PAPER

Behavior-Level Analysis of a Successive Stochastic Approximation Analog-to-Digital Conversion System for Multi-Channel Biomedical Data Acquisition

Sadahiro TANI[†], ***Member***, **Toshimasa MATSUOKA**[†a)], ***Senior Member***, **Yusaku HIRAI**[†], **Toshifumi KURATA**[†], **Keiji TATSUMI**[†], **Tomohiro ASANO**[†], **Masayuki UEDA**[††], ***Nonmembers***, ***and*** **Takatsugu KAMATA**[††], ***Member***

SUMMARY In the present paper, we propose a novel high-resolution analog-to-digital converter (ADC) for low-power biomedical analog front-ends, which we call the successive stochastic approximation ADC. The proposed ADC uses a stochastic flash ADC (SF-ADC) to realize a digitally controlled variable-threshold comparator in a successive-approximation-register ADC (SAR-ADC), which can correct errors originating from the internal digital-to-analog converter in the SAR-ADC. For the residual error after SAR-ADC operation, which can be smaller than thermal noise, the SF-ADC uses the statistical characteristics of noise to achieve high resolution. The SF-ADC output for the residual signal is combined with the SAR-ADC output to obtain high-precision output data using the supervised machine learning method.

key words: *SAR-ADC, DAC error calibration, stochastic A/D conversion, mismatch, machine learning*

1. Introduction

Due to rising health concerns, the need for wearable high-precision biomedical sensors that can operate under battery power for long periods of time is rapidly increasing. Integration of biomedical sensor nodes for monitoring various biopotential signals (e.g., electrocardiograms (ECGs), electroencephalograms (EEGs), or electromyogram (EMGs)), temperature, and pressure, is also highly attractive in practice because a small, low-power, low-cost device can be realized. One way to realize this integration is a flexible and reconfigurable circuit design platform that takes into consideration time-to-market pressure and design complexity [1], [2]. Such a platform can help to realize low-power high-precision analog-to-digital converters (ADCs) that are suitable for use in a flexible and reconfigurable multi-channel biomedical data acquisition analog front-end (AFE), as shown in Fig. 1. The digital signal processing block filters, decimates, and demultiplexes according to the target signals. Although Δ-Σ ADCs are often popular for use in biomedical sensor applications [1], they are not suitable for multi-channel AFEs having fast scan rates. In the present study, an oversampled successive-approximation-register ADC (SAR-ADC) [3] is used to reduce noise and errors within the signal bandwidth. The oversampling ratio *OSR* depends on the number of channels used. In addition, considering low-voltage operation (≤ 0.5 V) for a low-power AFE, a small gain for low-noise amplifiers (LNAs) is preferred in order to reduce nonlinear distortion. Such a situation requires a sufficient dynamic range for the ADC, even under a low supply voltage. There is a trade-off relationship between the power consumption and precision of the ADC [4].

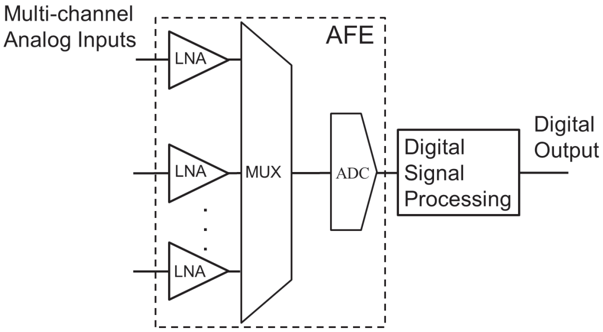


Fig. 1 Multi-channel biomedical analog front-end.

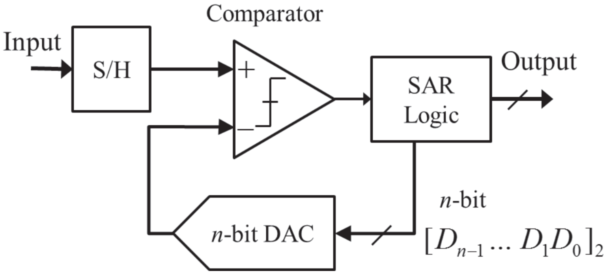


Fig. 2 Successive approximation register ADC (SAR-ADC).

