CALIBRATION-FREE HIGH-RESOLUTION LOW-POWER ALGORITHMIC AND PIPELINED AD CONVERSION

Patrick J. Quinn[1]
Maxim P. Pribytko[1]
Arthur H. M. van Roermund[2]

(1) Xilinx, IC design, Citywest Business Campus,
Logic Drive, Saggart, Dublin, Ireland
(2) Department of Electrical Engineering, Eindhoven University of Technology,
5600 MB Eindhoven, The Netherlands

Abstract

A novel implementation for algorithmic and pipelined ADCs is presented in this paper. A floating voltage hold buffer is proposed which enables the accurate addition of signal voltages without requiring precisely matching and linear components. A new 1.5-bit stage is presented based on the floating hold buffer in which voltage multiplication is replaced by voltage addition. An experimental 12-bit 3.3 MS/s algorithmic ADC in 0.25μm standard CMOS for a 2V application is described. It occupies 0.15mm^2 of die area and dissipates 5.5mW. The power and area FOMs are well below those previously reported for 1.5-bit algorithmic ADC stages.

J. H. Huijsing et al. (eds.), Analog Circuit Design, 327 -349 .

1. Introduction

There is an increasing trend to embed Analog-to-Digital Converters (ADCs) with the digital CMOS VLSI for applications requiring medium to high resolutions (10-14-bits) at sample frequencies up to a few MHz. This reduces cost, board space and board complexity, pin count and overall power consumption. For instance, such ADCs have become a ubiquitous peripheral in micro-controllers for servo applications, touch screens, measurement of supplies and die temperatures, etc.

The algorithmic or cyclic ADC [1],[2] is an excellent choice of architecture for applications where die area and power consumption are at a premium. The pipelined ADC [3],[4] is an excellent choice for applications demanding high speed. The core of both the algorithmic and the pipelined ADC is the highly efficient and robust 1.5-bit stage. It can be easily scaled in area and power down the pipeline for improved efficiency in pipelined converters [5]. The existing basic switched-capacitor (SC) charge-transfer circuit architecture for the implementation of the 1.5-bit ADC stage hasn't changed much in recent years [4],[6],[7]. Instead much attention is being spent on improving calibration routines for the cancellation of the effects of capacitor mismatches [8],[9],[18],[29]. Capacitor matching is difficult to control and usually limits the resolution of the 1.5-bit stage to below 10-bits in standard CMOS processes [14].

This paper proposes a novel circuit architecture for implementing the 1.5-bit ADC stage required for algorithmic and pipelined ADCs. The voltage multiplication operations needed for such ADCs are implemented by replacing multiplication by addition. Simple techniques are presented for achieving high accuracy addition without using precision components. The resolution for a given power dissipation and sample rate can thus be significantly increased.

1.1. ADC Efficiency

Area and power figures of merit, i.e. FOM_{area} and FOM_{power}, are useful measures of the relative performance of ADCs because they compare objectively the efficiency of different design solutions. They are defined as:

$$FOM_{area} = \frac{A}{2^{ENOB} \times f_s} \quad nm^2 / \text{conversion/Hz} \tag{1}$$

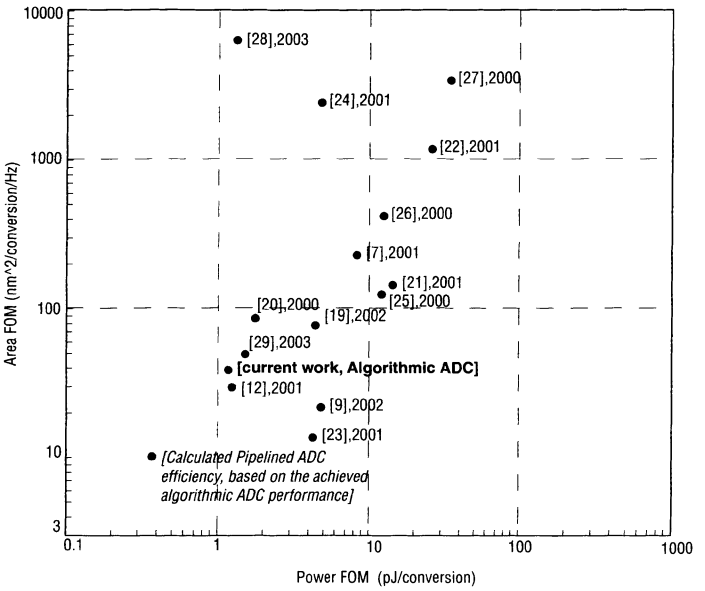


Fig. 1. Performance comparison of ADCs based on publications from 2000 onwards.

and

$$FOM_{power} = \frac{P}{2^{ENOB} \times f_s} \quad \text{pJ/conversion} \tag{2}$$

where *ENOB* is the effective number of bits and f_s is the sample frequency. These *FOM*s normalize area and power into silicon area use and energy use per ADC conversion. They can be used to compare all types of converters irrespective of architecture, frequency of operation or type and generation of process used. There are three primary reasons for the continual improvement in efficiency of ADCs, namely shrinking technology size, improving design techniques, and novel design solutions. A comparison of recently published pipelined ADCs (year 2000 onwards) for similar technologies is shown in Fig. 1. The area and power *FOM*s of the current work - 12-bit algorithmic ADC - are also depicted in the same diagram. FOM_{area} and FOM_{power} of the corresponding pipelined ADC, when calculated based on standard stage-to-stage scaling $\left\{1:\frac{1}{2}:\frac{1}{4}:\frac{1}{8}:\frac{1}{8}:\ldots\right\}$ [5], are each improved by a factor of 4. Indeed, the sample frequency increases by a factor of 12 while the area and power

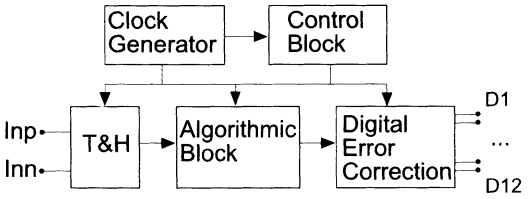


Fig. 2. Algorithmic ADC block(s) diagram

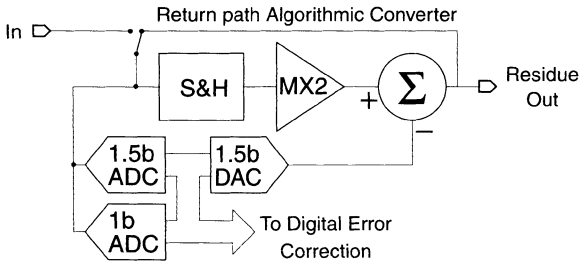


Fig. 3. 1.5-bit Algorithmic Block conversion stage.

each increase by a factor of 3 (1), (2). This too is depicted in Fig. 1. It can be seen that this solution achieves the lowest area and power *FOM*s.

