MDAC stage with *m*+1 levels and with a SHA gain of *2G*. For a 5-level MDAC, this corresponds to a stage with an SHA gain of 6 and with resolving 6 levels with 5 comparators. The residue transfer curve for such a stage is

This work was supported by the Catalyst Foundation, New York, NY.

shown in Fig. 2. This stage has almost no tolerance to comparator offsets and any offset forces the residue out of range introducing nonlinear distortion. Thus, the design of comparators becomes a great challenge.

The next step involves cutting the SHA gain in half to *G* while retaining the old comparator thresholds. The residue transfer curve for this scheme is shown in Fig. 3. Now, comparator errors as large as +/-V_{REF}/6 can be tolerated. This is a significant improvement over the previous case. However, the digital error correction for this case is more complicated. In order to understand this, we need to visualize a 2-stage pipeline where both stages have been designed for a residue transfer curve as shown in Fig. 3. The first stage suffers from comparator offsets which cause deviation from the ideal residue tranfer curve, but are not so large as to overload the next stage.

Now, the errors introduced in the digital output code of the first stage can be corrected by looking at the digital output code of the second stage. In the ideal scenario, the residue output of the first stage is bounded between +/- 0.5 V_{REF}. If the residue output exceeds this range, then it can be viewed as an *overrange error*. Thus, the output codes (000) and (101) never occur in the ideal case. However, a positive or negative comparator offset in the first stage introduces a positive or negative overrange error respectively. Such an error can be detected by looking for the output codes (000) and (101) in the digital output of the second stage, and can be corrected for by subtracting or adding 1 LSB (least significant bit) to the output of the first stage. However, this process involves two steps: (a) detection of the nature of overrange error, and (b) addition or subtraction of 1 LSB depending on the nature of the error. It is desirable to reduce the number of steps involved in the digital error correction.

If we deliberately introduce an offset of V_{REF}/6 (0.5 LSB) in the residue transfer curve using the arrangement of Fig. 4, then the residue transfer curve is altered to that shown in Fig. 5. Now, there can only be a positive overrange error and the digital error correction scheme is simplified. The last comparator threshold is not needed as the overrange error can be easily detected even without it. Thus, it is discarded to yield the residue transfer curve of Fig. 6. Now, the MDAC resolves only *m* levels and has *m*-1 comparators. This is similar in appearance to the residue transfer curves of the well-known 1.5-bit/stage and 2.5-bit/stage MDACs.

The same technique can be applied to realize MDACs with any arbitrary number of levels and 1 bit of digital redundancy.

III. Digital Error Correction

The digital error correction schemes for conventional MDAC implementations are based on an overlap-and-add technique for combining the digital output code from different pipeline stages. This scheme merely accounts for the SHA gain *G* while adding the bit outputs from successive stages of the pipeline. This suggests that digital error correction might be realized similarly for any arbitrary number of MDAC levels. For the example of a 5-level MDAC, digital error correction is easily realized if the final digital output B_{ADC} of a *k*-stage pipeline is computed as:

$$B_{ADC} = 3^{k-1} D_1 + 3^{k-2} D_2 + \ldots + D_k \tag{2}$$

Thus, the digital error correction simply accounts for the SHA gain *G* in the digital domain, which seems reasonable. Eqns. (3)-(5) illustrate the manner in which the bit outputs are combined in 3 different 2-stage pipelined ADCs. Eqn. (3) shows the case m = 4 and no digital redundancy, (4) represents the case m = 3 and 1 bit of redundancy while (5) represents correction for m = 5 and 1 bit of digital redundancy.

