LETTER

Algorithms for Digital Correction of ADC Nonlinearity

Haruo KOBAYASHI[†a)], *Regular Member*, Hiroshi YAGI[†], Takanori KOMURO[††], *and* Hiroshi SAKAYORI[††], *Nonmembers*

SUMMARY This paper describes two digital correction algorithms for ADC nonlinearity, targeted for mixed-signal LSI tester applications: an interpolation algorithm and a stochastic algorithm. Numerical simulations show that our algorithms compensate for ADC nonlinearity as well as missing codes and nonmonotonicity characteristics, and improve ADC SNDR and SFDR.

key words: *ADC, nonlinearity, digital error correction, digital signal processing, LSI tester*

1. Introduction

The performance of electronic devices is continuously improving, and accordingly there is an ever-growing demand for fast new instruments such as LSI testers to measure their performance [1]. LSI testers have to be designed with today's (relatively low performance) devices to measure tomorrow's (higher performance) devices, and hence performance (speed and accuracy) is the first priority in their development. Low power consumption, small size, and implementation with popular monolithic CMOS consumer devices, are desirable but of lower priority. Analog-to-Digital converters (ADCs) incorporated in such instruments have to operate at high sampling rates with high accuracy. In order to implement very high-speed ADCs, time-interleaved architectures are often used—even though there are channel mismatch issues [2]–[5] which require complex calibration systems to compensate for. In this paper, we focus on the implementation of high accuracy ADCs for mixed-signal LSI tester applications. By adding circuitry to a given ADC, and using our digital processing algorithms, linearity, Signal-to-(Noise+Distortion) Ratio (SNDR) and/or Spurious Free Dynamic Range (SFDR) [6] can be improved. Note that compensation for ADC nonlinearity is difficult, even though offset and gain errors are relatively easy to compensate for, and several digital and analog domain methods for these have already been proposed [7]. Implementation of our compensation algorithms for ADC nonlinearity may require relatively complex and costly hardware and software which may preclude their use in consumer product applications, but still be reasonable for LSI tester systems.

Section 2 describes problem formulation, Sects. 3 and 4 present proposed algorithms, while Sect. 5 shows numerical simulation results. Section 6 provides concluding remarks.

2. Problem Formulation

Consider a p-bit ADC whose input-output characteristics may exhibit some nonlinearities, whose input analog voltage range is from 0 to A_{FS}, and whose output digital codes are $0, 1, 2, \ldots, N-1$ ($N = 2^p$). Then define analog input voltage A_k as follows (Fig. 1):

- $A_0 := 0$
- A_k is the input voltage where the ADC output digital code changes from $k-1$ to k ($k = 1, 2, 3, \ldots, N-1$).
- $A_N := A_{FS}$.

The analog voltages $A_0, A_1, A_2, \ldots, A_N$ are measured and converted to digital values $M_0, M_1, M_2, \ldots, M_N$ which are stored in digital memory (Fig. 2). When D is the digital output of the ADC, we consider here how to calculate a corrected digital output from the value of D and $M_0, M_1, M_2, \ldots, M_N$. Suppose that when digital output D is equal to n, the corresponding corrected digital output C_n is given by

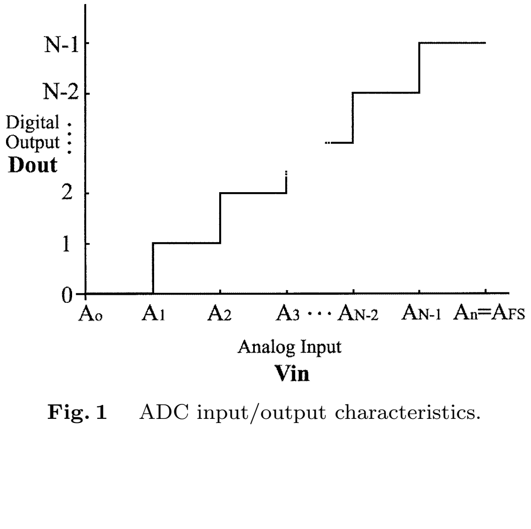


Fig. 1 ADC input/output characteristics.

