The Realization of a Mismatch-free and 1.5-bit Over-sampling Pipelined ADC

Shigeto Tanaka
Department of E.E.C.E.
Chuo University
Tokyo, Japan 112-8551
Email: tanaka@sugi.elect.chuo-u.ac.jp

Yuji Ghoda
Graduate School of E.E.C.E.
Chuo University
Tokyo, Japan 112-8551
Email: goda@sugi.elect.chuo-u.ac.jp

Yasuhiro Sugimoto
Department of E.E.C.E.
Chuo University
Tokyo, Japan 112-8551
Email: sugimoto@sugi.elect.chuo-u.ac.jp

***Abstract*— This paper proposes a simple method to realize an over-sampling pipelined analog-to-digital converter (ADC) with 1.5-bit bit-blocks. The ADC performs conversion by permuting internal capacitors in alternate clocks of the upper 1.5-bit bit-blocks in the analog domain, then averaging the data from bit-blocks in the digital domain. The behavioral simulation of a 14-bit ADC verified that this over-sampling pipelined ADC with 1.5-bit bit-blocks has more than 70 dB of spurious-free dynamic range (SFDR) for up to an 8 MHz input signal when each of the upper three bit-blocks has a gain error of $+0.8$ %. Using a S/H circuit in front improves the SFDR to 95 dB up to the signal frequency bandwidth of 25.6 MHz when the clock frequency is 102.4 MHz.**

I. Introduction

Although the pipelined ADC is very popular, its accuracy is generally limited to 10- to 12-bit equivalents, due to device mismatches such as capacitors that cause gain and offset errors [1].

Various ways to eliminate these errors have been demonstrated. Performing a calibration, either in the fore-ground [2] or in the back-ground, [3] is very effective. However, the calibration method requires precise error measurements, additional hardware and complicated algorithms.

Among the methods, those that are most promising include the over-sampling method [4] and the capacitor averaging method [5],[6]. The over-sampling method inverts the input signal and multiplies the output digital code by -1 at one of the alternate clocks. This rearranges the errors due to device mismatch in the opposite direction of the frequency region compared to the errors produced in the normal operation. However, this method's application are limited to a pipelined ADC with a 1-bit bit-block, even though most pipelined ADC adopts a 1.5-bit architecture. The capacitor averaging method is considered one of the derivatives of the over-sampling method. It reduces the capacitor mismatch error by taking the average of the size equivalence of the positive and negative errors that are produced by permuting capacitors in an error amplifier. However, until now the averaging has been done in the analog domain. It would be natural for the ADC to produce digital data with positive and negative errors alternately, so that the averaging could be done in digital domain.

The purpose of this paper, therefore, is to offer a simple and effective over-sampling method for a pipelined ADC with 1.5-bit bit-blocks, with an averaging method performed in the digital domain.

II. An over-sampling pipelined ADC with 1.5-bit bit-blocks

In the over-sampling pipelined ADC reported in reference 4, the error produced in the bit-block is an even function around the center of the input-versus-output voltage relationship. However, in a pipelined ADC with 1.5-bit architecture, the error becomes an odd function. Figure 1 shows transfer curves between the input signal voltage and the output signal voltage of a 1.5-bit bit-block.

In Figure 1, the input and output voltage values have the same range and are normalized to be $\pm$ 1. When the input voltage exceeds $+$ 1/4, digital code "1" is generated, and digital code "0" is generated when the input voltage is less than $-$ 1/4. When the input voltage is between $-$ 1/4 to $+$ 1/4, the digital code is not fixed in the current bit-block, and the decision is postponed until the digital codes of the succeeding bit-blocks are fixed. A solid line in the figure represents the ideal transfer curve. The chain line shows the one with the positive gain error caused by the capacitor mismatch, and the dotted line with the negative gain error. It is evident that the difference between the ideal transfer curve and the error curve becomes the odd function around the center. It is also seen in Figure 1 that the gain error is cancelled when both curves with positive and negative errors are added because they are

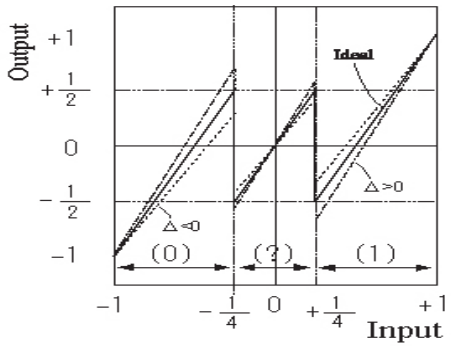

Fig. 1. The input vs. output voltage relationship of a 1.5-bit bit-block.

