JOURNAL OF SEMICONDUCTOR TECHNOLOGY AND SCIENCE, VOL.17, NO.3, JUNE, 2017
https://doi.org/10.5573/JSTS.2017.17.3.387
ISSN(Print) 1598-1657
ISSN(Online) 2233-4866

Mismatch-tolerant Capacitor Array Structure for Junction-splitting SAR Analog-to-digital Conversion

Youngjoo Lee[1,*], Taehyoun Oh[2], and In-Cheol Park[3]

***Abstract*—A new junction-splitting based SAR ADC with a redundant searching capacitor array structure in 0.13 μm CMOS process to alleviate capacitor mismatch effects, is presented. The normalized average power has a factor of 0.35 to the conventional SAR ADC at 10-bit conversion accuracy. Statistical experiments show the number of missing codes resulting from the mismatch reduces by 95% for 3% unit-capacitor mismatch ratio, while keeping the conversion energy to that of the conventional JS capacitor array.**

***Index Terms*—Capacitor mismatch, low-power ADC, redundant ADC, SAR analog-to-digital convertor**

I. INTRODUCTION

Along with the flash analog-to-digital converter (ADC) [1, 2] and the pipelined ADC [3, 4], the successive approximation register (SAR) ADC is widely used for Nyquist-rate analog-to-digital conversion. The charge-redistribution SAR ADC [5] has been conventionally used in many applications due to its simple control scheme. However, the conventional method is not efficient in the viewpoint of energy consumption, since all the capacitors are involved in the charging and discharging process. To reduce the energy consumed in the capacitor array and control circuit, many papers have suggested energy-efficient capacitor array architectures and switching methods such as charge sharing [6], capacitor splitting [7, 8] and comparator splitting [9]. More advanced works taking advantage of differential inputs have been reported in [10, 11]. Among the various solutions, the junction-splitting (JS) capacitor array in [12] consists of multiple capacitor sections, eliminating power-hungry discharging operations. Based on the JS architecture, a two-step charge-sharing method [13] has been proposed to reduce the energy consumed in the capacitive DAC.

The linearity of SAR ADC is constrained by the mismatches among capacitor array during implementation. To improve the linearity of ADC, several error-correction algorithms have been reported. The hard-decision output was digitally corrected in [14], and a background digital calibration method was presented in [15]. To reduce the differential nonlinearity (DNL) error, a noise-shaping method based on fingered registers was proposed in [16], and a flexible comparator in [17]. However, the above methods necessitate additional correction blocks and complex control circuits. Recent analysis reveals that providing multiple redundant paths is effective in correcting such errors, and can be realized at the cost of a little additional complexity [18-20]. However, all the previous redundant search algorithms assume relatively small capacitor mismatches, which are valid for the conventional capacitor array. As the capacitor size of JS SAR ADC is incrementally increased to generate the next voltage to be compared, the most significant bit (MSB) is determined with the smallest capacitors and can be significantly affected by a small capacitor mismatch ratio. In addition,

Manuscript received Aug. 25, 2016; accepted Mar. 27, 2017
[1] Department of Electrical Engineering, Pohang University of Science and Technology (POSTECH), Pohang 37674, Korea
[2] Department of Electronic Engineering, Kwangwoon University, Seoul 01897, Korea
[3] Department of Electrical Engineering, Korea Advanced Institute of Science and Technology, Daejeon 34142, Korea
E-mail : youngjoo.lee@postech.ac.kr

straightforwardly applying the conventional redundant search to the JS capacitor array would increase the total capacitance remarkably, consuming much more energy than the original JS capacitor array does.

Compared to complicated capacitor sets in [21], this paper presents a simpler array structure and its control scheme to alleviate the capacitor mismatch effects. To recover errors made in earlier steps, we provide redundant search steps by modifying the capacitor array slightly. In addition to the capacitor mismatch effects, the redundant array is effective in improving the linearity influenced by several factors such as the parasitic input capacitance of the comparator and the nonlinear characteristics of switches. A statistical analysis is presented to minimize the overheads by deriving the number of redundant steps appropriate for a given unit-capacitor mismatch.

The rest of this paper is organized as follows: Section II briefly describes previous capacitor array architectures, and Section III presents the proposed redundant capacitor array and its control scheme. The statistical analysis on the number of redundant steps is presented in Section IV, and simulation results are summarized in Section V. Finally, concluding remarks are made in Section VI.

II. PREVIOUS CAPACITOR ARRAYS

Fig. 1 shows the block diagram of an n-bit SAR ADC, using a binary-weighted capacitor array for the digital-to-analog converter (DAC). Initially, all the capacitors in the array sample the input voltage, V_{in} at a specific timing by closing S_{sample}. In the conventional SAR algorithm, the digital code corresponding to the input voltage is found by conducting binary search. Given a previously determined DAC code, the capacitor array generates a voltage V_{out} to be fed to the comparator by subtracting V_{in} from the voltage corresponding to the code. As the SAR algorithm determines one bit at a time, n cycles are taken to generate the final n-bit digital code. The comparison result d_i determined at the i-th step is fed back to the SAR control logic to produce the next digital code. In the first step, all the bits of the digital code except the most significant bit (MSB) are set to zero to indicate the half reference voltage. In the second step, the next digital code is generated by changing the MSB according to the comparison result and setting the second MSB to one.

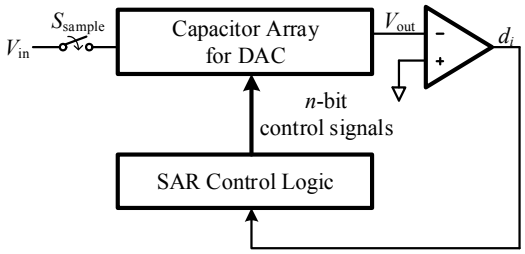

Fig. 1. Block diagram of SAR ADC.

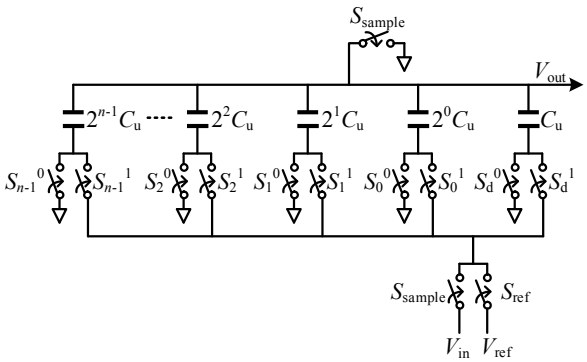

Fig. 2. Conventional DAC capacitor array.

This process continues until we reach the last bit of the digital code.

1. Conventional Capacitor Array Architecture

Fig. 2 shows a conventional DAC capacitor array [22], which consists of n capacitors and one dummy capacitor of capacitance C_{u}. At the i-th step, the output voltage is

$$V_{\text{out}} = -V_{\text{in}} + \frac{C_{\text{H}i}}{C_{\text{H}i} + C_{\text{L}i}} V_{\text{ref}}, \tag{1}$$

where $C_{\text{H}i}$ is the capacitance connected to the reference voltage (V_{ref}) and $C_{\text{L}i}$ is that connected to ground (GND) depending on the decision prior to i-th step. Note that $C_{\text{H}i} = \sum_k 2^k C_{\text{u}}$ for k such that S_k^1 is closed, and $C_{\text{L}i} = \sum_k 2^k C_{\text{u}}$ for k such that S_k^0 is closed. As S_{d}^0 is always closed during the redistribution mode, the dummy capacitor is always included in $C_{\text{L}i}$. In summary, V_{out} is dependent on the capacitance ratio of $C_{\text{H}i}$ to the total capacitance C_{tot} of $C_{\text{H}i} + C_{\text{L}i}$.