1.2. The ADC Algorithm

Any continuous analog voltage V_a can be approximately represented in a n-bit binary form according to the following recursive algorithm [1]:

$$V_{i+1} = 2 \cdot V_i - D_i \cdot V_{ref},$$
$$D_i = \begin{cases} 1, & V_i > 0 \\ -1, & \text{otherwise}, \end{cases} \tag{3}$$
$$i = 1, 2, \ldots, n,$$

where $V_1 = V_a$, V_{ref} determines the resolvable input signal range $V_a \in (+V_{ref}, -V_{ref})$ and D_i, $i = 1,2,\ldots,n$, is mapped onto b_i, $i = 1,2,\ldots,n$, to give the binary representation of V_a: $b_1 b_2 \ldots b_n$. Voltage V_{i+1}, $i = 1,2,\ldots,n$, in (3) is referred to as an analog residue voltage for the i-th iteration.

An analog-to-digital converter based on (3) is called an algorithmic (or cyclic) ADC if the recursiveness of (3) is mapped on to a single piece of hardware, and a pipelined ADC if the recursive algorithm is mapped to a pipelined cascade of hardware stages. A practical algorithmic ADC employing a SC circuit was first presented in [2]. In this ADC, a charge transfer technique is used to implement the arithmetic in (3) and the ADC resolution is limited by capacitor mismatch.

The charge transfer technique [2] is improved on in [6], with further circuit efficiencies presented in [10]. Digital error correction, DEC, for the removal of offsets in operational transconductance amplifiers (OTAs) and comparators, originally proposed in [11], has become an essential technique for modern algorithmic and pipelined ADCs [9],[12], including the current work. The operation of the 1.5-bit DEC is represented analytically by:

$$V_{i+1} = 2 \cdot V_i - D_i \cdot V_{ref},$$

$$D_i = \begin{cases} 1, & V_i > \dfrac{V_{ref}}{4} \\ 0, & -\dfrac{V_{ref}}{4} \le V_i \le \dfrac{V_{ref}}{4} \\ -1, & -\dfrac{V_{ref}}{4} > V_i \end{cases} \qquad (4)$$

$$i = 1,2,\ldots,n.$$

A block diagram of the algorithmic or pipelined ADC is shown in Fig. 2. The Clock Generator provides non-overlapping clocks and the Control Block allows the ADC to be configured for synchronous or asynchronous sampling. The 1.5-bit DEC block accumulates D_i (4) to form a 12-bit output word. The purpose of the (single) algorithmic block in the case of the algorithmic converter, or the series of algorithmic blocks in the case of the pipelined converter is to perform the arithmetic of (4).

Fig. 3 is a detailed representation of the Algorithmic block. It shows a 1.5-bit (two-comparator) flash ADC, a 1-bit flash ADC for finalizing the A-to-D conversion, a

1.5-bit DAC capable of generating $\left(+V_{ref}, 0, -V_{ref}\right)$, a S/H (sample-and-hold) and MX2 (multiply-by-2) blocks. Note that Fig. 3 is an alternative representation of (4) and is common to most algorithmic ADCs. The impact of the non-ideal building blocks in Fig. 3 on the performance of the algorithmic ADC is explained in [13].

To emphasize the hardware pertaining to existing implementations, the residue transfer function is rewritten as

$$V_{out} = 2 \times V_{in} - 1 \times D \cdot V_{ref} \tag{5}$$

The multiplier factors (1 and 2) depend on capacitor ratios in existing charge transfer hardware realizations of this equation (eg. [14]). Charge is actively transferred from capacitor to capacitor via the virtual earth node of an OTA so that the accumulated charge on the feedback capacitor of the OTA produces V_{out} (5). This method is limited by the inaccuracies of capacitor matching as well as any non-linearity of capacitors. The DAC output voltages $D \cdot V_{ref} \in \left\{+V_{ref}, 0, -V_{ref}\right\}$ can be produced relatively accurately since it is sufficient to switch polarities between $+V_{ref}$ and $-V_{ref}$ or short to 0 in a differential realization.

In the method proposed here, the MX2 stage is implemented without using multiplication (i.e. charge transfer) but instead an accurate analog adder is employed. Furthermore, the DAC voltage is added without the need for charge transfer. In fact, formula (5) is rewritten as $V_{out} = V_{in} + V_{in} - D\,V_{ref}$. This will be presented next.

2. Realization Concepts and Building Blocks

2.1. Analog Addition

In this section, a concept is presented for implementing the arithmetic operations (4) of the ADC [1] in analog hardware without using multiplication. Analog multiplication, within the context of a SC circuit, is a series of voltage-to-charge and charge-to-voltage conversions - known also as the charge transfer technique (*QT*).

Typical concepts for creation of the MX2 (multiply-by-2) function are shown in Fig. 4(a), (c). Voltage multiplication by 2 occurs in Fig. 4(a) ($C \rightarrow 2C$) with the charge transfer of $Q = V_0 \cdot 2C$ of capacitor $C_1 = 2C$ to another capacitor $C_2 = C$, initially discharged, via a virtual earth node to become $2V_0$. This method is sensitive

to the ratio of capacitors C_1/C_2. An adaptation of this method is shown in Fig. 4(c), which is called a charge transfer with flip-around ($C \rightarrow C$), where both $C_1 = C$ and $C_2 = C$ are pre-charged with V_0, after which the charge on C_1 is transferred to C_2. A half a bit of extra accuracy can be achieved with the $C \rightarrow C$ circuit compared to the $C \rightarrow 2C$ circuit but in practical applications no better than 9-10-bits accurate multiplication-by-2 is realistic nor has been shown possible using these methods in standard CMOS processes. Example implementations of these QT multiplication concepts are demonstrated in figures Fig. 4(b),(d), where the virtual earth node is created within the feedback loop of an OTA. The 1.5-bit DAC function is also created by multiplying the DAC voltage by a factor of 1 $(C_1 = C)/(C_2 = C)$ through charge transfer via the virtual earth node to the output capacitor C_2.

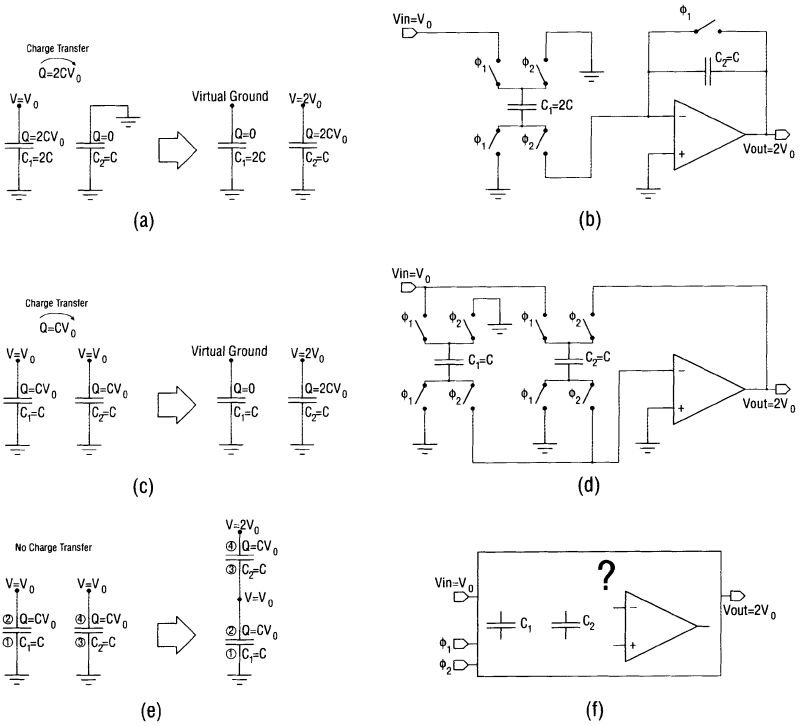


Fig. 4. Methods for creation of MX2 function with (a) charge transfer $C \rightarrow 2C$, with a realisation (b); (c) charge transfer and flip around $C \rightarrow C$, with a realisation (d);