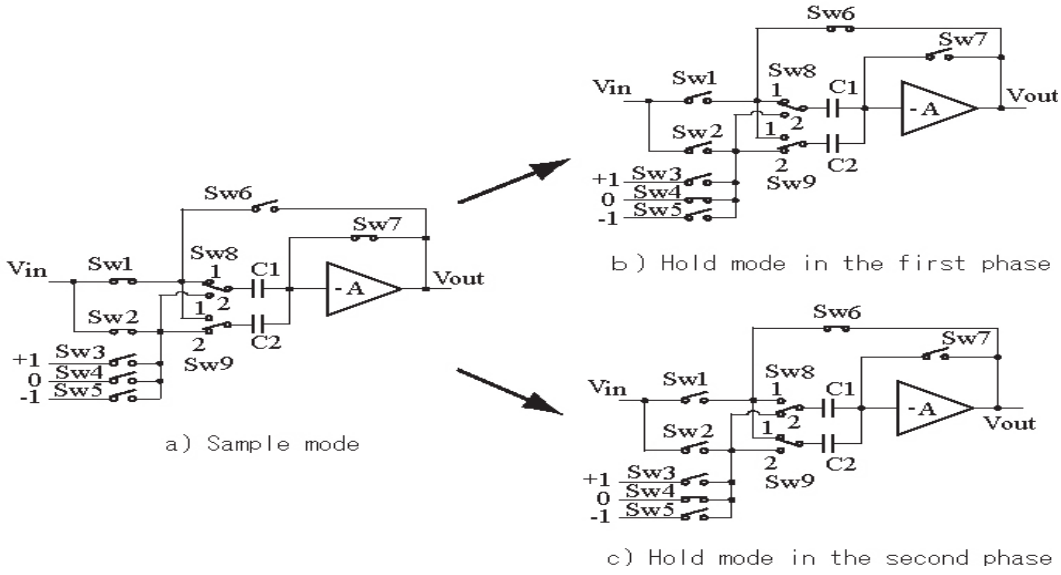

Fig. 2. The mismatch error reduction scheme.

of equal amounts but in different polarities.

This observation leads us to form a 1.5-bit bit-block in which the permutation of capacitors is performed at alternate sampling clocks as shown in Figure 2 as a circuit of an MOS pipelined ADC. The circuit resembles that of the capacitor averaging method in reference 5; in our case, however, the averaging is performed in the digital domain, and the control sequence to produce the digital code from the bit-block is different. The circuits consist of a sample-and-hold circuit, a sub-DAC, and a residue amplifier. Figure 2(a) shows the circuit in sample mode, with the input signal stored in capacitors C1 and C2. We assume a capacitor mismatch between C1 and C2, that is, $C_1 = C$, and $C_2 = C\,(1+\Delta)$, where Δ is the relative mismatch error. Because the charge stored in the capacitors is equal in both the sample and hold modes, equation (1) is derived for the circuit shown in Figure 2(b). V_{REF} in equation (1) becomes $+1$, 0, or -1, depending on which switch (SW3, SW4, or SW5) turns on.

$$\begin{aligned} V_{out} &= \left(1+\frac{C_2}{C_1}\right)V_{in} - \frac{C_2}{C_1}V_{REF} \\ &= \{1+(1+\Delta)\}\,V_{in} - \{(1+\Delta)\,,0,or-(1+\Delta)\} \end{aligned} \tag{1}$$

Figure 2(c) has a configuration such that the feedback capacitor of the amplifier is C2 instead of C1, in contrast to that shown in Figure 2(b). In this case, the V_{out}, with one of the three switches Sw3, Sw4, and Sw5 on and with other switches in the positions shown in Figure 2(c), becomes

$$\begin{aligned} V_{out} &= \left(1+\frac{1}{1+\Delta}\right)V_{in} - \left\{\frac{1}{1+\Delta},0,or\frac{-1}{1+\Delta}\right\} \\ &\approx \{1+(1-\Delta)\}\,V_{in} - \{(1-\Delta)\,,0,or-(1-\Delta)\} \end{aligned} \tag{2}$$