The SAR-ADC shown in Fig. 2 [5] converts continuous analog data to digital output via binary search through all possible quantization levels from D_{n-1} of the most significant bit (MSB) to D_0 of the least significant bit (LSB). At each step of the search, an internal digital-to-analog converter (DAC) and comparators are used repeatedly. Typically, an internal DAC in a SAR-ADC is composed of arrayed capacitors with capacitances weighted by powers of two. The precision of the entire SAR-ADC is significantly affected by the precision of the DAC. However, a

Manuscript received January 23, 2017.
Manuscript revised June 12, 2017.

[†] The authors are with Graduate School of Engineering, Osaka University, Suita-shi, 565-0871 Japan.
[††] The authors are with SPChange, LLC., Yokohama-shi, 224-0032 Japan.
a) E-mail: matsuoka@eei.eng.osaka-u.ac.jp

DOI: 10.1587/transfun.E100.A.2073

high-precision DAC is difficult to realize using small capacitors with a large mismatch. Therefore, error correction techniques are important for realizing high-resolution SAR-ADCs.

The present paper demonstrates the feasibility of a novel low-voltage (≤ 0.5 V) high-resolution ADC system for low-power AFEs, which we call the successive stochastic approximation ADC (SSA-ADC). Although the proposed system is based on SAR-ADC architecture, it has two key features. One is the use of a stochastic flash ADC (SF-ADC) with statistical device mismatches [6]–[9], which can operate as a comparator with a digitally controlled threshold [10]. It is useful to reduce influence of comparator offset in the SAR-ADC operation. In addition, the SF-ADC originates from the principle of stochastic resonance, which is used to detect small signals with noise [11]–[14]. By applying statistical noise characteristics of the SF-ADC to additionally quantize the residual signal in the last SAR-ADC operation, which can be smaller than the thermal noise, signal resolution enhancement can be expected. The other feature of the SSA-ADC is error correction using a machine learning method, which can realize a more complicated error correction function than a simple digital-domain calibration technique [5]. It corrects errors originated from capacitor mismatch in the internal capacitor DAC and non-ideality of the SF-ADC used for the additional quantization. This technique, which was developed based on previous studies on the SAR-ADC [15], [16], can combine the SF-ADC output for the residual signal with the SAR-ADC output to obtain high-precision output data.

2. Stochastic Flash ADC

As shown in Fig. 3, the SF-ADC comprises an array of N comparators and a ones adder. The ones adder outputs a binary code, which corresponds to the number of comparators outputting ‘high’. The SF-ADC is designed based on stochastic resonance, where a weak signal can be enhanced by adding white noise or by employing an input-referred (IR) offset distribution of comparator ensemble [11]–[14]. Although each offset voltage $\Delta_{off,i}$ for individual comparators cannot be predicted, statistical values such as the standard deviation and the mean can be calculated. For a sufficiently large number of comparators, the comparator offset follows a Gaussian distribution. Therefore, the probability $P(V_{in})$ that one comparator outputs one (high) can be expressed as follows [8], [9], [14]:

$$P(V_{in}) = 1 - \frac{1}{2}\mathrm{erfc}\left(\frac{V_{in}}{\sqrt{2}\,\sigma_{off}}\right) \approx \frac{n_H}{N}, \qquad (1)$$

where $\mathrm{erfc}(x) = (2/\sqrt{\pi})\int_x^{\infty}\exp(-z^2)dz$ is the complementary error function, V_{in} is an input voltage, σ_{off} is the standard deviation of the offset voltages, and the mean of the IR offset voltages is assumed to be zero. Moreover, n_H is the number of comparators outputting high and corresponds to the output of the ones adder in Fig. 3. The SF-ADC determines its digital output according to n_H. Based on the characteristics of the SF-ADC, n_H can be used as a digitally controlled variable-threshold comparator (DCVTC) by comparing n_H with the digital threshold D_{th} in order to quantize n_H to 1-bit code, as shown in Fig. 4. This operation mode of the SF-ADC can be used in the SAR-ADC in the present study (SAR-ADC mode). The concept of the DCVTC was also used in a previous study on multi-bit $\Delta - \Sigma$ ADC [10].