An alternative concept ($C+C$), which is new for creation of the MX2 function, is based on voltage addition. Capacitors can be used for addition, as demonstrated in Fig. 4(e), where instead of multiplying V_0 by 2, voltage addition occurs by first charging each of C_1 and C_2 to V_0 and then placing these capacitors end to end, with say the plate (3) of C_2 connected to plate (2) of C_1. Furthermore, to fulfill the 1.5-bit DAC function, the output can be easily level shifted by adding a voltage at plate (1) of C_1.

When realized in a CMOS process, both terminals of C_1, C_2 have parasitic coupling capacitance. This parasitic capacitance results in an inaccurate MX2 function. While the $C+C$ concept of Fig. 4(e) is simple, its accurate implementation is difficult - Fig. 4(f). In fact, direct implementation of the $C+C$ concept is only suitable for low resolution ADCs of less than 8-bits.

2.2. Floating Hold Buffer for Accurate Analog Addition

In order to accurately implement the voltage addition concept of Fig. 4(e), a floating hold buffer is proposed with one input and one output, where the output voltage is held relatively constant and kept insensitive to the input voltage (i.e. of the form $V_{out} = V_{hold} + V_{in}$).

Usually a single capacitor (C) is used to sample and hold a voltage - see, for example, Fig. 5(a),(b). Assume that C is used as a floating hold capacitor so that the only way for charge to escape from its top plate terminal is through the parasitic capacitor C_{par} associated with the top plate - Fig. 5(c). The voltage across C in the hold mode is then given by

$$V_{hold} = V_{hold_0} + V_A \cdot \frac{C_{par}}{C}, \tag{6}$$

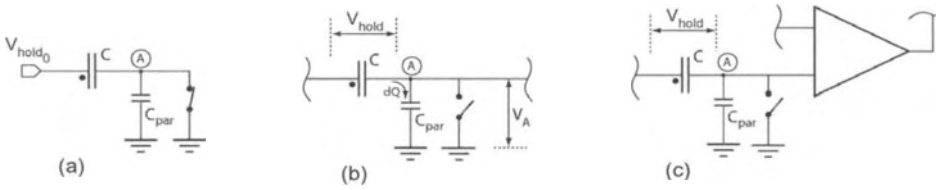


Fig. 5. (a) Sampling capacitor during sample phase, (b) sampling capacitor within hold circuitry, and (c) its connection to high impedance input of active element during hold phase.

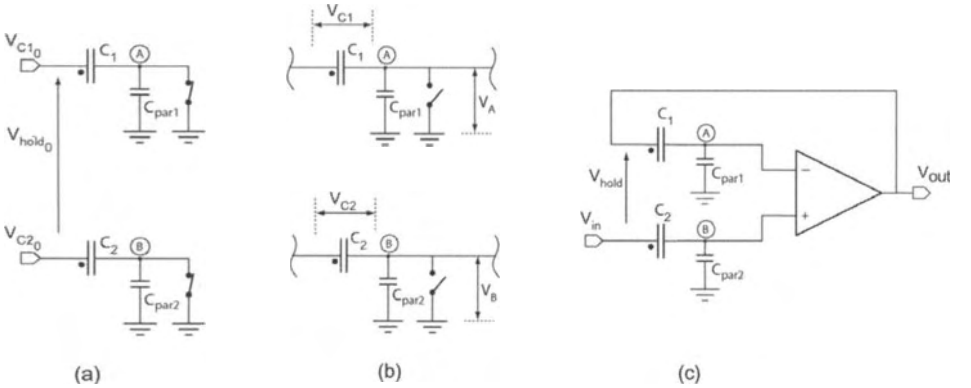


Fig. 6. (a) C_1 and C_2 used to sample a voltage, (b) hold phase, (c) proposed implementation using amplifier.

where V_{hold_0} is the initial correct value of the hold voltage at the sample moment and V_A is a floating voltage at node A. According to (6), V_{hold} depends on V_A. This limits the quality of the floating voltage hold circuitry employing a single sampling capacitor C.

Consider now two such capacitors used to sample V_{C1_0}, V_{C2_0} - Fig. 6(a). The equations for the voltages across each of C_1, C_2 in the hold mode, Fig. 6(b), are

$$V_{C1} = V_{C1_0} + V_A \cdot \frac{C_{par_1}}{C_1} \qquad V_{C2} = V_{C2_0} + V_B \cdot \frac{C_{par_2}}{C_2} \tag{7}$$

If we now set the condition in the hold mode that the voltages at the capacitor top plates are held equal, i.e. $V_A = V_B$, then the voltage held across both capacitors of Fig. 6(b) is given by

$$V_{C1} - V_{C2} = \left(V_{C1_0} - V_{C2_0}\right) + V_A \cdot \left(\frac{C_{par_1}}{C_1} - \frac{C_{par_2}}{C_2}\right). \tag{8}$$

Defining the voltage held as $V_{hold} = V_{C1} - V_{C2}$, with $V_{hold_0} = V_{C1_0} - V_{C2_0}$ the initial correct hold voltage at the sample moment, equation (8) can be re-written as

$$V_{hold} = V_{hold_0} + V_A \cdot \left(\frac{C_{par_1}}{C_1} - \frac{C_{par_2}}{C_2}\right). \tag{9}$$

Since $\left|\frac{C_{par1}}{C_1} - \frac{C_{par2}}{C_2}\right| \ll \frac{C_{par}}{C}$, then hold voltage (9) is significantly less sensitive to V_A when compared to equation (6) for a single capacitor implementation. Recall that $V_A = V_B$, and the only path for the top plate charge to escape is through the parasitics. This implies that nodes A and B must be physically disconnected.

Fig. 6(c) shows the implementation of the floating hold circuitry employing C_1, C_2 and a single-ended OTA. The OTA equalizes the voltages at the top plates of C_1 and C_2 by means of the negative feedback loop through C_1, thus satisfying $V_A = V_B$ of (9). V_{in} drives the bottom plate of C_2 causing a corresponding change in node voltage B at the OTA positive input. The hold voltage V_{hold} defined across C_1 and C_2 is insensitive to the voltage at the input terminal V_{in}. For each capacitor C_1 and C_2, there is only parasitic charge displacement between the top plate and the parasitic coupling capacitance connected to the top plate at the OTA input side - this is similar to a delta-charge redistribution technique described in reference [15].