It is evident that the capacitor mismatch error Δ is cancelled out by summing equations (1) and (2). In the proposed ADC, the circuits shown in Figure 2(a) and 2(b) are used for the sample and hold mode operation in one clock period of alternate sampling clocks, while the circuits shown in Figure 2(a) and 2(c) are used for the sample and hold mode operation in the other clock period. Figure 2(a) is common to the sample mode operation in both clock periods. As a result, we obtain a digital output code from the bit-block at each of the sampling clocks, but the digital output code which is the A-D conversion results from either of the circuits shown in Figure 2(b) and 2(c) at alternate sampling clocks. The error obtained from the bit-block in digital form in one clock period has a polarity opposite to that of the other clock period, as long as the input signal frequency is low. It therefore becomes possible to eliminate the error at the output of the ADC in digital form.

The whole ADC is constructed in such a way as shown in Figure 3. The 1.5-bit bit-block consists of a sample-and-hold circuit (S/H), a sub-ADC, a sub-DAC, a subtraction circuit, and an amplifier producing a voltage gain of 2. As described previously, the capacitors in the bit-block are permutated in hold mode at alternate sampling clocks, and the digital code from a bit-block contains the positive and negative errors in turn. Therefore, the output of the digital correction logic shown in Figure 3 contains the positive and the negative errors in turn, as well. In order to eliminate the offset error and gain error that are included in the alternative digital outputs in different polarities, the averaging function using a 1-clock delay and an adder is introduced. The absolute value of the transfer function of the averaging function by using Z-transform is expressed as,

$$|H(j\omega)| = |1+e^{-j\omega T}| = \sqrt{2\,\{1+\cos(\omega T)\}} \tag{3}$$

where T is the time interval of the sampling clock periods. As

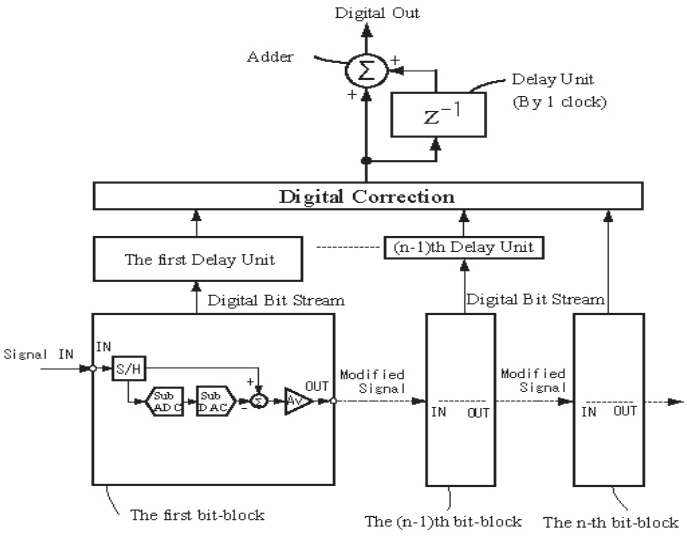

Fig. 3. Block diagram of the proposed ADC with 1.5-bit bit-blocks and an averaging function at the output.

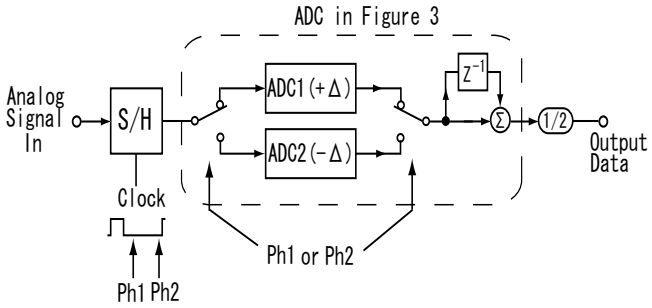

Fig. 4. The ADC with a sample-and-hold function in front of the ADC in Figure 3.

equation 3 becomes zero when

$$f = \frac{\omega T}{2\pi T} = \frac{(2n-1)}{2T} \tag{4}$$

where n is the integer, the frequency characteristics of the ADC in Figure 3 have the null at odd integer multiples of half the sampling clock frequency. This means that the offset error is completely eliminated because its frequency is half the sampling clock frequency, and that the gain error at low frequency is minimized also because its frequency is in the vicinity of half the sampling frequency.