During the conversion, the bottom plate of a certain capacitor in the DAC capacitor array is connected to GND at first and then switched to V_{ref} forcefully in the

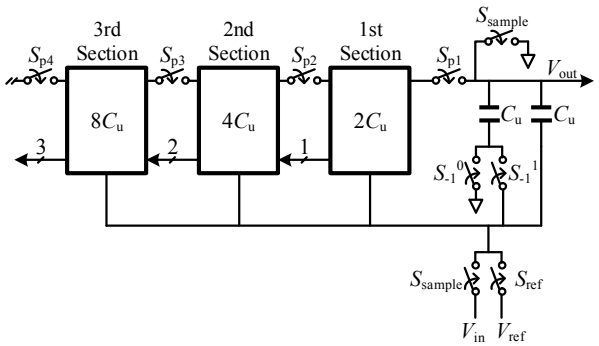

Fig. 3. Overall structure of the JS capacitor array.

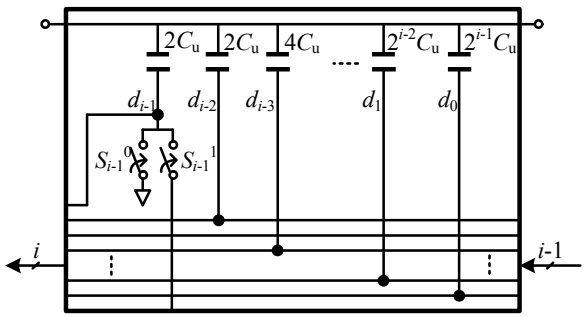

Fig. 4. Detailed structure of the i-th JS capacitor section.

next step. Depending on the previous decision, it can move back to GND in the following step as well. While this node is switching back and forth, it wastes energy. In the previous capacitor arrays including the conventional capacitor array [5] and the splitting capacitor array [7, 8], all the capacitors are involved in deriving V_{out} so that C_{tot} be always constant. Therefore, it is inevitable to switch some of the capacitors to make a new capacitance ratio needed for the next V_{out}.

2. Junction-splitting Capacitor Array

A low-power junction-splitting (JS) capacitor array has been presented in [12]. Fig. 3 shows the basic structure of the JS capacitor array. Each box represents a capacitor section of which structure is denoted in Fig. 4, and the number in a box of Fig. 3 is the total capacitance of the capacitor section. This capacitor array has additional serial switches, S_{pi} to decide whether or not connect the top plates of the capacitor sections. In contrast to the conventional capacitor array that uses all the capacitors and rearranges the switches connected to the bottom plates to make a desired V_{out}, the JS capacitor array makes a new V_{out} by appending a capacitor section

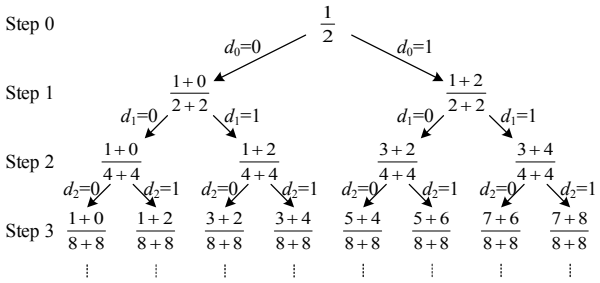

Fig. 5. Binary search tree for the JS capacitor array.

to the previous capacitor array. How the JS capacitor array works is illustrated in Fig. 5, where the denominator and numerator represent C_{tot} and C_{Hi}, respectively. First, the MSB, d_0, is determined by comparing the input voltage with $0.5V_{ref}$, which is achieved with the two smallest capacitors. Then, the next voltage to be compared is made by connecting the adjacent capacitor section, one at a time.

As shown in Fig. 5, the capacitance of the next section to be added to the denominator, $C_{\Delta tot}$, is the same as that of the current total capacitance. In addition, the capacitance to be added to the numerator, $C_{\Delta H}$, is determined by the sequence of the previous comparator results. In particular, $C_{\Delta H}$ to generate C_{Hi} at the i-th step is

$$C_{\Delta H} = \left(2^{i-1}d_0 + 2^{i-2}d_1 + \cdots + 2^1 d_{i-2} + 2d_{i-1}\right)\cdot C_u, \quad (2)$$

where C_u represents the unit capacitance. Therefore, the i-th capacitor section consists of i capacitors as shown in Fig. 4, where d_k is the k-th bit of the digital code. The total capacitance of the i-th section is 2^iC_u, which is equal to the total capacitance aggregated over all the previous sections. Each capacitor is associated with two switches controlled by one of the previously determined bits. A capacitor is connected to GND through switch S_j^0 if the corresponding bit d_j is 0, otherwise it is connected to V_{ref} through S_j^1. Since all the capacitors controlled by the same bit can be connected together, a section has a pair of switches for the left-most capacitor and has connections to the previous sections for the other capacitors. Once the position of a switch in the JS capacitor array is determined, the switch never changes its position during the redistribution mode. Therefore, the JS capacitor array can achieve remarkable energy saving compared to the splitting capacitor structure [7, 8] and

the conventional structure.

III. PROPOSED CAPACITOR ARRAY

1. Proposed Redundant Search

Although the JS capacitor array is effective in reducing energy consumption, it is more susceptible to mismatch than the conventional array is. Especially in the earlier steps, the total capacitance of the JS capacitor array is much smaller than that of the conventional array, which means that the comparison is easily affected by a small capacitor mismatch. The mismatch among unit capacitors makes V_{out} different from the desired voltage, and thus the comparison may result in incorrect decisions especially in the first few steps of the redistribution mode. As the binary search algorithm never overcomes such erroneous decisions, it is possible to have many missing codes.

The proposed redundant search allows multiple paths to overcome incorrect decisions in the earlier steps. Fig. 6 shows an example of the proposed search tree generated for 5-bit conversion. The binary steps and the redundant steps are denoted as Step Bx and Step Rx, respectively. Like the JS capacitor array, the total capacitance of the proposed array is increased by appending a section at a time. To introduce multiple recovery paths, each capacitor section is utilized two times: one for a redundant step and the other for a binary step. The redundant search starts from Step R2 as the earlier steps cannot provide multiple paths. If redundant steps occur all the time between binary steps except the first one, the proposed search tree can employ n–2 redundant steps for n-bit conversion.

Suppose that the right child is selected at Step B0. In this case, the sub-tree starting from the node of 3/4 contains leaf nodes ranging from 3/32 to 31/32. If the left node of 1/4 is selected at Step B0, the sub-tree covers leaf nodes ranging from 1/32 to 29/32. As the two sub-trees are overlapped a lot, the incorrect decision made at step B0 can be corrected at the later steps, if step B0 does not decide that 1/32 is greater than 1/2 or 31/32 is less than 1/2. The main point of the proposed redundant array is that the earlier incorrect decisions can be corrected at the later steps. This means that the redundant capacitor array allows a relatively large capacitor mismatch and thus can be built with small-sized unit-capacitors. Another meaning is that rough comparison is allowed in the earlier steps. Therefore, we can reduce the power consumed in the comparison by utilizing a low-precision comparator in the earlier steps, which is an additional advantage of the proposed redundant array [20]. As this is not the focus of this paper, however, we will not go further.