This high accuracy floating hold buffer operates such that the output is unity gain buffered from the input while maintaining a fixed pre-determined hold voltage between output and input. This is the fundamental building block needed for accurate voltage addition.

2.3. Implementation of Single-ended and Differential MX2, Σ, DAC

This section develops an application of the $C + C$ concept to a 1.5-bit algorithmic ADC stage [30]. Recall that all the arithmetic required to carry out equation (4) can be done with analog voltage addition. Since the resolution of the algorithmic ADC is limited by the precision with which (4) can be implemented, the accurate floating buffer of section 2.2. is employed.

An input voltage is sampled using two pairs of capacitors C_{1_a}, C_{2_a} and C_{1_b}, C_{2_b} - see Fig. 7. Each of these pairs of capacitors is used to create the feedback of a floating hold buffer of Fig. 6(c). A single-ended implementation of the 1.5-bit ADC stage is shown in Fig. 7(b). Two floating hold buffers connected in series double the sampled voltage. This function creates the MX2, while the toggle switch in Fig. 7(b) represents the connection to the 1.5-bit DAC and hence this fulfills the implementation of the arithmetic of (4). Note that if each floating hold buffer has the same single-pole settling, then the settling time of both buffers in series increases by just one time con-

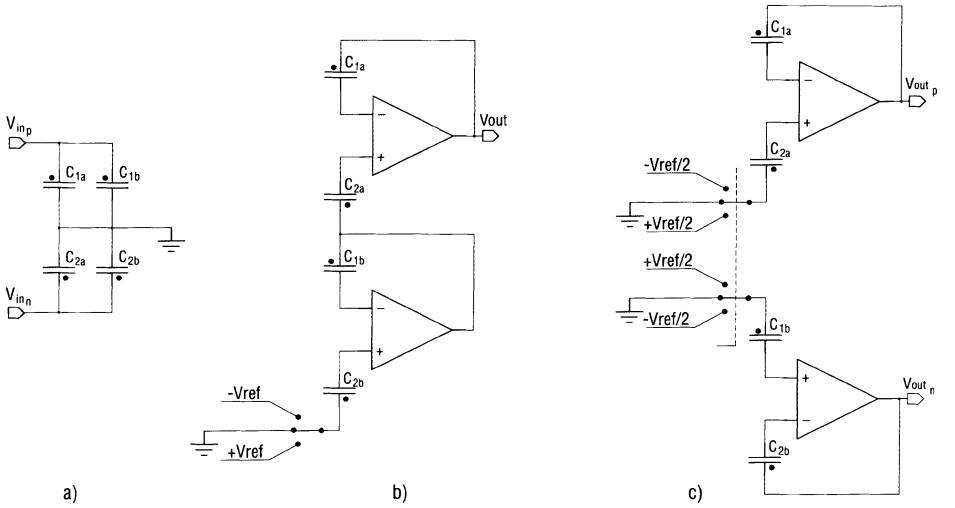


Fig. 7. Proposed C + C MX2, Summer and 1.5-bit DAC level shift, (a) pre-charge phase, (b) single-ended, (c) differential.

stant. For a 12-bit ADC, this means a 12% reduction in speed compared to a single buffer.

The differential configuration - defined in [16] - assumes that any signal propagates through a pair of traces with identical impedance and coupling to surrounding components and signals. The differential configuration is preferred to the single-ended because, for example, of improved noise immunity and linearity. A differential implementation of the $C+C$ concept is shown in Fig. 7(c). Note that in order to obtain a differential architecture, a series connection of floating hold buffers is replaced by a star connection. One floating buffer with C_{1_a}, C_{2_a} is used to obtain an accurate buffered version of the input, while the other floating buffer with C_{1_b}, C_{2_b} creates an accurate inversion so that the output voltage is double that of the input. The connection to the differential 1.5-bit DAC is again represented by the two toggle switches. This is then the differential implementation of (4).

Another feature of the differential 1.5-bit ADC stage proposed in Fig. 7(c) is that it is inherently common-mode stable, unlike conventional differential ADCs employing charge transfer techniques which require common-mode feedback to achieve stability. Furthermore, because the sampling technique in Fig. 7(a) is well suited to both

differential and single-ended sampling, and because the circuits in Fig. 7(b), (c) are suitable for driving single-ended and differential lines, respectively, it is easy to build very accurate single-ended to differential and differential to single-ended conversion using the proposed design solutions.

3. Practical Performance Issues

The maximum resolution of a practical switched capacitor 1.5-bit ADC stage is limited primarily by capacitor mismatch and less so by finite amplifier gain and charge feedthrough from the switches. For the purpose of comparison of the commonly used $C \rightarrow C$ and proposed $C+C$ 1.5-bit stages, similar capacitors are assumed to be used in both cases to sample the differential input signal, Fig. 8(a).

Consider first the $C \rightarrow C$ circuit in hold mode, Fig. 8(b), with all relevant parasitic capacitances included. The transfer function, including capacitor mismatch, is

$$V_{out_{C\rightarrow C}} = V_{in} \cdot 2 \cdot \left(1 + \frac{1}{2} \cdot \frac{\Delta C}{C}\right) - D \cdot V_{ref} \cdot \left(1 + \frac{\Delta C}{C}\right) \tag{10}$$

where the relative signal capacitor mismatch is given by $\Delta C/C = \frac{1}{2} \cdot \left(\Delta C_1/C_1 + \Delta C_2/C_2\right)$. Here, it is assumed $\Delta C \ll C$. Finite amplifier gain deteriorates the transfer function further giving rise to a total fractional transfer gain error of

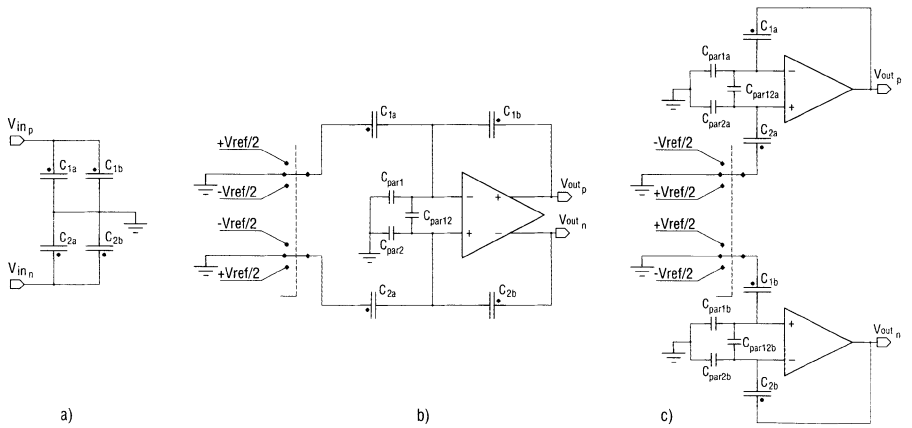


Fig. 8. Comparision of $C \rightarrow C$ and $C+C$ circuit architectures including parasitic capacitors.

$$\varepsilon_{C\to C} = \frac{1}{2}\cdot\frac{\Delta C}{C} - \frac{1}{A_0\cdot\beta_{C\to C}} \tag{11}$$

where the feedback factor $\beta_{C\to C}$ is given by

$$\beta_{C\to C} = \frac{C}{2C + 2C_{par_{12}} + C_{par1}}. \tag{12}$$

Due to the strong influence of capacitor mismatch on the $C \to C$ 1.5-bit stage, no better than 0.1% gain error can be expected so that some form of calibration or trimming must be used for an ADC accuracy of greater than 10-bits. Calibration generally adds latency to the algorithmic ADC and increases area and power consumption not only as a result of the added calibration circuitry but because of the extra parasitic loading on analog circuit nodes and the consequent increase in amplifier power.