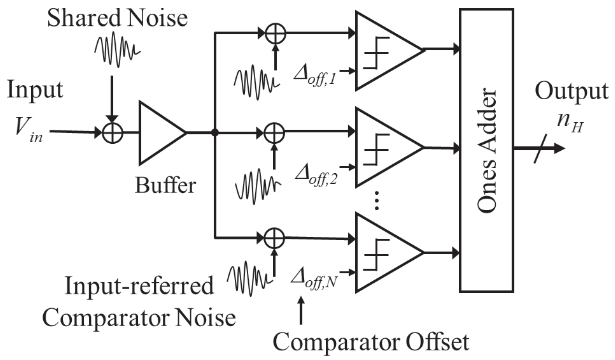


Fig. 3 Stochastic flash ADC (SF-ADC).

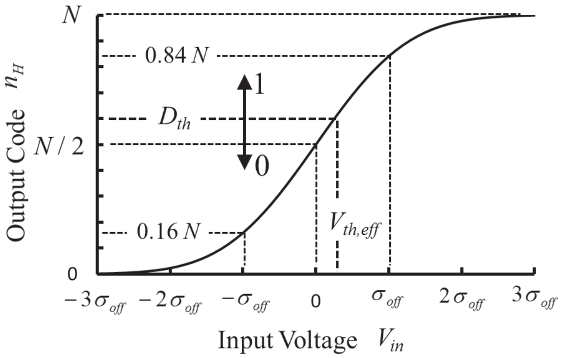


Fig. 4 Characteristics of the SF-ADC.

The SF-ADC also can use statistical noise to additionally quantize residual error in the last SAR-ADC operation in this work (SF-ADC mode), which can be buried under time-varying noise, such as thermal and flicker noise. This technique can enhance the signal resolution so as to detect even signals that are buried under noise. The detail is described in Sect. 3.

Figure 5 shows an example of small-signal detection using stochastic resonance. Even when the signal itself does not exceed the threshold in a comparator, adequately superimposing noise on the signal can generate comparator output according to the signal with a certain probability. If the noise can be assumed to follow a Gaussian distribution, the signal can be restored [14].

Table 1 shows types of noise produced inside the SSA-ADC and their characteristics. Each comparator generates noise independently. From Fig. 3, the IR noise for each comparator can be merged with its offset as an instantaneous comparator offset. In other words, the comparator offset is equivalent to the DC IR noise [14]. Neither the IR noise nor

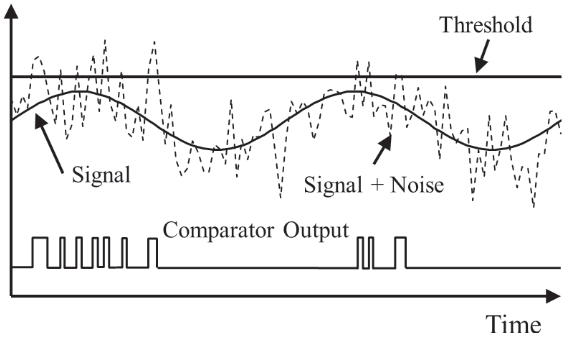


Fig. 5 Stochastic resonance.

Table 1 Classification of noise factors inside the SSA-ADC.

Noise Type	Ensemble Average	Time Average
Shared Noise from DAC and Buffer	Non-zero	Zero
Comparator Noise	Zero	Zero
Comparator Offset	Zero	Non-zero

the offset have a correlation in ensemble statistics. Thus, the standard deviation of the instantaneous comparator offset ($\sigma_{off,eff}$) can easily be estimated as follows:

$$\sigma_{off,eff} = \sqrt{\sigma_{off}^2 + \sigma_n^2}, \quad (2)$$

where σ_n is the standard deviation of IR comparator noise.

In order to use stochastic resonance, the probability of the comparator response to the noise-superimposed signal must be observed at higher resolution. There are two methods by which to enhance the resolution, i.e., increasing the total comparator counts and oversampling. Based on a previous study [14], [17], the resolution is inversely proportional to $\sqrt{N N_S}$, where N_S is the oversampling ratio for the SF-ADC. In the present study, a small value of $\sigma_{off,eff}$ is considered for detecting the noise-level signal, which requires an offset cancellation technique in circuit design. The area of the comparator must be large compared to that of the standard-cell-like comparator [18]. This limits the number of comparators N considering the total occupied area. Unlike the linearization of the piecewise inverse Gaussian approximation [18], error correction and encoding using the supervised machine learning, as described in Sect. 5, allows for the relaxed assumption of Gaussian comparator offset, which is useful for reducing N. In the SSA-ADC, an oversampling technique is also used to quantize the residual signal in the last SAR-ADC operation with high resolution, where the SF-ADC block repeatedly samples the same input several times (N_S), and the SF-ADC output is averaged in the time domain. As shown in Table 1, this oversampling reduces the influence of DAC and buffer noise shared in an ensemble of comparators in the SF-ADC block, which is not reduced by ensemble statistics [14].