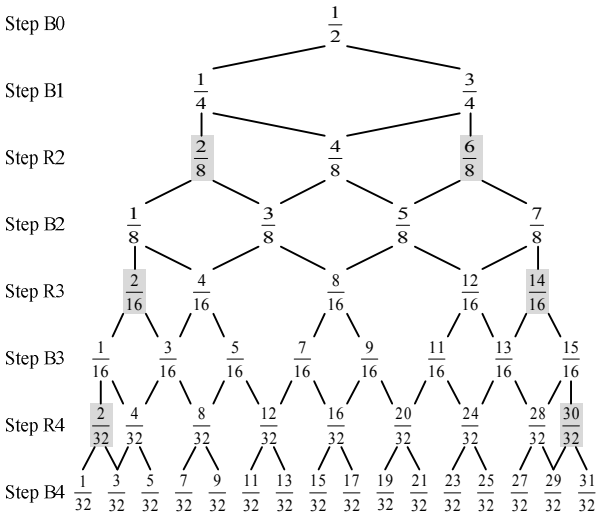

Fig. 6. Proposed redundant search tree for a 5-bit conversion. The irregular redundant nodes are colored gray.

The comparator decision in Step B(x–1) determines the next comparison node of redundant Step Rx. Every node in Step B(x–1) has a value of $(2k+1)/2^x$, where k is a non-negative integer less than 2^{x-1}. Based on the decision in Step B(x–1), the next comparison value of voltage Rx is

$$V_{Rx} = \frac{2\times(2k+1)+(-1)^{d_{B(x-1)}+1}\times 2}{2^{x+1}} - V_{in}, \qquad (3)$$

where $d_{B(x-1)}$ is the comparator decision in Step B(x–1). Note that the denominator in (3) is twice larger than that of Step B(x–1) as the JS capacitor array appends the same capacitance as that of the previous array. For example, a node of 5/8 in Step B2 induces a comparison node of 8/16 if the decision is 0. Otherwise, it leads to a comparison node of 12/16. Two nodes located at the left-most and the right-most positions in a redundant step are irregular, but they are considered to provide more multiple paths. The irregular nodes, which are shaded in Fig. 6, are invoked when the previous decisions are all

zeros or all ones. After completing Step Rx, the current section is reorganized to derive the next comparison for Step Bx. As shown in Fig. 6, the denominator of Step Bx is equal to that of Step Rx, and the numerator is different only by one, which means that the numerator of Step Bx can be derived by changing the numerator of Step Rx slightly instead of appending a capacitor section. The decision made in Step Rx is used to derive the comparison value of Step Bx as follows;

$$V_{\mathrm{B}x} = \frac{H_{\mathrm{R}x} + (-1)^{d_{\mathrm{R}x}+1}}{2^{x+1}} - V_{\mathrm{in}}, \tag{4}$$

where $H_{\mathrm{R}x}$ and $d_{\mathrm{R}x}$ are the numerator and the comparator decision of step Rx, respectively. As the proposed search can provide n–2 redundant steps for n-bit conversion, the digital code is obtained in 2n–2 cycles. For the example of 5-bit conversion, three redundant steps occur between binary searches and the conversion takes 8 cycles.

In the conventional or the JS capacitor array, the digital code for n-bit conversion is derived by concatenating the decision values. In other words, the digital code is

$$\sum_{i=0}^{n-1} 2^{n-1-i} \cdot d_i \tag{5}$$

However, as the proposed redundant search can correct incorrect decisions, the digital code should be decided considering some correction terms as follows;

$$\sum_{i=0}^{n-1} 2^{n-1-i} \cdot d_{\mathrm{B}i} + \sum_{i=2}^{n-1} 2^{n-1-i} \cdot (-1)^{d_{\mathrm{B}(i-1)}+1+L_i} + \sum_{i=2}^{n-1} 2^{n-1-i} \cdot (-1)^{d_{\mathrm{R}i}+1+R_i}, \tag{6}$$

where L_i (R_i) is 1 when we have to visit the left (right) irregular node in Step Ri.

Unlike the JS structure that monotonically increases the numerator in each step, the proposed redundant search tree sometimes decreases the numerator in order to recover incorrect decisions. As some capacitors should be discharged, the proposed search consumes more energy than the JS capacitor does. However, the energy increase is not severe because the proposed redundant search does not increase the entire capacitance. An efficient control scheme is also proposed in the next subsection to reduce the increment of energy consumption.

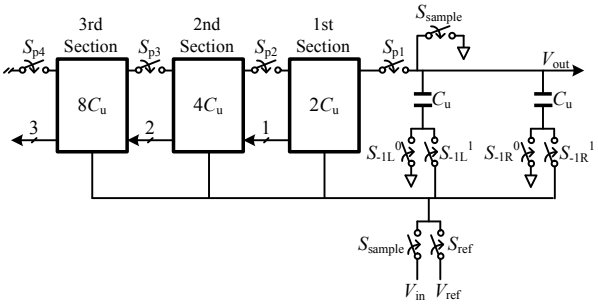

Fig. 7. Proposed capacitor array structure for redundant search.

2. Proposed Capacitor Control

The capacitor array for the proposed redundant search is depicted in Fig. 7. Compared to the JS architecture, two switches, $S_{-1\mathrm{R}}{}^0$ and $S_{-1\mathrm{R}}{}^1$, are additionally included in the capacitor array. Except these switches, each capacitor section is exactly identical to that of the JS capacitor array. After the sample mode, all the switches on the top plate are open. In Step B0, we close $S_{-1\mathrm{R}}{}^1$ and $S_{-1\mathrm{L}}{}^0$ and open $S_{-1\mathrm{R}}{}^0$ and $S_{-1\mathrm{L}}{}^1$ to compare $0.5V_{\mathrm{ref}}$ with the input voltage. After the comparison, the first section containing $2C_{\mathrm{u}}$ capacitance is connected by closing $S_{\mathrm{p}}{}^1$. At the same time, two switches in the first section are controlled by the previous comparison result. After that, the adjacent capacitor section is serially appended in every two cycles to provide a redundant step and a binary step alternatively.

Fig. 8 illustrates how to generate the next comparison node after Step B1. Fig. 8(a) depicts the method to generate the next comparison node of Step Rx from Step B(x–1), where shaded are two irregular nodes in Step Rx. In Step Rx, the next section is appended to increase the total capacitance. In Step B(x–1), we close two switches, $S_{-1\mathrm{R}}{}^1$ and $S_{-1\mathrm{L}}{}^0$, and open the counterparts, $S_{-1\mathrm{R}}{}^0$ and $S_{-1\mathrm{L}}{}^1$. The numerator in Step B(x–1) becomes a positive odd integer, as only one of the first two capacitors in Fig. 7 is connected to V_{ref}. The odd numerator can be represented as 2k+1, where k is an integer in the range of [0, 2^{x-1}).

When the comparator decision in Step B(x–1) is 0, i.e., $d_{\mathrm{B}(x-1)}$=0, the next capacitor section to be appended is controlled to increase the numerator by 2k. More specifically, the next capacitor section is controlled such

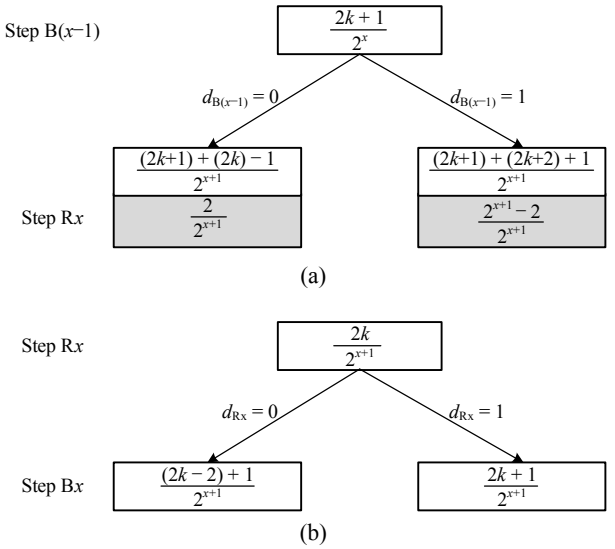

Fig. 8. Construction method for the proposed redundant tree. (a) Generation of Step R*x* from Step B(*x*–1), (b) generation of Step B*x* from Step R*x*.

that 2*k* unit-capacitors are connected to V_{ref}. To produce a proper numerator value, we close $S_{-1R}{}^0$ and open $S_{-1R}{}^1$ to connect the first capacitor to GND. As we discharge one unit-capacitor when $d_{B(x-1)}$=0, the numerator of Step R*x* in Fig. 8(a) has a minus-one term. If all the previous decisions including $d_{B(x-1)}$ are zeros, an irregular node of which numerator is 2 is considered to provide more redundant paths. The irregular node is the left shaded child in Fig. 8(a). In this case, all the capacitors in the capacitor sections including the newly appended section are connected to GND because all the previous decisions are zeros. To make a numerator of 2 required in the irregular node, both the first two capacitors are connected to V_{ref}.