The voltage transfer function of the $C + C$ 1.5-bit stage is

$$V_{out_{C+C}} = V_{in}\left(1+\left(\frac{C_2}{C_2+C_{par_2}}\right)\left(\frac{C_1+C_{par_1}}{C_1}\right)\right) - DV_{ref}\left(\frac{C_2}{C_2+C_{par_2}}\right)\left(\frac{C_1+C_{par_1}}{C_1}\right) \tag{13}$$

With $\Delta C_{par}/C_{par} = \frac{1}{2}\cdot\left(\Delta C_{par1}/C_{par1} + \Delta C_{par2}/C_{par2}\right)$ as the relative mismatch of the equivalent input parasitic capacitors C_{par_1} and C_{par_2}, equation (13) can be rewritten as

$$V_{out_{C+C}} = 2V_{in}\left(1+\frac{C_{par}}{2C}\left(\frac{\Delta C_{par}}{C_{par}}+\frac{\Delta C}{C}\right)\right) - DV_{ref}\left(1+\frac{C_{par}}{C}\left(\frac{\Delta C_{par}}{C_{par}}+\frac{\Delta C}{C}\right)\right) \tag{14}$$

A small transfer error is produced proportional to C_{par}/C times the mismatch of the signal and parasitic capacitors. Note that the extra term of $\Delta C_{par}/C_{par}$ must be accounted for when calculating the effect of mismatch. After inclusion of the effect of the amplifier, the total gain error for the $C + C$ stage becomes

$$\varepsilon_{C+C} = \frac{1}{2}\left(\frac{C_{par}}{C}\right)\left(\frac{\Delta C_{par}}{C_{par}}+\frac{\Delta C}{C}\right) - \frac{1}{\beta_{C+C}}\left(\frac{1}{A_0}+\frac{1}{2\cdot CMRR}\right) \tag{15}$$

with the feedback factor being given by:

$$\beta_{C+C} = \frac{C}{C + 2C_{par_{12}} + C_{par1}}. \tag{16}$$

For an N-bit converter having at least 0.5-bit accuracy, it is required that $\varepsilon_{C+C} < V_{ref}/2^{N-1}$.

Standard OTA design techniques [17] allow the attainment of very high A_0 and CMRR above 90dB which is sufficient for at least 15-bit accuracy. Indeed the DC CMRR can be designed to be of the same order of magnitude as A_0 as explained in section 4. It is the *capacitor matching* that is the bottle neck to achieving higher than 10 bits accuracy in charge transfer based 1.5-bit stage configurations, such as the $C \rightarrow C$ method. Comparing formulae (11) and (15), the capacitor mismatch error of the $C+C$ method is reduced by $1\Big/\frac{C_{par}}{C} \times \left(1+\frac{\Delta C_{par}/C_{par}}{\Delta C/C}\right)$ compared to the $C \rightarrow C$ method. The relative capacitor mismatch errors are given by $\Delta C/C \propto 1/\sqrt{C}$ and $\Delta C_{par}/C_{par} \propto 1/\sqrt{C_{par}}$. From section 4., a C/C_{par} ratio of 35 has been obtained for this design. The matching of the parasitic capacitors (metal-metal and gate oxide) are expected to be at least as good as the signal capacitors (metal-metal only), so that a conservative estimate for the improvement in capacitor mismatch error of the $C+C$ method compared to the $C \rightarrow C$ method is $\boldsymbol{C/2C_{par}}$. This implies an extra 4 bits accuracy (linearity) is possible over previous 1.5-bit conversion stages. Since the capacitors determine the efficiency of the ADC, it is possible to create a highly efficient architecture using the proposed $C+C$ method giving rise to low power and silicon area. A further benefit is that the $C+C$ architecture is relatively insensitive to capacitor non-linearity so that small area gate oxide capacitors could also be used.

The feedback factor of the $C+C$ stage is almost double that of the $C \rightarrow C$. The benefits of a larger feedback factor are to be found in improved amplifier gain, settling speed and noise. Switch charge feedthrough is largely cancelled at the input of

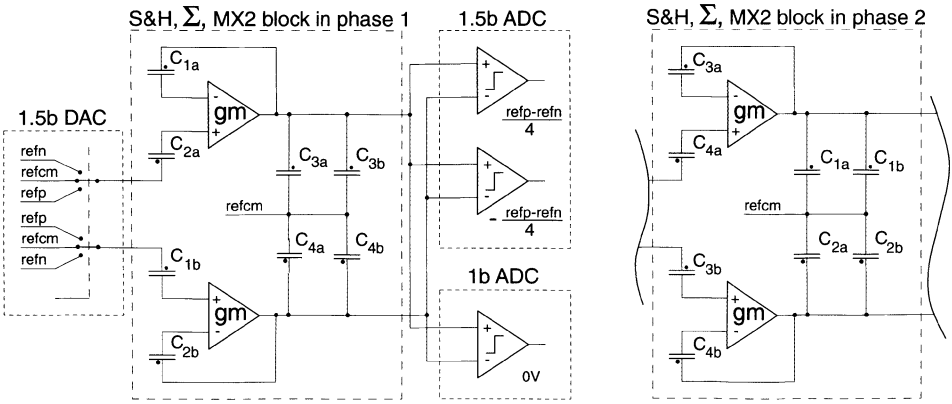


Fig. 9. Implementation of C + C ADC 1.5-bit conversion stage.

each OTA due to the differential switching scheme and the appropriate use of early clocking ensures that the residue output of the $C+C$ 1.5-bit stage is cleanly sampled before the switches connected around the OTA change phase.