Manuscript received January 30, 2002.
Manuscript revised August 6, 2002.
Final manuscript received October 22, 2002.

[†]The authors are with the Department of Electronic Engineering, Faculty of Engineering, Gunma University, Kiryu-shi, 376-8515 Japan.
[††]The authors are with Agilent Technologies Japan, Ltd., Hachioji-shi, 192-8510 Japan.
a) E-mail: k_haruo@el.gunma-u.ac.jp

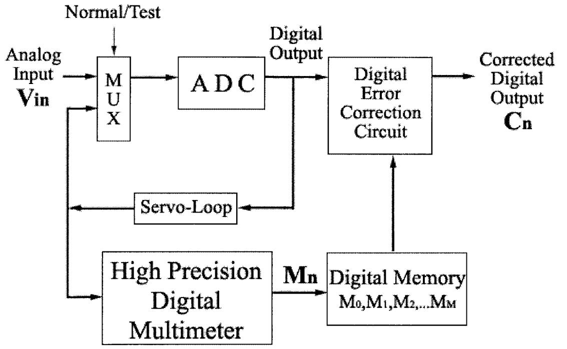

Normal/Test
Analog Input Vin
M U X
ADC
Digital Output
Digital Error Correction Circuit
Corrected Digital Output Cn
Servo-Loop
High Precision Digital Multimeter
Mn
Digital Memory
M0,M1,M2,...MM

Fig. 2 Digital correction system for ADC nonlinearity.

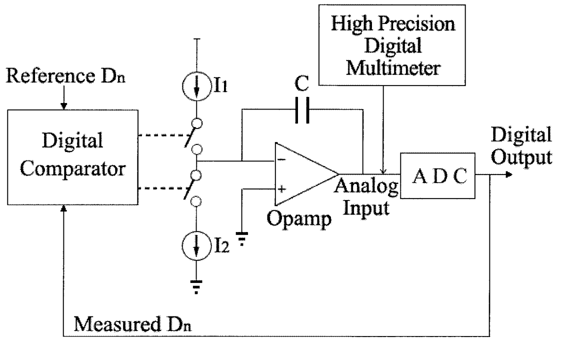


Fig. 3 Servo-loop circuitry to measure the ADC static characteristics. Some of these circuits can be embedded in a device under test (ADC chip).

$$C_n = f(n, M_0, M_1, M_2, \ldots, M_N). \tag{1}$$

Note that all of $C_n, n, M_0, M_1, M_2, \ldots, M_N$ are digital values, so digital signal processing can be used to calculate C_n.

In the following sections, we show two compensation algorithms which calculate C_n so as to improve SNDR and SFDR of the ADC.

Remark (i) The servo-loop method [6] can be used to measure the values of $A_0, A_1, A_2, \ldots, A_N$ (Fig. 3), and some of the servo-loop method test circuitry can be included in devices to facilitate testing.