In case of $d_{B(x-1)}$=1, the control scheme is similar to the above case. The newly appended section is controlled to have 2*k*+2 unit-capacitors connected to V_{ref} by using the previous decisions. However, the first capacitor remains to be connected to V_{ref} by controlling $S_{-1R}{}^0$ and $S_{-1R}{}^1$. Instead of changing the connection of the first capacitor, we change the bottom plate of the second capacitor from GND to V_{ref} by closing $S_{-1L}{}^1$ and opening $S_{-1L}{}^0$. As one unit-capacitor is newly connected to V_{ref}, the numerator in Fig. 8(a) has a plus-one term. If the previous decisions are all ones, we consider an irregular node that is the right shaded one in Fig. 8(a). In this case, both the first two capacitors are connected to GND.

Fig. 8(b) shows how to generate the comparison node

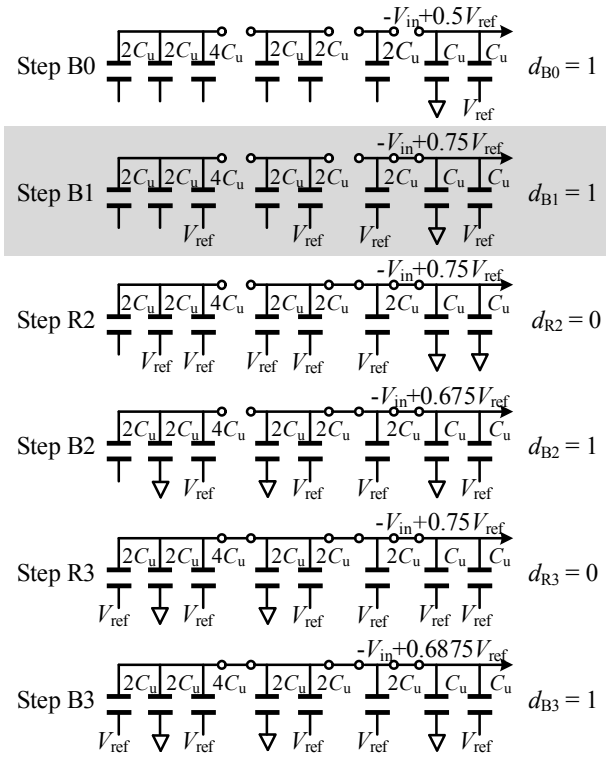

Fig. 9. Switching example of a 4-bit ADC. The correct digital code is 1011, but a single error occurs in Step B1.

of a binary Step B*x* from the previous Step R*x*. For the sake of simplicity, the previous comparison node in Step R*x* is represented as $2k/2^{x+1}$. As the numerator is odd in Step B*x*, only one of the first two capacitors in Fig. 7 should be connected to V_{ref}. To make the numerator odd, we close $S_{-1R}{}^1$ and $S_{-1L}{}^0$ and open $S_{-1R}{}^0$ and $S_{-1L}{}^1$. The rest capacitors are controlled based on d_{Rx}. If d_{Rx} is 1, 2*k* unit-capacitors are connected to V_{ref}. Otherwise, 2*k*–2 unit-capacitors are connected to V_{ref}. Note that the capacitor sections retain their previous values, if the previous decisions have no errors. Only the error correction paths may change the control signals of the capacitor sections and result in the energy-wasting discharge process. As the error-correction step changes the capacitance connected to V_{ref} by at most one unit-capacitance, the energy waste caused by the recovery step is not significant. During the conversion based on the proposed redundant tree, the first two capacitors may charge or discharge to produce the proper numerator. As the first two capacitors are the smallest ones, the energy consumption of the capacitors is relatively smaller than those of the other capacitors.

The proposed control scheme is illustrated in Fig. 9 by

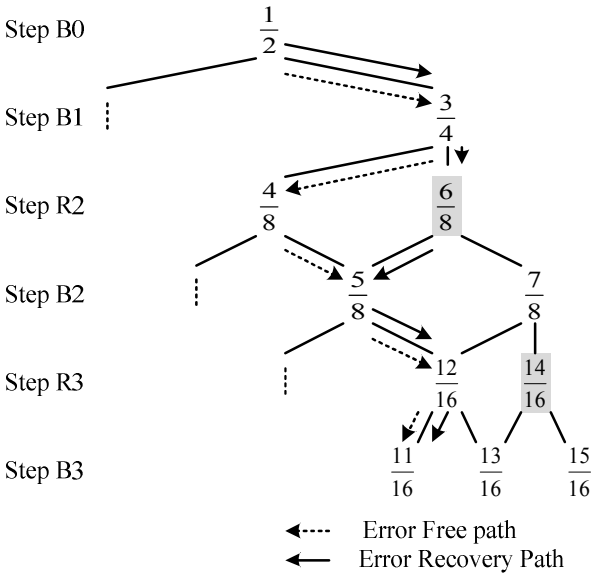

Fig. 10. Error correcting sequence for the 4-bit ADC example.

taking an example of 4-bit conversion. Suppose that the input voltage corresponds to a digital code of 1011, and assume that an error occurs in Step B1. Fig. 10 depicts how the error can be recovered by using redundant paths. Note that both the error-free path and the error recovery path reach the same destination corresponding to the digital code of 1011. The error resulting from Step B1, which is shaded in Fig. 10, is recovered through the following steps, as the redundant search provides multiple error-correction paths. As the first two decisions are both ones, Step R2 considers the irregular node shown at the right-hand side of Fig. 10. In this case, the first two capacitors are connected to GND. Hence, the error of Step B1 is recovered in Step R2. In the second section, the capacitors connected to S_1^0 and S_1^1 are discharged to provide a recovered comparison node and the path is recovered from the error.

The comparison voltage of a redundant node is equal to the voltage compared in the previous step. Due to the capacitance increased by a factor of two, the comparison voltage of the redundant node is always more reliable than the previous comparison voltage which is determined based on the smaller capacitance. As the capacitance ratio resulting from small capacitors is more subject to capacitor mismatch, in addition, the proposed redundant search can be applied only to a few earlier steps, if the capacitor mismatch is small enough to guarantee a reliable ratio at the remaining steps. For example, a 10-bit ADC can be implemented by applying the redundant search to the first 5-bit conversion and the traditional binary search to the rest 5-bit conversion. As the entire capacitance is equal to that of the JS capacitor array, the proposed redundant search is expected to consume almost the same energy while enhancing the linearity dramatically.