4. 1.5-bit Per Stage Circuit Implementation

The merged implementation of the 1.5-bit conversion stage of Fig. 3, comprising the S&H, Σ, MX2 and 1.5-bit DAC (implemented recursively in this work as an algorithmic converter) is demonstrated in Fig. 9 according to the method proposed in section 2. The ADC using the $C+C$ technique is fully differential with the signal range given by $V_{ref} = (\text{RefP-RefN})$. Assuming initially that the DAC output is at RefCM, i.e. differential 0V, then on a given cycle of the ADC, say *phase* 1, differential output voltage $V_{out} = V_0$ is present across the series combination of C_{3_a} and C_{4_a} in parallel with the series combination of C_{3_b} and C_{4_b}. On the following cycle, *phase* 2, V_0 is applied across the top OTA with C_{3_a} between the top OTA output and its negative input and C_{4_a} placed between the OTA positive input and the top output terminal of the DAC. Similarly, across capacitors C_{3_b} and C_{4_b}, a further V_0 is available between the bottom output of the DAC and the output of the bottom OTA. The net effect after *phase* 2 is that a voltage of $V_{out} = 2 \times V_0$ is present between the output

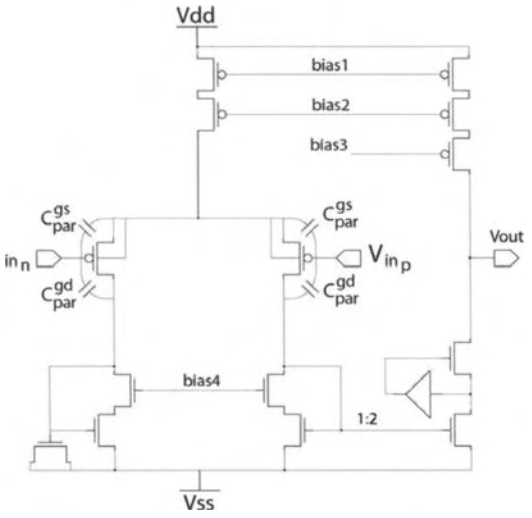


Fig. 10. Current mirror amplifier with gain boosting.

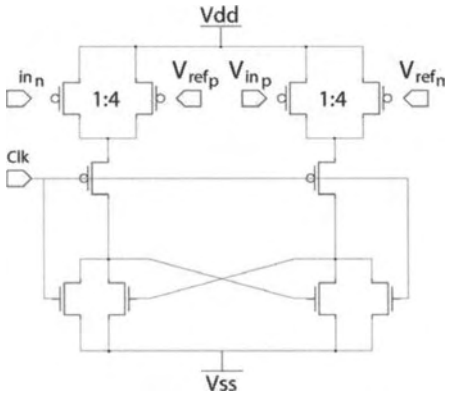


Fig. 11. Dynamic comparator.

terminals of the OTAs. If the DAC voltage changes to either RefP or RefN, then a voltage of either $+V_{ref}$ or $-V_{ref}$ is added to V_{out}.

The 8 ADC sample capacitors are each 2pF and composed of inter-digitated metal-metal capacitors. Metal layers M1, M2, M3 form the main active capacitance, being laid out in thin strips to create inter-digitated bottom and top plates. N-well, Poly and M4 are used for shielding.

The circuit is designed to work from a 2V supply. Single-ended current mirror OTAs are used with gain-boosted NMOS cascoded output stage [31] - see Fig. 10 for simplified version. Both the output stage PMOS transistors and input stage PMOS current source are have the same output resistance. With g_m the transconductance of the PMOS input stage and $r_{out_{PMOS}}$ the output resistance of either the output stage or input stage current sources, A_0 and CMRR are each given approximately by $g_m \cdot r_{out_{PMOS}}$. This was simulated to be at least 70dB (sufficient for our 12-bit embedded application). The parasitic input capacitance C_{par} of section 3. is given approximately by the gate-drain overlap capacitance of one transistor (36fF) of the input differential pair together with the wiring capacitance at its gate (20fF). Hence, $C/2C_{par}$ is given by $2/2 \times 0.056 \approx 17$ which equates to about 4 bits improved accuracy (linearity) over previous charge transfer approaches. Dynamic comparators were used - see Fig. 11 - with PMOS input stages to enable fast switching for low level input voltages. In this design, RefP and RefN were created as 1.4V and 0.4V.

The circuit diagram of Fig. 12 shows the switches and clock phases used to switch the capacitors referred to in Fig. 9. The start of a 12-bit conversion, ADC cycle 1, is

determined by *aclk*, at which time the output of the T&H is sampled across C_{1a} and C_{2a} in series and also C_{1b} and C_{2b} in series. The T&H signal is also fed to the 1.5-bit sub-ADC to determine the MSB. On cycle 2, the T&H signal is multiplied by 2 and the DAC output is subtracted off. This is then sampled across C_{3a} and C_{4a} in series and C_{3b} and C_{4b}. It is also fed to the 1.5-bit ADC to determine the MSB-1. For the 11

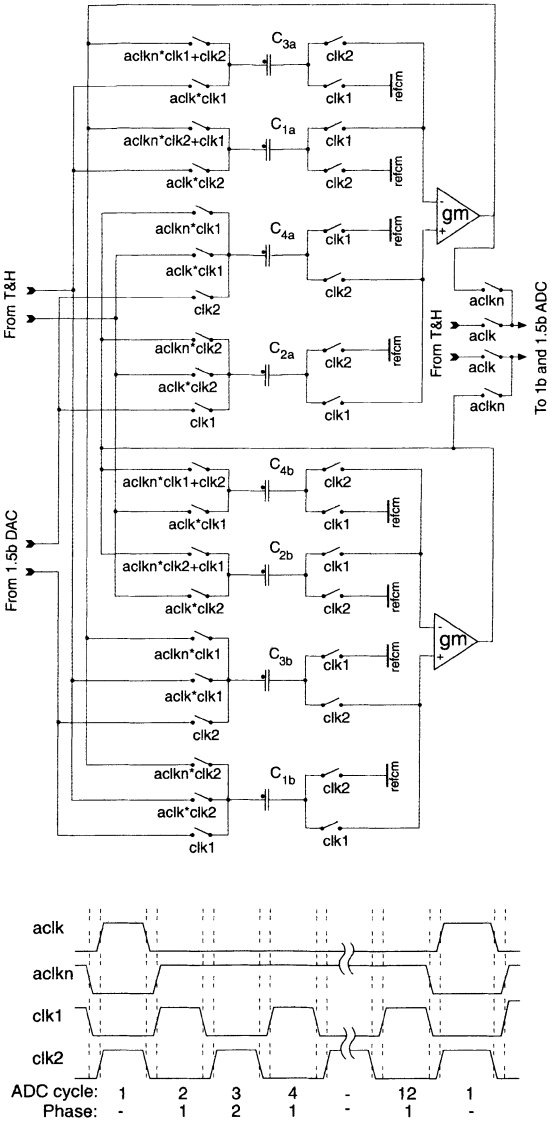


Fig. 12. Schematic of S&H, Summer and MX2 block.

cycles following acquisition of the T&H signal, *aclk* is low and *aclkn* is high, so that the differential output of the OTAs is used as input for each cycle of the ADC. The circuit works in a double sampled mode with clocks *clk*1 and *clk*2. There is no latency, where 12 conversions of the ADC produce a series of 12 bits from MSB to LSB.