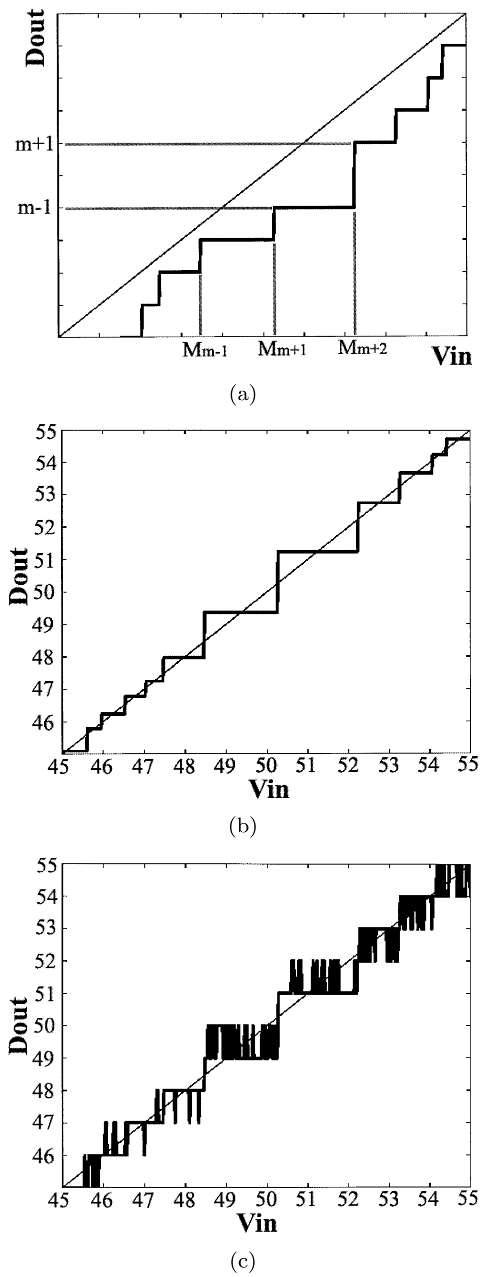
(ii) The rapid progress of VLSI technology has reduced the cost of digital circuit and improved its performance significantly. Hence the cost of digital memory to store $M_0, M_1, M_2, \ldots, M_N$, and the speed and cost of digital signal processing circuitry to calculate C_n are unlikely to be a big burden.

(iii) However, the progress in VLSI technology does not benefit analog performance as much as digital performance, hence it still remains difficult to develop and produce precision analog circuits. The compensation method that is proposed here improves ADC precision utilizing digital rather than analog circuit technology.

3. Interpolation Algorithm

In this section, we propose a digital interpolation algorithm which improves SNDR and SFDR of ADCs. When the ADC digital output D is equal to n, the digital correction output C_n is given by

$$C_n := \frac{1}{2}\frac{N}{M_{FS}}(M_n + M_{n+1}). \tag{2}$$



(a)

(b)

(c)

Fig. 4 (a) A part of ADC characteristics with a missing code m. (b) The corresponding ADC characteristics after the interpolation algorithm is applied. (c) The corresponding ADC characteristics after the stochastic algorithm is applied.

Remark (i) The digital output D is given in p-bit form, but the corrected digital data C_n is given in $(p+q)$-bit form $(q > 0)$; in other words, the corrected digital data C_n require q extra bits.