3. System Configuration

The system configuration of the proposed SSA-ADC system is shown in Fig. 6. It has two modes for A/D conversion and a test mode for DCVTC calibration described in Sect. 4. In the SAR-ADC mode for MSB-side conversion (solid lines in Fig. 6), the SF-ADC operates as a DCVTC. Some errors associated with the internal DAC in this mode are corrected by controlling the digital threshold D_{th} dynamically at each conversion step, where the controlling data is stored in the register according to a foreground test. In the SF-ADC mode for LSB-side conversion (double lines in Fig. 6), the SF-ADC quantizes residual error in the last SAR-ADC operation. Oversampling and averaging for high resolution can be implemented with a simple cumulative adder. The machine learning function combines the output D_U from the SAR-ADC mode operation and the output D_L from the SF-ADC mode and generates the total output D_{out} with calibration parameters obtained during the learning mode. Bit configurations of D_U, D_L, and D_{out} are expressed in Fig. 7 and the following equation:

$$\begin{aligned} D_U &= (D_{U,N_U-1}, \ldots, D_{U,0}) \\ D_L &= (D_{L,N_L-1}, \ldots, D_{L,0}) \\ D_{out} &= (D_{out,N_{out}-1}, \ldots, D_{out,0}) \end{aligned} \quad (3)$$

Considering N_S oversampling in the SF-ADC mode (oversampling only inside the SF-ADC block in addition to the oversampling described in Sect. 1), the conversion for a sampling point requires at least $N_U + N_S + 1$ steps. In the present study, in order to achieve the target resolution ($\propto 1/\sqrt{N N_S}$ [14], [17]) under limited conversion speed and occupation area, $N = 511$, $N_S = 8$, and $N_L = \log_2((N+1)N_S) = 12$. In addition, $N_U = n = 12$, $N_{L2} = 6$, and $N_{out} = 18$ are set. Addition of $N_{L1} = 4$ can correct the residual error in the SAR-ADC mode. For digital re-quantization to reduce non-linearity in the SF-ADC mode, $N_{L3} = N_L - N_{L1} - N_{L2} = 2$ is used as the fractional part.

Both a foreground test to determine optimum digital thresholds ($V_{in,test}$, $D_{in,test}$, and D_{test} in Fig. 6) and machine learning to obtain calibration parameters ($V_{in,ideal}$ and $D_{in,ideal}$ in Fig. 6) are repeated until the conversion error is reduced below the allowable level. Note that the comparator number N is smaller than that in a previous report (2,047 in [18]) thanks to the oversampling and supervised machine learning technique.

Details of some calibration techniques for the SSA-ADC system are described in the following sections.

4. DAC Error Correction

The internal capacitor DAC in the SAR-ADC mode of the SSA-ADC has a differential configuration and is implemented using capacitors and MOS switches, as shown in Fig. 8 [5]. A buffer is used to drive the input capacitance of the SF-ADC. The sample and hold function is also embedded in this circuit. In sampling, the switches are in the state connecting them to input terminals ($V_{in,p}$,$V_{in,n}$), as indicated by the dashed lines. In successive comparisons, the switches are in the state connecting them to $V_{REF,p}(D_i = 1)$