5. Measurement Results

The ADC was prototyped in standard 0.25μm CMOS. A picture of the die showing the partitioning of the blocks can be seen in Fig. 13. Detailed measurements were carried out on 40 samples. The main performance parameters are presented in Table 1. The complete ADC with DEC, clocks and comparators occupies 0.15mm^2. The total power consumption including digital circuitry is 5.5 mW. Typical DNL and INL plots are presented in Fig. 14. DNL of less than 0.25 LSBs and INL of less than 0.8LSBs at 12-bit level is achieved. The THD and SNR as a function of the sampling frequency is represented in Fig. 15(a), while a typical FFT spectral density plot is shown in Fig. 15(b) for a 200kHz 1 Vpp input signal. The THD at 1MS/s was measured to be 77dB. The SNR was 64.5dB which is lower than expected. This was attributed to the layout where there was coupling from the digital output buffers back into the analog supply. It is expected that a new version of the ADC, presently in diffusion, will alleviate this problem and have improved conversion efficiency.

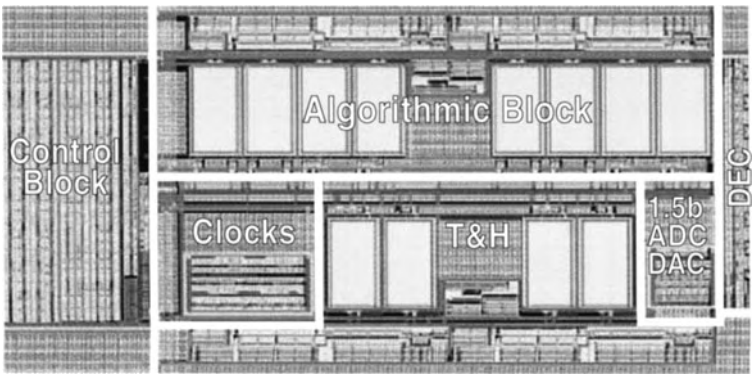
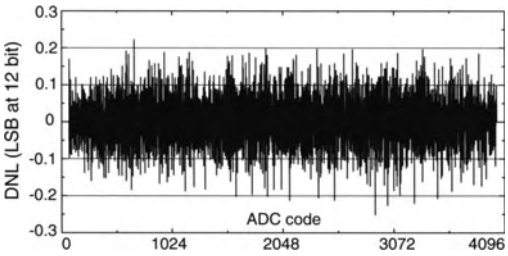


Fig. 13. Die photograph showing partitioning of blocks.

(a)



(b)

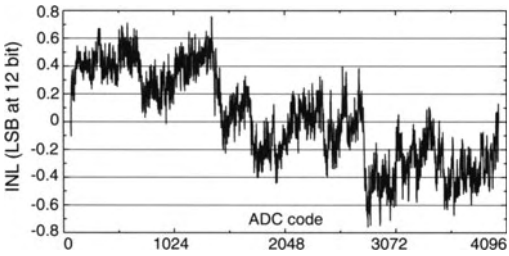
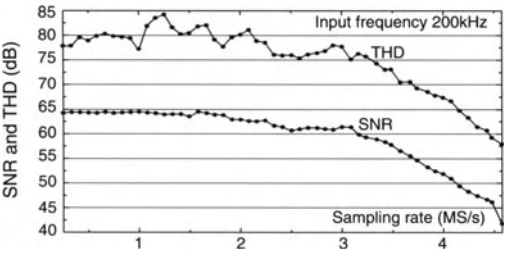


Fig. 14. Measured static ADC linearity at 12-bit level with (a) DNL and (b) INL.

(a)



(b)

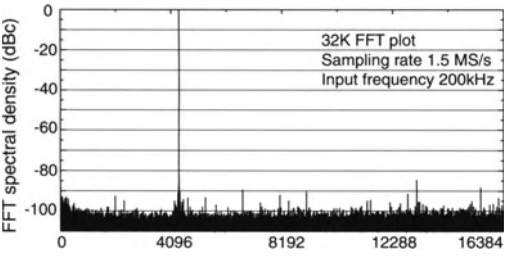


Fig. 15. Measured ADC dynamic performance.

6. Conclusions

A novel implementation of the 1.5-bit ADC stage for use in both algorithmic and pipelined ADCs has been presented. Voltage multiplication has been replaced by accurate addition and a *floating hold buffer* has been proposed for its implementation. The accuracy of the ADC employing the new circuitry is less sensitive to the matching accuracy and linearity of the sampling capacitors compared to previous circuit techniques presented in the literature.

An analytical equation has been obtained of the transfer error of the proposed algorithmic stage and it is compared with the analytical equation of the traditional charge-transfer algorithmic stage. With C the value of the sampling capacitors and C_{par} the parasitic capacitance at each differential amplifier input terminal, the proposed algorithmic stage has its sensitivity to capacitor mismatch reduced by a factor of $C/2C_{par}$ giving a corresponding improvement in linearity. Ensuring that $C_{par} \ll C$ suggests a high resolution ADC stage (12-bits and beyond) can be practically realized without the need for calibration.

Within the die area allocated to this work for an embedded application, 12-bit accuracy has been achieved at 3.3 MS/s using uncharacterized metal-metal capacitors and neither trimming nor calibration was needed. The ADC is suitable for embedding in CMOS digital VLSI and it will scale easily with the process without the need for special analog characterization.

Table 1: Measurements

Technology	250nm standard industrial CMOS
Resolution	12 bits
Conversion Rate	3.3 MS/s with 40 MHz clock
Clock cycles for 12 bits	12
Active area	0.15 mm^2
Power	5.5 mW
DNL at 12b	< 0.25 LSBs
INL at 12b	< 0.8 LSBs
THD at 1 MSamples/s	77 dB
SNR at 1 MSamples/s	64.5dB
SFDR at 1 MSamples/s	80dB
Power FOM	1.2 pJ/conversion
Area FOM	31 nm^2/conversion

References

[1] H. Schmid, *Electronic Analog/Digital Conversions*, Van Nostrand-Reinhold, p.195, 1970.

[2] Robert H. McCharles, Vikram A. Saletore, William C. Black, David A. Hodges ''An Algorithmic Analog-to-Digital Converter'', IEEE ISSCC, SECTION IX, pp. 96-97, 1977.

[3] Y. Ren, B. H. Leung, Y-M. Lin, "A Mismatch-Independent DNL Pipelined Analog-to-Digital Converter," *IEEE Trans. Circuits and Systems II: Proc. Analog and Digital Processing*, vol. 46, no. 5, pp. 517-526, May, 1999.

[4] Thomas B. Cho and Paul R. Gray, "A 10 b, 20 Msample/s, 35mW Pipeline A/D Converter," *IEEE J. Solid-State Circuits*, vol. 30, no. 3, pp. 166-172, Mar, 1995.

[5] David W. Cline and Paul 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. 31, no. 3, pp. 294-303, Mar, 1996.

[6] Bang-Sup Song, Michael F. Tompsett, Kadaba R. Lakshmikumar ''A 12-bit 1-Msample/s Capacitor Error-Averaging Pipelined A/D Converter.''*IEEE J. Solid-State Circuits*, vol. 23, no. 6, pp. 1324-1332, Dec, 1988.