$$\begin{array}{l} D_1 = 01, D_2 = 10; Output = 4.D_1 + D_2 \\ \quad\ 0\ \ 1 \\ +\quad\ \ 1\ \ 0 \\ \hline \quad 0\ \ 1\ \ 1\ \ 0 \end{array}; Case(i) - m = 4, no\ redundancy \tag{3}$$

$$\begin{array}{l} D_1 = 01, D_2 = 10; Output = 2.D_1 + D_2 \\ \quad\ \ 0\ \ 1 \\ +\quad\ \ 1\ \ 0 \\ \hline \quad 1\ \ 0\ \ 0 \end{array}; Case(ii) - m = 3,1 - bit\ redundancy \tag{4}$$

$$\begin{array}{l} D_1 = 011, D_2 = 100; Output = 3.D_1 + D_2 \\ \quad\ 0\ \ 1\ \ 1 \\ \quad\ \ 0\ \ 1\ \ 1 \\ +\quad\ 1\ \ 0\ \ 0 \\ \hline \quad 1\ \ 1\ \ 0\ \ 1 \end{array}; Case(iii) - m = 5,1 - bit\ redundancy \tag{5}$$

The proposed scheme for digital error correction has been found to be effective in eliminating the effect of comparator offsets, and detailed discussions may be found in [4]. The validity of this scheme has been verified through behavioral simulations of pipelined ADCs using MATLAB©. An example result (Fig. 7) shows the simulated INL & DNL of a 10-bit ADC with the 6-level MDAC described in the previous sections, in the presence of random comparator offsets within the correction range. The nonlinearity is found to be within the requisite bounds, proving the validity of the proposed scheme for digital error correction.

IV. Implementation Issues

The proposed scheme requires very little change in terms of implementation. The only difference is in the digital error correction where extra adders are required for realization of the SHA gain *G* in the digital domain. In conventional

implementations, G assumes values of the form 2^n and this can be realized easily by left-shifting the digital code from each stage by a certain number of bits. In the proposed scheme, G may assume arbitrary integer values, and these need to be decomposed into bit-shifts. However, the practical values of G required for most m-level MDACs require only 2 terms and hence the cost of the extra digital hardware is minimal. The simplest implementation of G may require use of subtraction instead of addition. For instance, $G = 7$ may be represented as $(4 + 2 + 1)$ or as $(8 – 1)$. The latter representation requires just 2 terms, although it requires the realization of a negative number. This can be done easily, especially with 2's complement code, which is often used for encoding the output of pipelined ADCs.

Another point of interest is the practical values of m used for the MDAC. The proposed scheme may be used to realize MDACs with an arbitrary integer number of levels. However, for even-valued m, the SHA gain G is not an integer, but may assume values such as 1.5, 2.5, 3.5 etc. Although this can still be implemented correctly at the cost of some more digital hardware, it is desirable to avoid this and use only odd-valued m for the MDAC. The reason can be found by observing the residue transfer curve for such a case. The problem can be explained with the residue transfer curve of Fig. 2. This has an even number of MDAC levels, and hence an odd number of comparator thresholds, the middle one of which lies at zero. Although, the A/D conversion is not affected much by this, the arrangement is not preferred if the digitized signal is to be converted back to the analog form later as the presence of a comparator decision level at zero can cause *en masse* switching when there is a transition from 011…1 to 100…0, which can be a potential source of distortion [1]. Thus, it is desirable to avoid such an arrangement and hence it is preferable to use odd-valued m for the MDAC design.

Other than the slight increase in digital hardware, the proposed scheme does not require any special considerations or calibration when compared with conventional schemes.

V. Conclusion

A novel scheme was proposed for implementation of generic m-level MDACs with 1 bit of digital redundancy and error correction. The methodology for incorporating digital redundancy and error correction was presented with an example. The scheme was shown to blend seamlessly with conventional schemes, which are shown to be special cases of the proposed algorithm. The proposed scheme entails almost no design overhead. The scheme allows designers considerably greater freedom in choosing the number of MDAC levels in order to optimize the design tradeoffs. Thus, it makes it realistic to design ADCs which could not possibly be designed using existing techniques. As the design of cyclic ADCs is very similar to that of pipelined ADCs, the proposed scheme is also relevant to this category of ADCs.

Acknowledgment

This research was supported by the Catalyst Foundation, New York, NY.

References

[1] D.W. Cline, "Noise, speed and power trade-offs in pipelined analog-to-digital converters," Ph.D. dissertation, Dept. Elect. Eng. & Comp. Sci., U.C. Berkeley, 1995.