(ii) Let us consider the case that an ADC has some

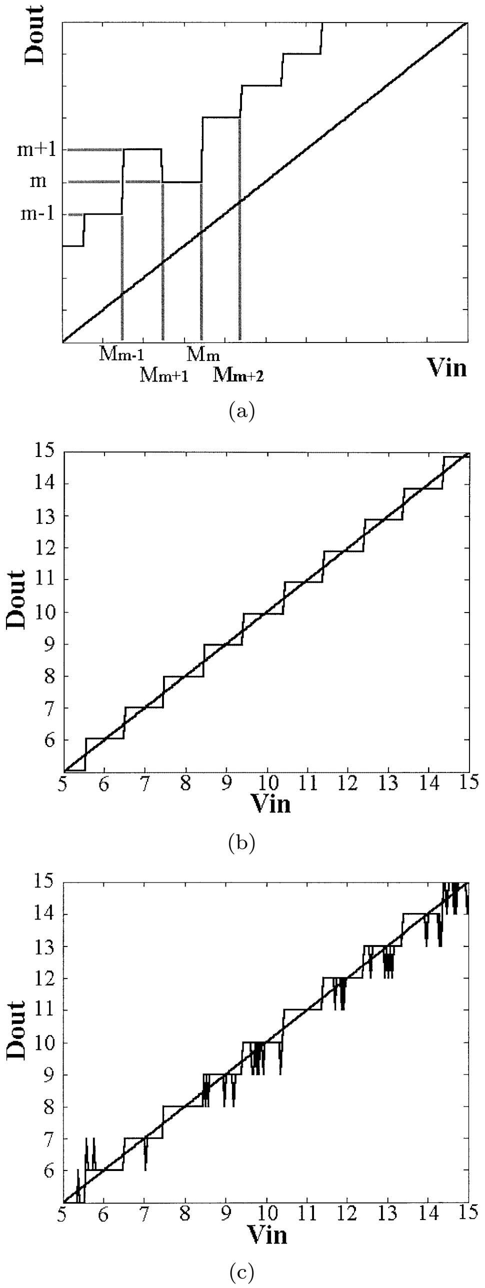


Fig. 5 (a) A part of ADC characteristics when the input-output monotonicity is violated. (b) The corresponding ADC characteristics after the interpolation algorithm is applied. (c) The corresponding ADC characteristics after the stochastic algorithm is applied.

missing codes as shown in Fig. 4(a); for example, the ADC output does not produce m, and hence A_m and M_m do not exist. In this case, Eq. (2) around the missing code m should be modified as follows: when the ADC digital output D is equal to $m-1$, the digital correction output C_{m-1} is given by

$$C_{m-1} := \frac{1}{2}\frac{N}{M_{FS}}(M_{m-1} + M_{m+1}).$$

Figure 4(b) shows the ADC characteristics after this algorithm is applied.

(iii) Let us consider the case that the monotonicity of the ADC input-output characteristics is violated as shown in Fig. 5(a), where $M_{m-1} < M_{m+1} < M_m < M_{m+2}$. Then the algorithm for codes $m-1$, m and $m+1$ should be modified as follows:

$$C_{m-1} := \frac{1}{2}\frac{N}{M_{FS}}(M_{m-1} + M_{m+1})$$

$$C_{m} := \frac{1}{2}\frac{N}{M_{FS}}(M_{m} + M_{m+2})$$

$$C_{m+1} := \frac{1}{2}\frac{N}{M_{FS}}(M_{m} + M_{m+1}).$$

Figure 5(b) shows the ADC characteristics after this algorithm is applied.

4. Stochastic Algorithm

In this section, we propose a digital "stochastic algorithm" which especially improves SFDR of ADCs. When the ADC digital output D is equal to n, calculate K_n by

$$K_n := \frac{1}{2}\frac{N}{M_{FS}}(M_n + M_{n+1}). \tag{3}$$

Next let us decompose K_n as follows:

$$K_n = I_n + F_n, \tag{4}$$

where I_n is the integer part of K_n and F_n is the fractional part of K_n; I_n is equal to 0 or a positive integer and $0 \leq F_n < 1.0$. Then the corrected digital output C_n is given by

$$C_n = \begin{cases} I_n & \text{with probability of } 1 - F_n \\ I_n + 1 & \text{with probability of } F_n. \end{cases} \tag{5}$$

For example, when $I_n = 15$ and $F_n = 0.3$ and we have 10 samples of output data $D = n$, then 7 samples of C_n are 15, 3 samples of C_n are 16, and outputs 15 or 16 occur in pseudo-random order.

Remarks (i) The average of C_n is given by

$$\overline{C_n} = I_n(1 - F_n) + (I_n + 1)F_n = I_n + F_n = K_n.$$

(ii) The corrected digital data C_n and the digital output D are both represented in p-bit form; their number of bits is the same. Hence this algorithm does not require that connected digital systems which handle p-bit digital data input be modified. On the other hand, the output of the interpolation algorithm is represented by $(p+q)$-bits (which is larger than p-bits), and hence connected digital systems have to accept $(p+q)$-bit digital data input.

(iii) When we apply this algorithm to DC input, the

ADC output fluctuates by 1LSB (because when $D = n, C_n = I_n$ or $C_n = I_n + 1$), which is sometimes described as "1LSB noise." This algorithm can be used for applications where "1LSB noise" is acceptable.