[7] Lauri Sumenen, Mikko Waltari, Kari A. I. Halonen''A 10-b 200-MS/s CMOS Parallel Pipeline A/D Converter''*IEEE J. Solid-State Circuits*, vol. 36, no. 7, pp. 1048-1055, July 2001.

[8] Hae-Seung Lee, "A 12-b 600 ks/s Digitally Self-Calibrated Pipelined Algorithmic ADC," I*EEE J. Solid State Circuits*, vol. 29, no. 4, pp. 509-515, Apr. 1994.

[9] Shang-Yuan Chuang, Terry L. Sculley ''A Digitally Self-Calibrating 14-bit 10-MHz CMOS Pipelined A/D Converter''*IEEE J. Solid-State Circuits*, vol. 37, no. 6, pp. 674-683, June 2002.

[10] Krishnaswami Nagaraj ''Efficient Circuit Configurations for Algorithmic Analog to Digital Converters'', *IEEE Trans. Circuits and Systems II: Proc. Analog and Digital Processing*, Vol.40, No.12, pp.777-785, Dec. 1993.

[11] Stephen H. Lewis and Paul Gray, ''A Pipelined 5-Msample/s 9-bit Analog-to-Digital Converter.'' *IEEE J. Solid-State Circuits*, vol. SC-22, no. 6, pp. 954-961, Dec. 1987.

[12] W. Yang, D. Kelly, I. Mehr, M. Sayuk, L. Singer, "A 3-V 340-mW 14-b 75-Msample/s CMOS ADC With 85-dB SFDR at Nyquist Input", *IEEE J. Solid State Circuits*, vol. 36, no. 12, pp. 1931-1936, Dec, 2001.

[13] Stephen H. Lewis, "Optimizing the Stage Resolution in Pipelined, Multstage, Analog-to-Digital Converters for Video Rate Applications," *IEEE Trans. Circuits and Systems II: Proc. Analog and Digital Processing*, vol. 39, no. 8, pp. 516-523, Aug, 1992.

[14] Andrew M. Abo and Paul R. Gray, "A 1.5-V, 10-bit, 14.3 MS/s CMOS Pipeline Analog-to-Digital Converter," I*EEE J. Solid State Circuits*, vol. 34, no. 5, pp. 599-606, May, 1999.

[15] P. J. Quinn, K. van Hartingsveldt, A.H.M. van Roermund, "A 10.7-MHz CMOS SC Radio IF Filter Using Orthogonal Hardware Modulation," *IEEE J. Solid-State Circuits*, vol. 35, no. 12, pp. 1865-1876, Dec, 2000.

[16] Douglas Brooks, "Differential Signals, The Differential Difference!", *Printed Circuit Design*, CMP Publication, May 2001.

[17] Klaas Bult and Govert J. G. M. Geelen ''A Fast-Settling CMOS Op Amp for SC Circuits with 90-dB DC Gain''*IEEE J. Solid-State Circuits*, vol. 25, no. 6, pp. 1379-1384, Dec. 1990.

[18] O. Erdogan, P. J. Hurst, S. H. Lewis, "A 12-b Digital-Background-Calibrated Algorithmic ADC with -90-dB THD", *IEEE J. Solid State Circuits*, vol. 34, no. 12, pp. 1812-1820, Dec, 2001.

[19] Shafiq M. Jamal, Diahong Fu, Nick C.-J. Chang, Paul J. Hurst, Stephen H. Lewis ''A 10-b 120-Msample/s Time-Interleaved Analog-to-Digital Converter With Digital Background Calibration.''*IEEE J. Solid-State Circuits*, vol. 37, no. 12, pp. 1618-1627, Dec. 2002.

[20] Iuri Mehr and Larry Singer''A 55-mW, 10-bit, 40-MSamples/s Nyquist-Rate CMOS ADC''*IEEE J. Solid-State Circuits*, vol. 35, no. 3, pp. 318-324, March 2000.

[21] Myung-Jun Choe, Bang-Sup Song, Kantilal Barcania ''An 8-b 100-MSample/s CMOS Pipelined Folding ADC''*IEEE J. Solid-State Circuits*, vol. 36, no. 2, pp. 184-194, Feb. 2001.

[22] Jun Ming, Stephen Lewis ''An 8-bit 80-Msample/s Pipelined Analog-to-Digital Converter With Background Calibration''*IEEE J. Solid-State Circuits*, vol. 36, no. 10, pp. 1489-1497, Oct. 2001.

[23] Hendrik van der Ploeg, Gian Hoogzaad, Henk A. H. Termeer, Maarten Vertregt''A 2.5-V 12-b 54-Msample/s 0.25-um CMOS ADC in 1-mm^2 With Mixed-Signal Chopping and Calibration''*IEEE J. Solid-State Circuits*, vol. 36, no. 12, pp. 1859-1867, Dec. 2001.

[24] Ming-Huang Liu and Shen Iuan Liu ''An 8-bit 10MS/s Folding and Interpolating ADC Using the Continuous-Time Auto-Zero Technique''*IEEE J. Solid-State Circuits*, vol. 36, no. 1, pp. 122-128, Jan 2001.

[25] Myung-Jun Choe, Bang-Sup Song, Kantilal Barcania ''A 13-b 40-MSamples/s CMOS Pipelined Folding ADC with Background Offset Trimming''*IEEE J. Solid-State Circuits*, vol. 35, no. 12, pp. 1781-1790, Dec. 2000.

[26] Ion E. Opris, Bill C. Wong and Sing W. Chin ''A Pipeline A/D Converter Architecture with Low DNL''*IEEE J. Solid-State Circuits*, vol. 36, no. 2, pp. 281-285, Feb. 2000.

[27] Siamak Mortezapour and Edward Lee, "A 1-V, 8-bit Successive Approximation ADC in Standard CMOS Process," *IEEE J. Solid-State Circuits*, vol. 35, no. 4, pp. 642-646, Apr 2000.

[28] Michael Scott, Bernhard Boser and Kristofer Pister, "An Ultralow-Energy ADC for Smart Dust," *IEEE J. Solid-State Circuits*, vol. 38, no. 7, pp. 1123-1129, July 2003.

[29] Boris Murmann and Bernhard Boser, "A 12-bit 75-MS/s Pipelined ADC Using Open-Loop Residue Amplification," *IEEE J. Solid-State Circuits*, vol. 38, no. 12, pp. 2040-2049, Dec 2003.

[30] Patrick Quinn and Maxim Pribytko, "Capacitor Matching Insensitive 12-bit 3.3 MS/s Algorithmic ADC in 0.25μm CMOS," in *IEEE 2003 Proc. Custom Integrated Circuits Conf.*, San Jose, Sept. 2003, pp. 425-428.

[31] Maxim Pribytko and Patrick Quinn, "A CMOS Single-Ended OTA With High CMRR," in *Proc. 29th Eur. Solid-State Circuits Conf.*, Estoril, Portugal, Sept. 2003, pp. 293-296.