3. Discussion on Overheads

The proposed algorithm can insert n–2 redundant steps maximally, and thus the overall conversion steps including the sampling step can be increased to $2n$–1 when the full maximum redundant steps are employed. Hence, the overall conversion time increases compared to the previous binary search. In addition, the redundant array increase the number of comparisons invoked for a sample, increasing the energy consumed in the comparator for a sample, increasing the energy consumed in the comparator. However, the number of redundant steps is dependent on the capacitor mismatch. If the capacitor mismatch is small, it is sufficient to insert a few redundant steps into earlier steps. In achieving 10-bit accuracy with 1% capacitor mismatch, for example, it is enough to insert 4 redundant steps. Without employing the redundant steps, we can increase the unit-capacitor size as an alternative way to meet the specified accuracy, which increases the overall area and energy consumption of the SAR ADC. As the proposed redundant array overcomes the capacitor mismatch without enlarging the unit-capacitor size, it is an effective method to achieve a low-area SAR ADC.

In this paper, we choose 5.4 fF as unit capacitance, the smallest metal-insulator-metal (MIM) capacitor provided in a 0.13 μm CMOS process. Note that the number of redundant steps should be optimized to minimize energy and speed overheads. For this, a statistical approach is discussed in Section IV. In addition, simulation results in [21] show that the additional conversion time caused by the redundant search algorithm can be reduced by allowing the incomplete settling in the DAC.

The control circuit of the proposed architecture is more complex than that of the conventional structure requiring only shifted registers and simple control circuitry. The proposed redundant algorithm requires more complicated circuits including a recalculation block

and a dedicated FSM for irregular steps. In a 0.13 μm process, the control circuit of the proposed architecture requires 40% more equivalent gates compared to the conventional method, if designed for the full redundant case. The increment is also dependent on the number of redundant steps employed. If 4 redundant steps are used to achieve 10-bit accuracy with 1% capacitor mismatch, for example, the increment is about 25%, which is acceptable if we consider the linearity improved by the redundant steps. To improve linearity, previous works have considered calibration techniques [15, 16] that are in general associated with much more complex algorithms and large hardware complexity. The adaptive background digital calibration circuit presented in [15] requires a down converter, an average-calculation unit and a truncation unit. In [16], the complicated analog input stage having multiple-fingered registers is used to compensate the peak DNL. Moreover, some calibration techniques necessitate an additional reference ADC. Note that the proposed work improves the linearity without using such calibration circuits.

As the proposed capacitor array is connected through the top-plate switches as shown in Fig. 7, the nonlinear characteristics of switches may degrade the linearity. However, the previous incorrect decisions caused by the nonlinear characteristics of the top-plate switches can be corrected by multiple paths enabled in the proposed redundant steps. Various simulations will be discussed in Section V, including the effects of nonlinear switches.

IV. STATISTICAL ANALYSIS

More recovery paths can be enabled by providing more steps, but the switching energy increases according to the increased number of redundant steps. Therefore, the number of redundant steps should be optimized to achieve a low-energy, mismatch-tolerant SAR ADC. Given a capacitor mismatch ratio, the required number of redundant steps can be calculated statistically. A practical method to make a large capacitor is to connect multiple unit-capacitors as shown in Fig. 11. Note that dummy capacitors are placed around the connected unit-capacitors to remove in effect the mismatches between the unit-capacitors at the boundary and those on the inside [23].

For the statistical analysis of mismatch effects,

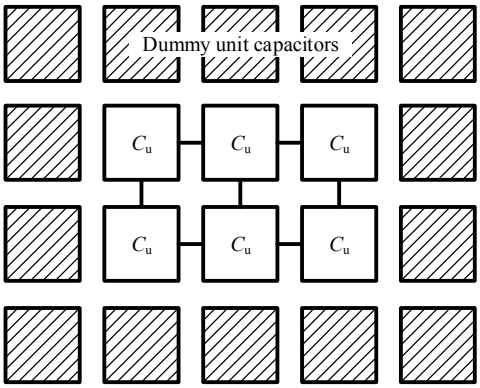

Fig. 11. Layout technique for drawing a large capacitor.

suppose that the capacitance of the unit-capacitor has a Gaussian distribution with mean C_u and variance σ^2. A single unit-capacitor C_1 can be modeled as

$$C_1 = C_u + e_1 C_u, \quad e_1 \sim N(0, \sigma^2), \tag{7}$$

where e_1 is the mismatch ratio of a single unit-capacitor. If we set the maximum mismatch ratio of the unit-capacitor, e_{max}, to 3σ, 99.7% of unit-capacitors are associated with capacitor mismatch ratios less than e_{max}. For the large capacitor C_N generated by connecting N unit-capacitors, we have a statistical model as follows;

$$C_N = \sum_{i=1}^{N} C_1 = NC_u + e_N C_u, \quad e_N \sim N(0, N\sigma^2). \tag{8}$$

With a confidence of 99.7%, therefore, the maximum mismatch ratio of C_N is

$$e_{N\max} = \frac{\sqrt{N}C_u}{NC_u} e_{max} = \frac{1}{\sqrt{N}} e_{max}. \tag{9}$$

Note that the maximum mismatch ratio of C_N is decreased by a factor of $\sqrt{N}$ compared to that of one unit-capacitor, which means that the smaller capacitor is more subject to capacitor mismatch than the larger capacitor does. In the binary search, the comparison voltage at the i-th step is

$$\frac{C_H}{C_{2^i}} = \frac{H \cdot C_u + e_H \cdot C_u}{2^i \cdot C_u + e_H \cdot C_u + e_L \cdot C_u}, \tag{10}$$

where H and L stand for the numbers of unit-capacitors

Table 1. Maximum tolerance of capacitor mismatch for various redundant ADC configurations

Redundant level	5-bit ADC	6-bit ADC	7-bit ADC	8-bit ADC	9-bit ADC	10-bit ADC
1 step	0.1290	0.0648	0.0325	0.0163	0.0081	0.0041
2 steps	0.1778	0.0891	0.0446	0.0223	0.0112	0.0056
3 steps	0.2501	0.1252	0.0626	0.0313	0.0157	0.0078
4 steps	-	0.1778	0.0884	0.0442	0.0221	0.0111
5 steps	-	-	0.1250	0.0625	0.0313	0.0156
6 steps	-	-	-	0.0884	0.0332	0.0221
7 steps	-	-	-	-	0.0624	0.0313
8 steps	-	-	-	-	-	0.0442

connected to V_{ref} and GND, respectively. For n-bit conversion, the error from the mismatch should be less than $1/2^n$. Using (9) and (10), we can derive constraints to be satisfied for the 99.7% confidence level. At Step (i-1) in which the total capacitance is 2^iC_{u}, the constraints are

$$\frac{H+\sqrt{H}\cdot e_{\max}}{2^i+\sqrt{H}\cdot e_{\max}-\sqrt{2^i-H}\cdot e_{\max}}<\frac{H}{2^i}+\frac{1}{2^n}$$
$$\frac{H-\sqrt{H}\cdot e_{\max}}{2^i-\sqrt{H}\cdot e_{\max}+\sqrt{2^i-H}\cdot e_{\max}}>\frac{H}{2^i}-\frac{1}{2^n} \quad (11)$$

Therefore,

$$e_{\max}<\frac{2^i}{\sqrt{H}\cdot 2^n-(\sqrt{H}-\sqrt{2^i-H})(H\cdot 2^{n-i}+1)} \quad (12)$$

The inequalities in (11) are considering two extreme cases of (10). The upper inequality considers one case that all the unit-capacitors related to H have positive mismatches and the other unit-capacitors have negative mismatches, and the lower considers the opposite. From (12), we can calculate the mismatch ratio allowed for every node in the binary search. The maximum ratio at Step (i–1) associated with 2^i unit-capacitors, $e_{\max}{}^{i-1}$, can be calculated by exploring all the possible values of H as follows;

$$e_{\max}^{i-1}=\text{MIN}\frac{2^i}{\sqrt{H}\cdot 2^n-(\sqrt{H}-\sqrt{2^i-H})(H\cdot 2^{n-i}+1)} \quad (13)$$

Given a desired precision and a unit-capacitor mismatch ratio, we can identify a reliable step by evaluating (13). For various configurations, Table 1 shows the maximum mismatch ratio that can be tolerated in each redundant step. If the mismatch ratio is 3%, for example, Table 1 says that 7 redundant steps should be used for a 10-bit ADC.