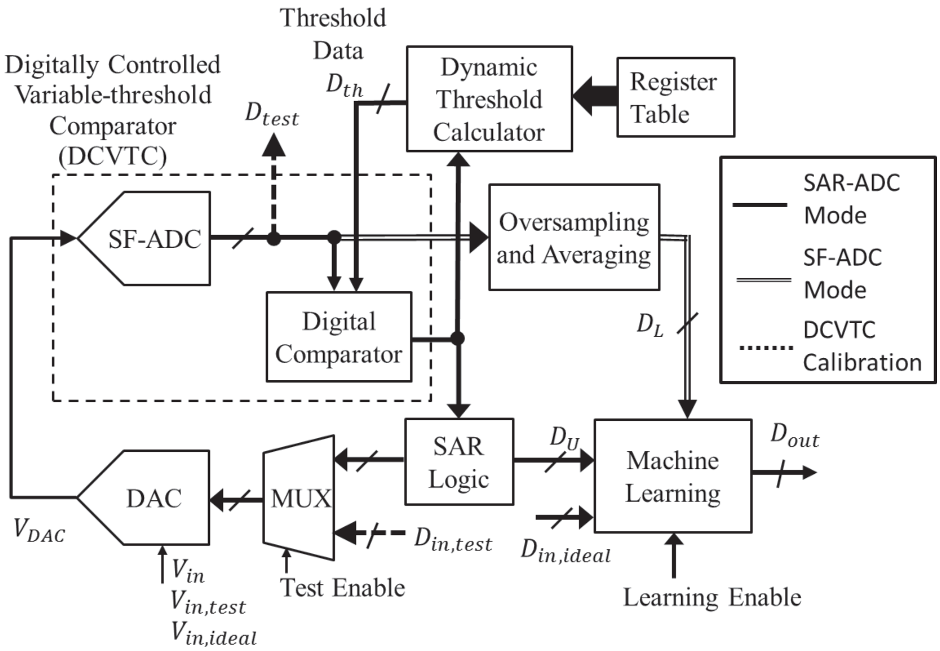


Fig. 6 System configuration of the SSA-ADC.

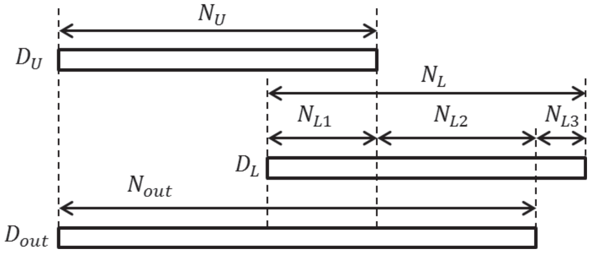


Fig. 7 Bit configuration in the SSA-ADC.

or $V_{REF,n}$ ($D_i = 0$) according to capacitor DAC input D_i ($i = 0, \ldots, n-1$), as indicated by the solid lines. The differential input voltage of the comparator (the SF-ADC block in the SSA-ADC) corresponds to the difference between the sampled value and the analog value generated from digital input D_i.

The positive-side (negative-side) capacitors $C_{i,p(n)}$ are weighted by powers of two with errors, as expressed by

$$C_{i,p(n)} = \begin{cases} 2^i C_u \left(1 + \varepsilon_i + s_{p(n)} \dfrac{\Delta\varepsilon_i}{2}\right), & i \leq d \\ 2^{i-1-d} C_u \left(1 + \varepsilon_i + s_{p(n)} \dfrac{\Delta\varepsilon_i}{2}\right), & i > d, \end{cases} \tag{4}$$

where $s_p = 1$, $s_n = -1$, C_u is unit capacitance, ε_i is the deviation from the ideal value of $(C_{i,p} + C_{i,n})/2$, and $\Delta\varepsilon_i$ is the difference between $C_{i,p}$ and $C_{i,n}$. Both ε_i and $\Delta\varepsilon_i$ originate from capacitance mismatch. Split capacitor $C_{C,p(n)}$, which can scale $C_{i,p}$ and $C_{i,n}$ to reduce the capacitance area, has a capacitance of C_u with some errors [19]. Here, $C_{p1,p(n)}$ and $C_{p2,p(n)}$ are parasitic capacitances, such as wiring capacitances. This section describes the DAC error from the capacitance mismatch and parasitic capacitances. The parasitic capacitance is assumed to be independent of applied voltage.