(iv) Let us consider the case again that an ADC has some missing codes as shown in Fig. 4(a); the ADC output does not produce m, and hence A_m and M_m do not exist. The algorithm for missing code parts should be modified as follows: when the ADC digital output D is equal to $m - 1$, calculate K_n by

$$K_{m-1} := \frac{1}{2}\frac{N}{M_{FS}}(M_{m-1} + M_{m+1}). \qquad (6)$$

Similarly decompose K_{m-1} as follows:

$$K_{m-1} = I_{m-1} + F_{m-1}, \qquad (7)$$

where I_{m-1} is equal to 0 or a positive integer and $0 \leq F_{m-1} < 1.0$. Then the corrected digital output C_{m-1} is given by

$$C_{m-1} = \begin{cases} I_{m-1} & \text{with probability of } 1 - F_{m-1} \\ I_{m-1} + 1 & \text{with probability of } F_{m-1}. \end{cases} \qquad (8)$$

Figure 4(c) shows the ADC characteristics after this algorithm is applied.

(v) Let us consider the case that the monotonicity of the ADC input-output characteristics is violated as shown in Fig. 5(a), where $M_{m-1} < M_{m+1} < M_m < M_{m+2}$. Then Eq. (6) for codes $m - 1$ and $m + 1$ should be modified as follows:

$$K_{m-1} := \frac{1}{2}\frac{N}{M_{FS}}(M_{m-1} + M_{m+1})$$

$$K_{m+1} := \frac{1}{2}\frac{N}{M_{FS}}(M_m + M_{m+2}).$$

Note that K_m is defined by

$$K_m := \frac{1}{2}\frac{N}{M_{FS}}(M_m + M_{m+1})$$

which is the same as Eq. (3). Figure 5(c) shows the ADC characteristics after this algorithm is applied.

5. Numerical Simulations

We have conducted numerical simulations to evaluate our algorithms. Eight-bit ADCs with some nonlinearities are used and Tables 1, 2 and 3 summarize the simulation results, where units of SNDR and SFDR are dB. We see that SNDR and SFDR are improved by our algorithms. The ADC input-output characteristics used

Table 1 Case 1 of numerical simulation result.

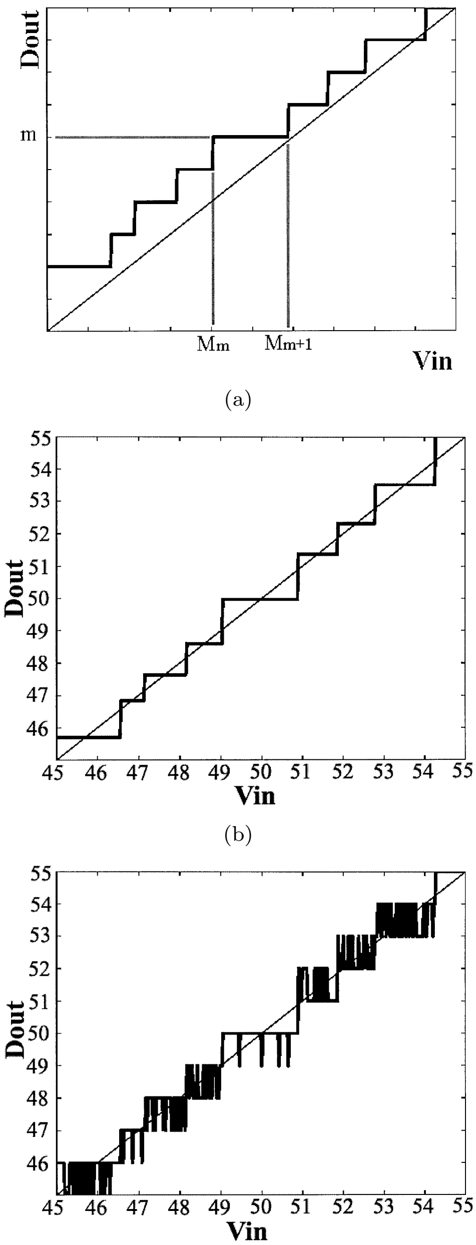
Original ADC output		Interpolation Algorithm		Stochastic Algorithm	
SNDR	SFDR	SNDR	SFDR	SNDR	SFDR
33.2	34.7	49.2	64.0	45.1	63.5

Table 2 Case 2 of numerical simulation result.