Table 2. Switches and conversion steps required in n-bit ADCs

ADC type	Conversion steps	Required Switches	Normalized Average Power
Conventional [22]	n+1	2n+5	1.000
Splitting Cap [7]	n+1	4n+3	0.625
Original JS [12]	n+1	3n+2	0.250
Redundant (n–2 steps)	2n-1	3n+4	0.350

V. SIMULATION RESULTS

1. Switching Energy Consumption

To compare the switching energy in the capacitor arrays, the C-level simulator is newly developed to model several 10-bit SAR ADCs including the conventional capacitor array, the splitting capacitor array [8], the JS capacitor array [12] and the proposed redundant array. Table 2 compares four methods in terms of the number of conversion steps, the number of switches and the normalized energy consumption. In counting the number of switches, we include the switches required in the sample and hold modes. The total capacitance required in each structure is not presented in Table 2, because all the structures have the same amount of capacitance. Compared to the JS capacitor array, Fig. 12(a) shows the energy consumed in the proposed capacitor array having x redundant steps. All the values are normalized by the average energy of the conventional structure. Note that the maximum number of redundant steps in the proposed architecture is 8 for 10-bit conversion. Fig. 12(b) reveals that the average energy dissipated in the proposed architecture increases as the number of redundant steps increases. However, the increment is not severe even compared to the average energy consumption of the JS capacitor array. The proposed redundant array saves about 65% of energy compared to the conventional array.

2. Accuracy of Conversion

For the sake of precise comparisons, the conversion

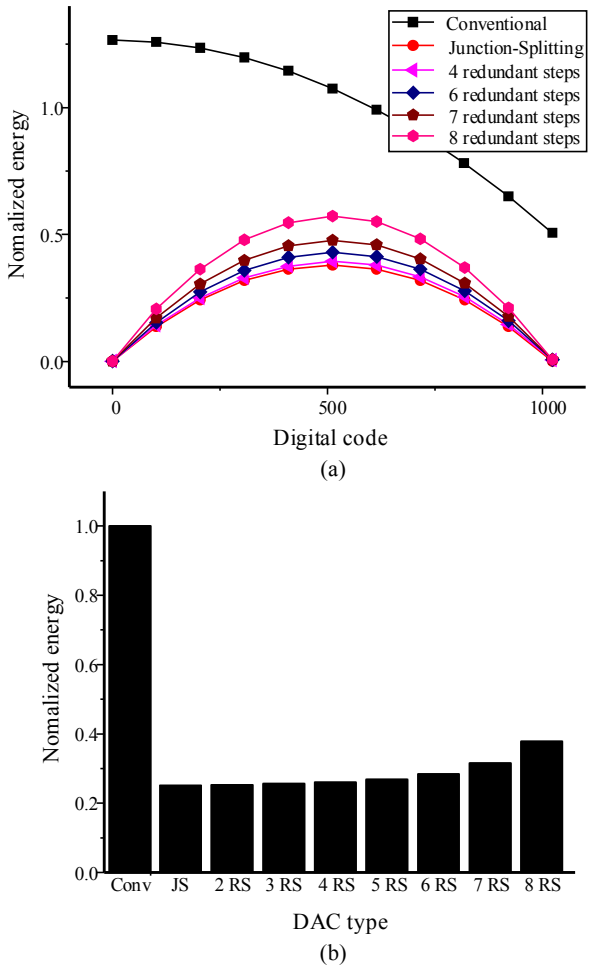

Fig. 12. Switching energy dissipated in the redundant DAC and comparisons with previous works (a) Comparison for each input code, (b) Average energy comparison for the conventional (Conv), the junction-splitting (JS) and the proposed redundant capacitor arrays, where RS stands for redundant steps.

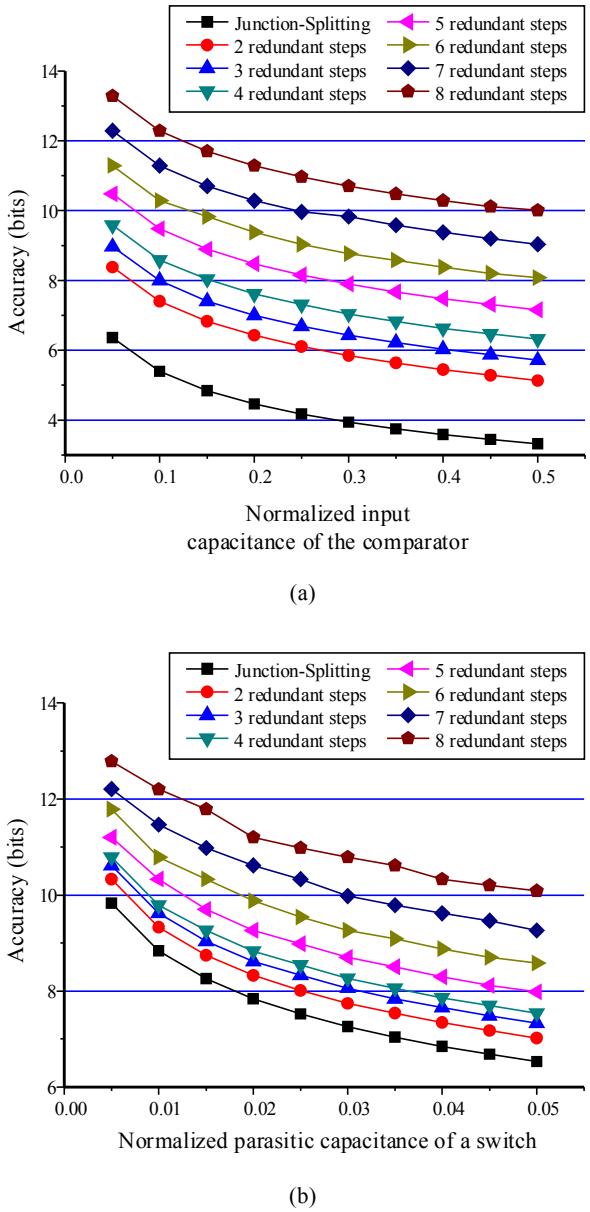

Fig. 13. Accuracy improvement (a) for the input capacitance of the comparator, (b) for the parasitic capacitance of top-plate switches. The capacitance is normalized by C_u.

accuracy is expressed in units of bit as follows;

$$\text{Accuracy} = \log_2\left(\text{Error}_{\text{MAX}}\right) \tag{14}$$

where $\text{Error}_{\text{MAX}}$ represents the maximum error between the ideal voltage, that changes output digital code, and the actual one resulting from the capacitor array. If there are missing codes, the accuracy becomes lower than the desired resolution. To optimize the number of redundant steps, the accuracy of 10-bit conversion resulting from various configurations is plotted shown in Fig. 13. According to Table 1 obtained by (13), we need to insert 7 redundant steps if the capacitor mismatch ratio is 3%. In Fig. 13, the accuracy of 10 bits is achieved by using 7 or 8 redundant steps for 3% capacitor mismatch. Note that the statistical analysis is consistent with the simulation result.