In the present study, focusing charge conservation on the floating node of the capacitances in sampling and successive comparison, $V_{DAC,p(n)}$ and $V_{DACsub,p(n)}$ in Fig. 8 were calculated for SAR-ADC numerical simulation. Considering the first-order terms of small errors, such as ε_i and $\Delta\varepsilon_i$, $V_{DAC}(= V_{DAC,p} - V_{DAC,n})$ in successive comparison is expressed as follows:

$$\begin{aligned} V_{DAC} \approx \sum_{i=0}^{n-1} 2^{i-d-1} \alpha C_u \Bigg[& \left(1 + \varepsilon_i + \beta\, u\left(d - i + \frac{1}{2}\right)\right) \\ & \times \{(2D_i - 1)V_{REF} - V_{in}\} \\ & + \left(\Delta\varepsilon_i + \frac{\Delta\alpha}{\alpha} + \Delta\beta\, u\left(d - i + \frac{1}{2}\right)\right) \\ & \times (V_{CM} + \Delta V_{in,REF})\Bigg] \\ & + \alpha C_{p,eff,0}\left(\varepsilon_{p,eff} + \frac{\Delta\alpha}{\alpha}\right) V_{CM}, \end{aligned} \tag{5}$$

where $u(x)$ ($u(x) = 0$ ($x < 0$), 1 ($x \geq 0$)) is the unit step function, and $V_{REF} = V_{REF,p} - V_{REF,n}$. The remaining parameters are given by

$$\Delta V_{in,REF} = \frac{V_{REF,p} + V_{REF,n} - V_{in,p} - V_{in,n}}{2}, \tag{6}$$

$$\alpha + s_{p(n)} \frac{\Delta\alpha}{2} = \frac{C_{t1,p(n)} + C_{C,p(n)}}{C_{C,p(n)} C_{tot,p(n)} + C_{t1,p(n)} C_{t2,p(n)}}, \tag{7}$$

$$\beta + s_{p(n)} \frac{\Delta\beta}{2} = \frac{2^{d+1} C_{C,p(n)}}{C_{t1,p(n)} + C_{C,p(n)}} - 1, \tag{8}$$

$$C_{t1,p(n)} = \sum_{i=0}^{d} C_{i,p(n)} + C_{p2,p(n)}, \tag{9}$$

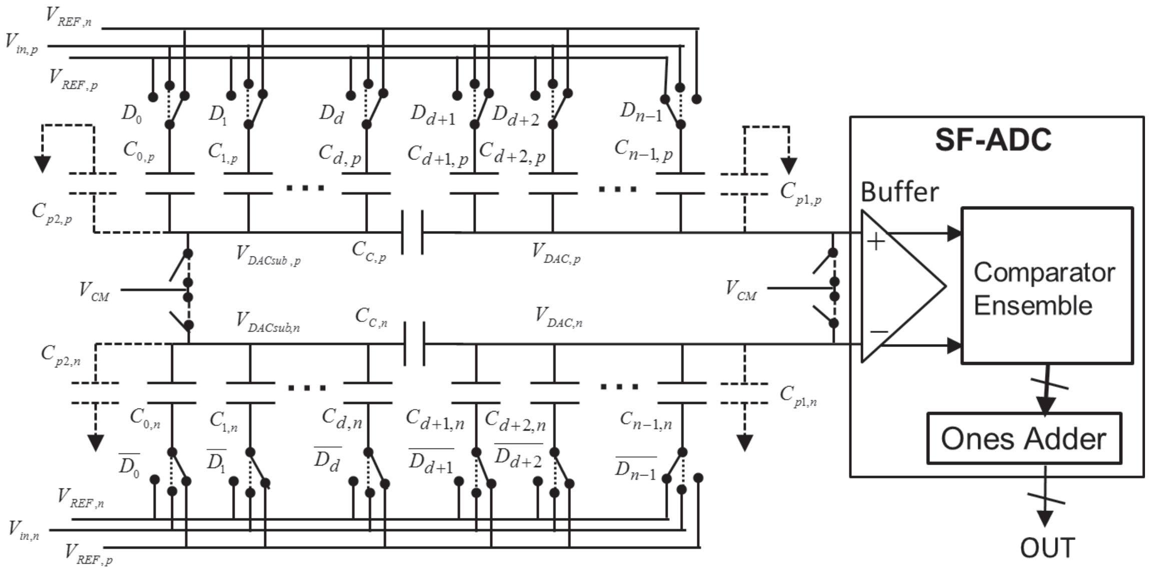


Fig. 8 n-bit capacitor DAC with an SF-ADC block.