III. The Over-sampling pipelined ADC with an input sample-and-hold

The proposed ADC produces output data by summing two consecutive conversion results from digital correction logic as shown in Figure 3. The configuration eliminates the constant offset error; however, the error caused by the gain error remains when the input signal level changes between consecutive sampling periods.

In order to solve the problem, a S/H circuit was placed in front of the over-sampling pipelined ADC, as shown in Figure 4. Half of a clock period is used for sampling the input signal, and the following one-and-a-half clock periods are used for the hold operation. Two clock periods are needed to complete the operation in this S/H circuit. The over-sampling pipelined ADC shown in Figure 3 performs A-D conversions with errors in opposite polarities at alternate clocks, and is considered to have two different ADCs (though in reality they comprise just one ADC) with a positive error and a negative error, which are combined at the output as shown in Figure 4. In the hold period of the sample-and-hold circuit, the ADC converts its input signal twice, first by ADC1 and then by ADC2. As the output data of ADC1 and ADC2 have errors that have been analyzed in equations (1) and (2), the errors of these two output data are cancelled out by taking their average, and the non-linearity disappears. In this configuration, shown in Figure 4, the sampling frequency and the signal frequency bandwidth become $1/2$ and $1/4$ of the clock frequency, respectively.

IV. The simulation results

In order to demonstrate the effectiveness of the proposed ADCs shown in Figures 3 and 4, the behavioral simulations with a C program were performed. The program consisted of 14 1.5-bit bit-blocks, delay units and the digital correction logic to form a 14-bit ADC. It imitates the ADC shown in Figure 3. Gain errors at each of the first three bit-blocks were set at $+0.8$ %. Clock timing was assumed to be aligned and the timing jitter was not taken into account.

Figure 5 shows the Fast-Fourier-Transform (FFT) result of the output data. The rest of bit-blocks are assumed ideal. The input signal voltage and frequency are ± 2 V and 1.00625 MHz, respectively, and the sampling clock frequency is 102.4 MHz. It does not have the function to permutate capacitors or to take the average of the output data at alternate clocks. The number of collected data points is $16,384$. As seen in Figure 5, gain errors produce harmonic components in the frequency domein and odd multiple of harmonics become eminent because the gain error in the bit-block is the odd function. The signal frequency bandwidth is 51.2 MHz and the SFDR is measured 57.5 dB.

When the permutation of capacitors in the first three bit-blocks at alternative clocks is done, the frequency spectrum changes such that as shown in Figure 6. The averaging function at the output has not been taken yet. Harmonic components become aligned in the reverse order, in contrast to the Figure 5, in the frequency domain starting from 51.2 MHz. The SFDR doesn't change much from that of Figure 5 and is measured 56.5 dB with the signal frequency bandwidth of 51.2 MHz.

The improvement of the SFDR is further achieved by taking the sum of the ADC outputs at alternate clocks by using a unit delay function and an adder as shown in Figure 3. As equation (4) shows, the averaging function introduces zero at multiples of half the clock frequency. The offset error which is caused by permuting capacitors has the frequency component at half the clock frequency, and is eliminated by

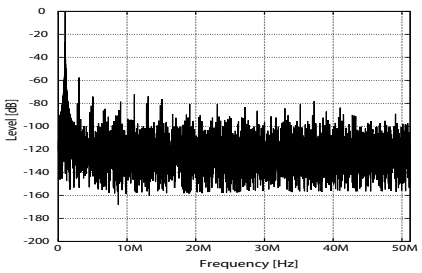

Fig. 5. The frequency spectrum of the simulated conventional 14-bit ADC.

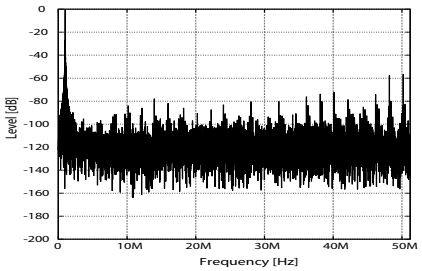

Fig. 6. The frequency spectrum of the simulated 14-bit ADC with permuting capacitors.

the zero. However, the introduction of the zero reduces the signal frequency bandwidth to half the clock frequency.