In the SAR ADC, some analog components such as comparators and switches have to be carefully designed, because their characteristics affect the linearity directly. The parasitic capacitors connected to the top plate of the DAC capacitor array shown in Fig. 1 may involve in the charge redistribution process and degrade the linearity severely. Especially, the original JS capacitor array is easily affected by the parasitic capacitors in the earlier redistribution steps associated with small capacitance. As

the comparator has a relatively large input capacitance sometimes, the input capacitance can affect the first few steps seriously, leading to lots of errors. Fig. 13(a) shows how the input capacitance of the comparator degrades the accuracy of conversion, where the input capacitance is normalized by the unit-capacitance. To take into account only the effect of the input capacitance, the simulation assumes that there is no mismatch in the capacitor array. When the input capacitance is only 5% of C_u, the accuracy is severely degraded if redundant steps are not used at all. However, errors from the input capacitance can be almost recovered by using redundant steps. As shown in Fig. 13(a), the full redundant case maintains 10-bit accuracy until the input capacitance increases up to 50% of C_u.

In the JS array that generates the next comparison voltage by appending a capacitor section, a switch is used to connect the top plates of adjacent two sections. There are two parasitic factors to be considered in the design of the switches: resistance and capacitance. The parasitic resistance affects the redistribution time a little, but the influence can be ignored in our SPICE simulations. The parasitic capacitance can reduce the accuracy. Fig. 13(b) shows how the accuracy is affected by the parasitic capacitance, where the parasitic capacitance is normalized by C_u. Though the JS capacitor array loses its accuracy even if the parasitic capacitance is as small as 0.5%, the full redundant capacitor array can tolerate up to 5%.

3. Linearity

The redundant search is also effective in enhancing the linearity of the ADC, especially the integral nonlinearity (INL) and differential nonlinearity (DNL), as it can drastically reduce the number of missing codes. For 10-bit conversion, Fig. 14 depicts how the INL and DNL change according to the increase of capacitor mismatch. The INL and DNL of the JS array are severely deteriorated by capacitor mismatch, but the linearity is recovered by adopting the full redundant search. Fig. 15 shows the peak INL and DNL caused by capacitor mismatch, where we can see that inserting redundant steps is an efficient way of reducing them. The peak INL and DNL caused by the input capacitance of the comparator and the parasitic capacitance of the top-plate

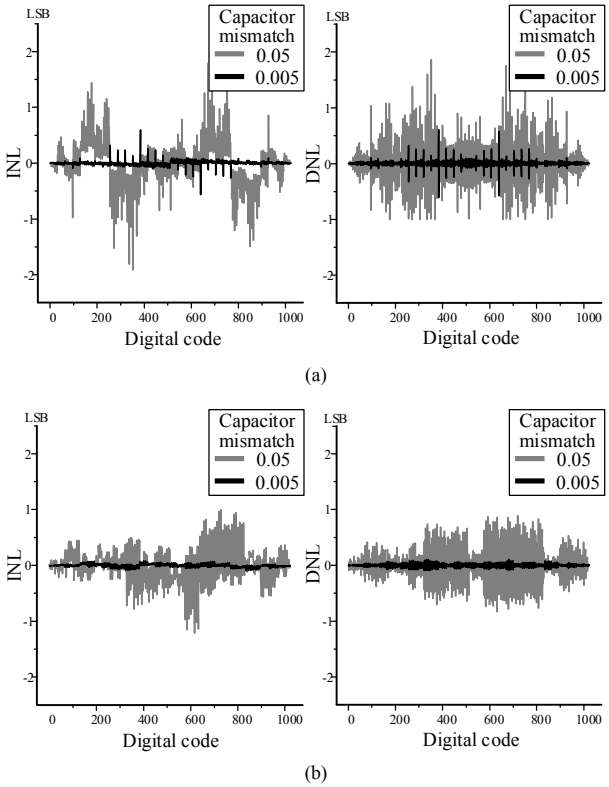

Fig. 14. INL and DNL of a 10-bit ADC for (a) the JS capacitor array, (b) the redundant array with 8 redundant steps.

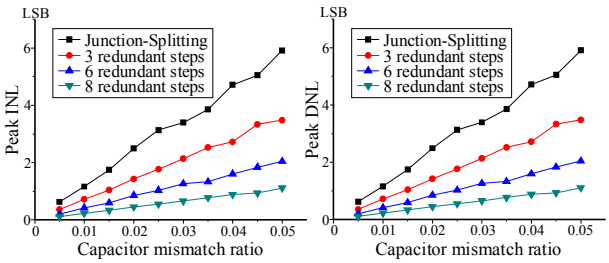

Fig. 15. Peak INL and DNL for a 10-bit ADC.

switches are shown in Fig. 16 and 17, respectively. In addition, charge injection may reduce the linearity and increase the number of redundant steps required. As the charges stored in the capacitor array can be charged or discharged by the top-plate switches, the charge injection may reduce the linearity of ADC severely. The amount of injecting charges is modeled as an additional capacitor connected to V_{ref} or GND. The additional capacitors corresponding to charge injection are connected to V_{ref} or GND in a random manner, and assumed to have the same capacitance to simulate the worst case. Fig. 18 depicts the peak INL and DNL caused by the charge injection, where the capacitance value corresponding to charge

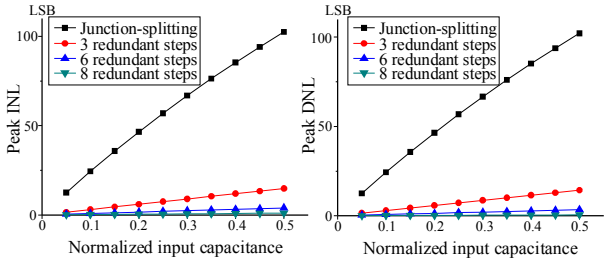

Fig. 16. Peak INL and DNL caused by the input capacitance of the comparator.

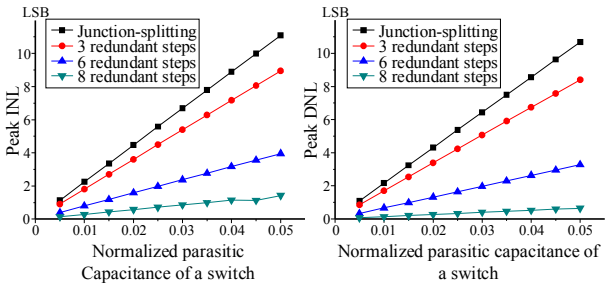

Fig. 17. Peak INL and DNL caused by the parasitic capacitance of top-plate switches.

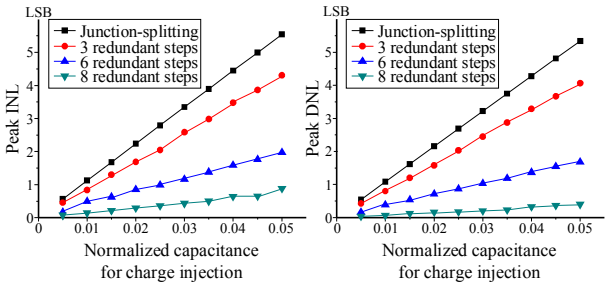

Fig. 18. Peak INL and DNL caused by the charge injection.

injection is normalized to the unit capacitance. The proposed redundant array is effective in improving the linearity that can be degraded by charge injection.

VI. CONCLUSIONS

In this paper, we have presented an efficient redundant search algorithm suitable for the JS capacitor array and its switching algorithm that alleviates the effects of capacitor mismatch and parasitic capacitors. To recover incorrect decisions, redundant steps are adopted by modifying the structure of the JS capacitor array slightly. The proposed redundant search, which is enabled by inserting two additional switches into the original JS array, enhances the linearity resulting from capacitor mismatch and parasitic capacitors without much increasing switching energy. In addition, the number of redundant steps appropriate for a given mismatch ratio has been analyzed statistically to minimize the overheads of the proposed redundant array. Simulation results show that the proposed redundant search recovers most of the linearity degradation caused by mismatch, parasitic capacitors and nonlinearities of switches, while keeping the energy consumption similar to that of the JS capacitor array.