$$C_{t2,p(n)} = \sum_{i=d+1}^{n-1} C_{i,p(n)} + C_{p1,p(n)}, \tag{10}$$

$$C_{tot,p(n)} = C_{t1,p(n)} + C_{t2,p(n)}, \tag{11}$$

$$\begin{aligned} &C_{p,eff,0}\left(1 + s_{p(n)}\frac{\varepsilon_{p,eff}}{2}\right) \\ &\quad = C_{p1,p(n)} + \frac{C_{p2,p(n)}C_{C,p(n)}}{C_{t1,p(n)} + C_{C,p(n)}}. \end{aligned} \tag{12}$$

In the ideal case without capacitance mismatch and parasitic capacitances, $\alpha = 2^{d+1}/(2^n - 1)$, $\beta = 0$, and V_{DAC} is given by

$$V_{DAC,ideal} = \sum_{i=0}^{n-1} \frac{2^i}{2^n - 1} C_u\{(2D_i - 1)V_{REF} - V_{in}\}. \tag{13}$$

In the above analysis, capacitance mismatches between the positive and negative sides influence the effective comparator offset for each successive comparison, which corresponds to the terms related to V_{CM} and $\Delta V_{in,REF}$ in the right-hand side of Eq. (5). This causes incorrect DAC code selection and residual error in the last SAR-ADC operation over the SF-ADC input range. In addition, nonlinearity of the capacitance (capacitor voltage coefficient) must be considered for high DAC linearity in the practical case [20], [21]. This dynamically influences capacitance mismatches between both sides according to V_{in} and D_i.

In order to overcome such complicated capacitance mismatches, dynamic compensation for the effective comparator offset is proposed in the present study, assuming that the error of capacitor DAC ΔV_{DAC} ($= V_{DAC} - V_{DAC,ideal}$) depends on capacitance selection according to the digital input $D_{in} = (D_{n-1}, \cdots, D_1, D_0)_2$. The dependence of the DAC error on the digital input D_{in} can be conceptually expressed as

$$\Delta V_{DAC}(D_{in}) = E_{off} + \sum_{i=0}^{n-1} E_i(D_i), \tag{14}$$

where E_{off} is the error that is independent of the input code, and $E_i (i = 0, \cdots, n-1)$ is the error that is dependent only on the input code D_i. The system configuration shown in Fig. 6 can correct errors expressed in Eq. (14) by determining the optimal digital threshold D_{th} of the SF-ADC for each conversion step. A multiplexer inserted at the input terminal of the DAC replaces the input with test data $D_{in,test}$ in a test mode (DCVTC calibration mode indicated by dotted lines in Fig. 6), where $D_{in,test}$ and $V_{in,test}$ are obtained from external components as D_{in} and V_{in}, respectively. Setting $V_{in,test} = V_{dac,ideal}(D_{in,test})$ according to $D_{in,test}$ generates the DAC error ΔV_{DAC} at the input terminal of the comparator. Since the SF-ADC operates as a DCVTC in this system, the SF-ADC output corresponds to the optimal digital threshold D_{th} for a given $D_{in,test}$. By using each value of the determined D_{th} dynamically in the SAR-ADC mode, the DAC error can be reduced. Since the error factors from ε_i in the right-hand side of Eq. (5) can be easily corrected via machine learning, as described later herein, the target factors for DAC error correction are capacitance mismatches between the positive and negative sides, which prevent incorrect DAC code selection. As a natural outcome, this correction process can reduce the offset in a comparator function realized by the SF-ADC in the SAR-ADC mode. The required precision for $V_{in,test}$ is related to the SF-ADC input range. Although the required precision for $V_{in,test}$ influences D_{th}, error correction by machine learning can recover the precision in the final ADC output.

However, the number of optimal values of D_{th} for all cases of D_i ($i = 1, \cdots n-1$) is 2^{n-1}. Note that LSB D_0 is 0 for the first to the ($n-1$)-th conversion steps and is 1 in the last step. For example, in the case of $n = 12$, the number of optimal D_{th} values is 2,048. Considering registers prepar-

Table 2 Example of test input codes (n = 12).

$D_{in,std}$	0000_0000_0000
$D_{in,11}$	1000_0000_0000
$D_{in,10}$	0100_0000_0000
$\vdots$	$\vdots$
$D_{in,2}$	0000_0000_0100
$D_{in,1}$	0000_0000_0010
$D_{in,0}$	0000_0000_0001

ing them and evaluation tests determining them, it is not practical. Therefore, an efficient technique is required for generating the optimal D_{th} dynamically according to D_{in}.