Original ADC output		Interpolation Algorithm		Stochastic Algorithm	
SNDR	SFDR	SNDR	SFDR	SNDR	SFDR
35.6	41.2	46.7	65.9	43.6	69.2

Table 3 Case 3 of numerical simulation result.

Original ADC output		Interpolation Algorithm		Stochastic Algorithm	
SNDR	SFDR	SNDR	SFDR	SNDR	SFDR
33.1	35.7	49.0	66.6	45.1	65.4



(a)

(b)

(c)

Fig. 6 (a) A part of ADC characteristics used in case 1 (Table 1). (b) The corresponding ADC characteristics after the interpolation algorithm is applied. (c) The corresponding ADC characteristics after the stochastic algorithm is applied.

in case 1 is monotonic and has no missing codes. Figure 6(a) shows a part of the original characteristics, and Fig. 6(b) shows the corresponding characteristics after the interpolation algorithm is applied while Fig. 6(c) shows the one after the stochastic algorithm is applied. The ADC input-output characteristics used in case 2 is monotonic but has a missing code as shown in Fig. 4, and also that in case 3 it has no missing codes but is not monotonic as shown in Fig. 5. Note that SNDR and SFDR of an ideal 8-bit ADC are 49.8 [dB] and 67.3 [dB] respectively.

Remark The stochastic algorithm is especially useful for improving SFDR. However, our simulations show that when ADC nonlinearity is small, applying this algorithm sometimes leads to slight degradation of SNDR (though when the ADC nonlinearity is modest to large, SNDR usually improves as shown above).

6. Concluding Remarks

- It is known [3]–[5] that time-interleaved ADC systems suffer from channel mismatch problems which especially cause deterioration in SFDR. The proposed algorithms would alleviate this problem.
- The algorithms described here are to correct the DC characteristics of ADCs, but digital algorithms for correcting AC characteristics would also be interesting.

Acknowledgement

We would like to thank K. Wilkinson for valuable discussions.

References

[1] M.L. Bushnell and V.D. Agrawal, Essentials of Electronic Testing for Digital, Memory and Mixed-Signal VLSI Circuits, Kluwer Academic Publishers, Boston, 2000.

[2] H. Kobayashi, K. Kobayashi, H. Sakayori, and Y. Kimura, “ADC standard and testing in Japanese industry,” Computer Standards & Interfaces, vol.23, pp.57–64, Elsevier Publishers, March 2001.

[3] N. Kurosawa, H. Kobayashi, K. Maruyama, H. Sugawara, and K. Kobayashi, “Explicit analysis of channel mismatch effects in time-interleaved ADC systems,” IEEE Trans. Circuits Syst. I, Fundam. Theory Appl., vol.48, no.3, pp.261–271, March 2001.

[4] N. Kurosawa, H. Kobayashi, and K. Kobayashi, “Channel linearity mismatch effects in time-interleaved ADC systems,” IEICE Trans. Fundamentals, vol.E85-A, no.3, pp.749–756, March 2002.

[5] M. Tamba, A. Shimizu, H. Munakata, and T. Komuro, “A method to improve SFDR with random interleaved sampling method,” Proc. International Test Conference, pp.512–520, 2001.

[6] B. Razavi, Principles of Data Conversion System Design, IEEE Press, 1995.

[7] D. Fu, K.C. Dyer, S.H. Lewis, and P.J. Hurst, “A digital background calibration technique for time-interleaved analog-to-digital converters,” IEEE J. Solid-State Circuits, vol.33, no.12, pp.1904–1911, Dec. 1998.