ACKNOWLEDGMENTS

This work was supported by the National Research Foundation (NRF) grant funded by the Korea government (MSIP) (2016R1C1B1007593) and the IC Design Education Center (IDEC).

REFERENCES

[1] J.-I. Kim, D.-R. Oh, D.-S. Jo. B.-R.-S. Sung, and S.-T. Ryu, “A 65nm CMOS 7b 2 GS/s 20.7 mW flash ADC with cascaded latch interpolation,” *IEEE J. Solid-State Circuits*, Vol. 50, No. 10, pp. 2319-2330, Oct. 2015.

[2] H.-Y. Lee, D.-G. Jeong, Y.-J. Hwang, H.-B. Lee, and Y.-C. Jang, “A 1-V 1.6-GS/s 5.58-ENOB CMOS flash ADC using time-domain comparator,” *J. Semicond. Technol. Sci.*, Vol. 15, No. 6, pp. 695-702, Dec. 2015.

[3] J. Yuan, S. W. Fung, K. Y. Chan, and R. Xu, “A 12-bit 20 MS/s 56.3 mW pipelined ADC with interpolation-based nonlinear calibration,” *IEEE Trans. Circuits Syst. I, Reg. Papers*, Vol. 59, No. 3, pp. 555-565, Mar. 2012.

[4] J.-S. Park, et al., “A 12b 100 MS/s three-step hybrid pipelined ADC based on time-interleaved SAR ADCs,” *J. Semicond. Technol. Sci.*, Vol. 14, No. 2, pp. 189-197, Apr. 2014.

[5] J. L. McCreary and P. R. Gray, "All-MOS charge redistribution analog-to-digital conversion techniques," *IEEE J. Solid-State Circuits*, vol. SC-10, no. 12, pp. 371-379, Dec. 1975.

[6] J. Craninckx and G. V. D. Plas, "A 65fJ/conversion-step 0-to-50MS/s 0-to-0.7mW 9b charge-sharing SAR ADC in 90nm digital CMOS," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech.*

Papers, 2007, pp. 246-247.

[7] B. P. Ginsburg and A. P. Chandrakasan, "500-MS/s 5-bit ADC in 65-nm CMOS With Split Capacitor Array DAC," *IEEE J. Solid-State Circuits*, vol. 42, no. 4, pp. 739-747, Apr. 2007.

[8] B. P. Ginsburg and A. P. Chandrakasan, "An Energy-Efficient Charge Recycling Approach for a SAR converter With Capacitive DAC," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2005, pp. 184-187.

[9] W. Y. Pang et al., "A 10-bi 500-KS/s Low Power SAR ADC with Splitting Comparator for Bio-Medical Application," in *IEEE Asian Solid-State Circuits Conf. Dig. Tech. Papers*, 2009, pp. 149-152.

[10] C. C. Liu et al., "A 0.92mW 10-bit 50-MS/s SAR ADC in 0.13um CMOS Process", in *Proc. IEEE Symp. VLSI Circuits*, 2009, pp. 236-237.

[11] C. C. Liu et al., "A 10-bit 50-MS/s SAR ADC With a Monotonic Capacitor Switching Procedure," *IEEE J. Solid-State Circuits*, vol. 45, no. 4, pp. 731-740, Apr. 2010.

[12] J. S. Lee and I. C. Park, "Capacitor Array Structure and Switch Control for Energy-Efficient SAR Analog-to-Digital Converters," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2008, pp. 236-239.

[13] W. Yu et al., "Two-step split-junction SAR ADC," *Electron. Lett.*, vol. 46, no. 3, pp. 208-210, Feb. 2010.

[14] K. Bacrania, "A 12-bit Successive-Approximation-Type ADC with Digital Error Correction," *IEEE J. Solid-State Circuits*, vol. SC-21, no. 6, pp. 1016-1025, Dec. 1986.

[15] W. Liu and Y. Chiu, "Background digital calibration of successive approximation ADC with adaptive equalisation," *Electron. Lett.*, vol. 45, pp. 456-458, Apr. 2009.

[16] F. Ohnhaeuser and M. Huemer, "Methods to eliminate dynamic errors in high-performance SAR A/D converter," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2008, pp. 2398-2401.

[17] V. Giannini et al., "An 820uW 9b 40MS/s Noise-Tolerant Dynamic-SAR ADC in 90nm Digital CMOS," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, 2008, pp. 238-239.

[18] M. Hesener et al., "A 14b 40MS/s Redundant SAR ADC with 480MHz Clock in 0.13 CMOS," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, 2007, pp. 248-249.

[19] F. Kuttner, "A 1.2V 10b 20MSample/s Non-Binary Successive Approximation ADC in 0.13um CMOS," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Papers*, 2002, pp. 176-177.

[20] T. Ogawa et al., "SAR ADC algorithm with redundancy," in *Proc. IEEE Asia Pacific Conf. Circuits Syst.*, 2008, pp. 268-271.

[21] Y. Lee and I.-C. Park, "Capacitor array structure and switching control scheme to reduce capacitor mismatch effects for SAR analog-to-digital converters," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2010, pp. 1464-1467.

[22] A. Johns and Ken Martin, Analog Integrated Circuit Design. New York: John Wiley & Sons, Inc., 1997.

[23] Y. Lee, J. Song, and I.-C. Park, "Statistical modeling of capacitor mismatch effects for successive approximation register ADCs," in *Proc. IEEE Int. SoC Design Conf.*, 2011, pp. 302-305.

Youngjoo Lee received the B.S., M.S. and Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 2008, 2010 and 2014, respectively. Since February 2017, he has been an Assistant Professor in the department of Electrical Engineering, POSTECH, Pohang, Korea. Prior to joining POSTECH, he was with Interuniversity Microelectronics Center (IMEC), Leuven, Belgium, from May 2014 to February 2015, where he researched reconfigurable SoC platforms for software-defined radio systems. From March 2015 to February 2017, he was with the Faculty of the Department of Electronic Engineering, Kwangwoon University, Seoul, Korea. His current research interests include the algorithms and architectures for embedded processors, intelligent transportation systems, advanced error-correction codes, and mixed-signal circuit designs.

Taehyoun Oh received the B.S. and M.S. degrees from Seoul National University, Seoul, Korea, in 2005 and 2007, respectively, and the Ph. D. degree from the University of Minnesota, Minneapolis, MN, USA, under the supervision of Dr. R. Harjani, all in electrical engineering. His doctoral research focused on high-speed I/O circuits and architectures. During the summer of 2010, he worked on I/O channel modeling at AMD Boston Design Center, MA. In the fall semester of 2011, he researched I/O architecture and jitter budgeting of the link with Intel Corporation. In the fall of 2012, he joined the IBM System Technology Group, where he worked on performance verification of high-speed decision feedback equalizer for server processors. In the spring of 2013, he joined the Department of Electronic Engineering, Kwangwoon University, Seoul, Korea, as an Assistant Professor.

In-Cheol Park received the B.S. degree in electrical engineering from Seoul National University, Seoul, Korea, in 1986, and the M.S. and Ph.D. degrees in electrical engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, in 1988 and 1992, respectively. Since June 1996, he has been an Assistant Professor and is currently a Professor with the Department of Electrical Engineering, KAIST. Prior to joining KAIST, he was with the IBM T. J. Watson Research Center, Yorktown, NY, USA, from May 1005 to May 1996, where he researched high-speed circuit design. His current research interests include computer-aided design algorithms for high-level synthesis and very large scale integration architectures for general-purpose microprocessors.