[2] S.H. Lewis, "Video-rate analog-to-digital conversion using pipelined architectures," Ph.D. dissertation, Dept. Elect. Eng. & Comp. Sci., U.C. Berkeley, 1987.

[3] D.W. Cline, P.R. Gray, "A power optimized 13-b 5 Msamples/s pipelined analog-to-digital converter in 1.2 μm CMOS," *IEEE J. Solid-State Circuits*, vol. 35, no. 3, pp. 294-303, Mar. 1996.

[4] V. Sharma, "Generalized radix design techniques for low-power, low-voltage pipelined & cyclic ADCs," M.S. thesis, Dept. Elect. Eng. & Comp. Sci., Oregon State Univ., Corvallis, 2004.

[5] S.H. Lewis, "Optimizing the stage resolution in pipelined, multistage, analog-to-digital converters for video-rate applications," *IEEE Trans. Circuits & Systems II*, vol. 39, no. 8, pp. 516-523, Aug. 1992.

[6] K. Nagaraj *et al.*, "A 250-mW, 8-b, 52-Msamples/s parallel pipelined A/D Converter with Reduced Number of Amplifiers," *IEEE J. Solid-State Circuits*, vol. 32, no. 3, pp. 312-320, Mar. 1997.

[7] S.H. Lewis *et al.*, "A 10-b 20-Msample/s Analog-to-Digital Converter," *IEEE J. Solid-State Circuits*, vol. 27, no. 3, pp. 351-358, Mar. 1992.

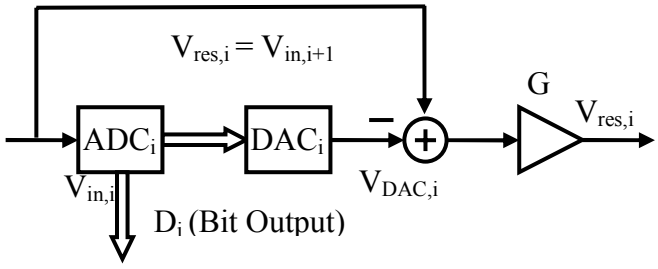

Fig. 1 Simplified representation of a pipeline stage (Stage i).

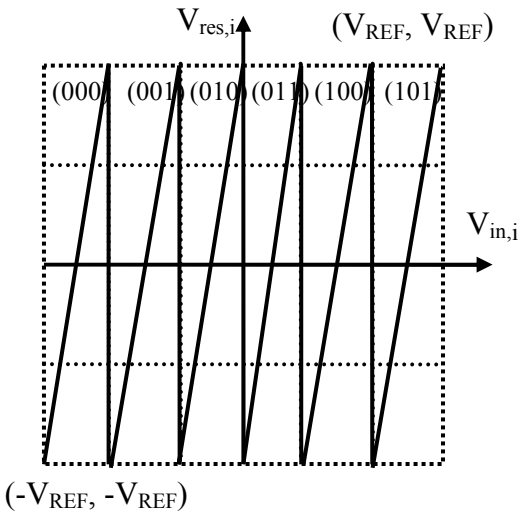

Fig. 2 Residue transfer curve of a 6-level MDAC with no digital redundancy.

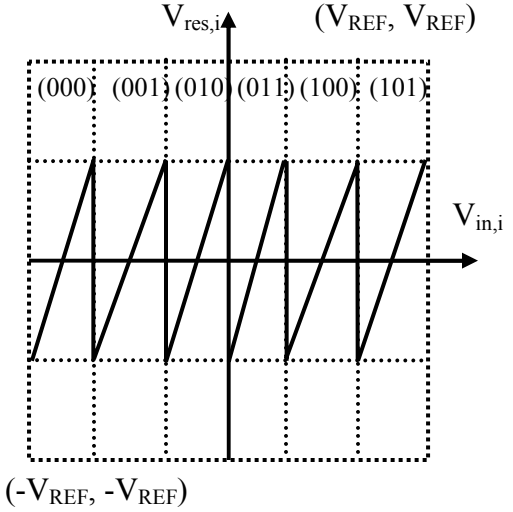

Fig. 3 Residue transfer curve of a 6-level MDAC with 1 bit of digital redundancy.