When the input signal frequency increases, however, the cancellation of the non-linearity becomes less effective. Figure 7 shows the input signal frequency vs. SFDR characteristics. Due to the complex error function, the SFDR degradation does not become linear, and the SFDR goes down to approximately 55 dB at one-fourth the clock frequency of 25.6 MHz.

The complete cancellation of the harmonics or distortion components was achieved by A-D converting the held output value of the S/H circuit twice as shown in Figure 4. Figure 8 shows the output frequency spectrum. The frequency range in the figure is limitted to 25.2 MHz because the signal frequency bandwidth remains 25.2 MHz. The input signal was sampled by an input S/H circuit at an equivalency rate of 51.2 MHz when the clock frequency was 102.4 MHz. In the hold mode of the S/H circuit, two rounds of A-D conversions were performed by permuting capacitors. As the input signal was constant, the offset error and the gain error contained in the two conversion digital data cancelled each other out. With an ideal S/H circuit and with aligned clocks, the SFDR reached 95.6 dB. The SFDR value does not depend on the input signal frequencies.

Table-1 summarizes the relationships of the input signal frequency bandwidth, the sampling clock frequency, and the SFDR among various types of ADCs discussed in this paper.

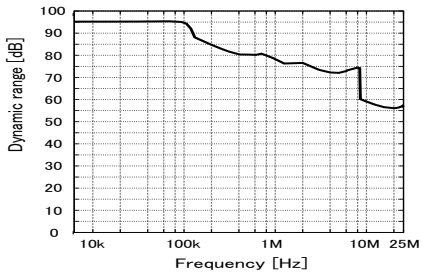

Fig. 7. The input signal frequency vs. SFDR characteristics of the ADC in Figure 3.

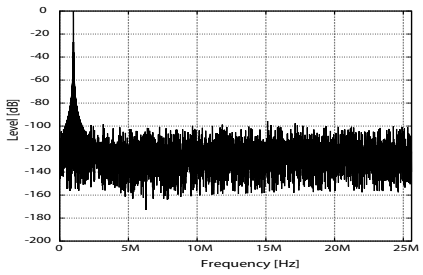

Fig. 8. The frequency spectrum of the simulated 14-bit ADC in Figure 4.

TABLE 1
ADC performance comparison

ADCs	BW	fs	SFDR (1.00625 MHz Input)
Conventional	$51.2MHz$	$102.4MHz$	$57.5dB$
Permuting	$51.2MHz$	$102.4MHz$	$56.5dB$
ADC in Fig. 3	$25.6MHz$	$102.4MHz$	$78.7dB$
ADC in Fig. 4	$25.6MHz$	$51.2MHz$	$95.6dB$

V. Conclusion

This paper offers a very simple implementation of an over-sampling ADC with 1.5-bit bit-blocks. Using permuting capacitors in a bit-block with alternate clocks in the analog domain, offset and gain error cancellation became possible by taking their average in the digital domain.

References

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

[2] S-Y.Chuang and T.L.Sculley, "A Digitally Self-Calibrating 14-bit 10-MHz CMOS Pipelined A/D Converter", IEEE J. of Solid State Circuits, Vol. 37, no. 6, pp.674-683, June 2002.

[3] Y-M.Lin, B.Kim and P.R.Gray, "A 13-b 2.5-MHz Self-Calibrated Pipelined A/D Converter in 3-μm CMOS", IEEE J. of Solid State Circuits, Vol. 26, no. 4, pp.628-636, Appril 1991.

[4] A.Shabra and H.S.Lee, "Oversampled Pipeline A/D Converters With Mismatch Shaping", IEEE J. of Solid State Circuits, Vol. 37, no. 5, pp.566-578, May 2002.

[5] Y.Chiu, "Inherently Linear Capacitor Error-Averaging Techniques for Pipelined A/D Conversion", IEEE Trans. on Circuits and Systems-II: Analog and Digital Signal Processing, Vol. 47, no. 3, pp.229-232, March 2000.

[6] R.J.Baker, "CMOS Mixed-Signal Circuit Design", IEEE Press, ISBN 0-471-22754-4, pp.368-372.