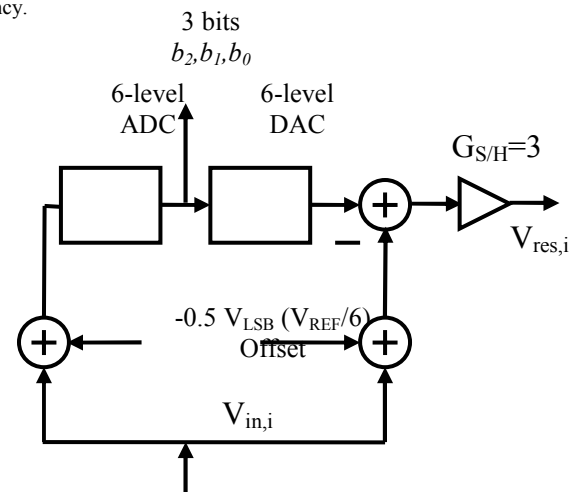

Fig. 4 Modification to the 6-level MDAC to simplify digital error correction.

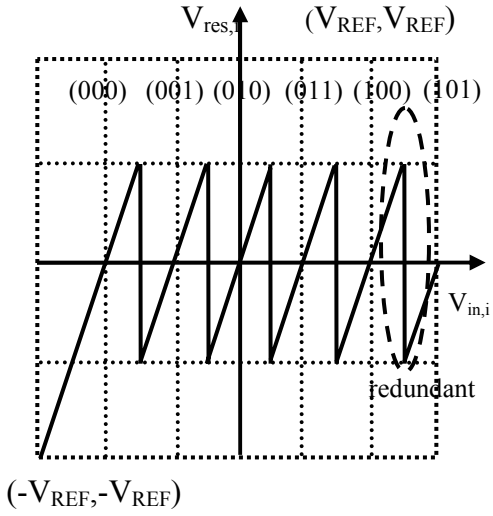

Fig. 5 Modified residue transfer characteristic with redundant comparator threshold (circled).

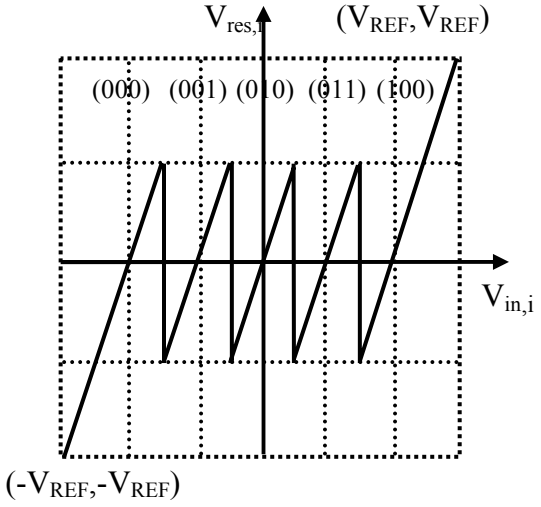

Fig. 6 Final residue transfer curve of a 5-level MDAC with 1 bit of digital redundancy.

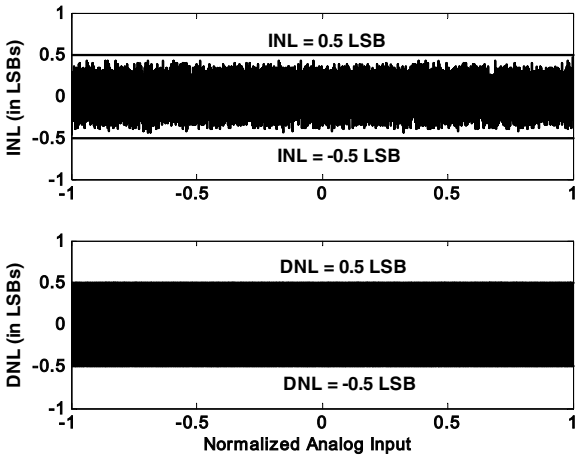

Fig. 7 Simulated INL & DNL of a 10-bit ADC with a 6-level MDAC.