A technique to efficiently generate the optimal D_{th} is proposed based on Eq. (14). This technique uses $E_i(1) - E_i(0)$, which is detected more easily than $E_i(1)$ and $E_i(0)$. First, $\Delta V_{DAC}(D_{in,std})$ is detected using the standard test input code $D_{in,std}$. Although the standard test input code can be arbitrarily selected, $D_{in,std}$ is defined as the all zero code in the present study in order to simplify the control circuit, i.e., $D_i = 0, (i = 0, \cdots, n-1)$. In this case, $\Delta V_{DAC}(D_{in,std})$ is expressed as follows:

$$\Delta V_{DAC}(D_{in,std}) = E_{off} + \sum_{i=0}^{n-1} E_i(0). \tag{15}$$

Next, the specific test input code $D_{in,j}(j = 0, \cdots, n-1)$, which is defined as the D_{in} with $D_i = \delta_{i,j}$ (Kronecker's delta $\delta_{i,j} = 0 (i \neq j), = 1 (i = j)$), is considered. As an example, test input codes of $n = 12$ are shown in Table 2. The DAC error for $D_{in,j}$ ($\Delta V_{DAC}(D_{in,j})$) is expressed as

$$\Delta V_{DAC}(D_{in,j}) = \Delta V_{DAC}(D_{in,std}) + E_j(1) - E_j(0). \tag{16}$$

As described above, the error related to the j-th bit can be obtained and stored in registers as the difference from that for the standard test input code $D_{in,std}$.

The dynamic threshold calculator shown in Fig. 6 generates the optimal D_{th} in the following manner. First, D_{th} is set equal to $\Delta V_{dac}(D_{in,std})$. Then, when $D_j = 1$, $E_j(1) - E_j(0)$ is added to D_{th}. Otherwise, nothing is done. In this way, for all of the 2^{n-1} DAC input codes in the successive comparison, the DAC error correction can be executed only with one standard data $\Delta V_{dac}(D_{in,std})$ and all differences $E_i(1) - E_i(0)$. These data are stored in the register as configuration data and are read to generate optimal D_{th} dynamically at each conversion in the SAR-ADC mode, as shown in Fig. 6.

This dynamic digital threshold configuration can ensure the residual error in the last conversion step of the SAR-ADC mode within the input range of the SF-ADC block. Although the precision of the voltage source for $V_{in,test}$ influences D_{th}, error correction by machine learning can recover the precision in the final ADC output, as shown later in Sect. 6. The required voltage source accuracy is related to the SF-ADC input range. Noted that the DAC error correction is robust to the variation of the standard deviation of instantaneous comparator offset $\sigma_{off,eff}$ shown in Eq. (2).

5. Encoding and Error Correction by Machine Learning

In the SF-ADC mode, an output code D_L from the SF-ADC block through the averaging circuit expresses the probability corresponding to the residual error in the SAR-ADC mode. However, this code does not have the same scale as in the SAR-ADC mode because the SF-ADC uses comparator offset including IR noise as a reference, as described in Sect. 2. Hence, encoding D_L to binary code with the same scale as D_U obtained in the SAR-ADC mode is required in order to combine D_U and D_L codes for the total ADC code D_{out}. This encoding is the re-quantization that quantizes the output D_L from the averaging circuit in the digital domain. As an output range of the re-quantization can be determined after production, a programmable encoder is needed. The programmable encoder uses encoding parameters, i.e., quantization levels, that can be determined and registered after production. This encoding for re-quantization can improve the SNDR characteristics of the SF-ADC [22].

In a previous study [10], the encoding SF-ADC output was used to correct the error originating from the feedback DAC in multi-bit Δ-Σ ADC. Similarly, the encoding in the SSA-ADC can also correct the error from the SAR-ADC mode, which corresponds primarily to the term related to $(2D_i - 1)V_{REF} - V_{in}$ in Eq. (5). Here, N_L bits of D_L in Fig. 7 are prepared for this purpose. Note that the DAC error correction described in Sect. 4 targets primarily capacitance mismatches between the positive and negative sides, although target classification is not strict.

Thus, the SSA-ADC uses encoding with error correction for the SF-ADC characteristics and the non-ideality of the capacitor DAC, which remain even when using the DAC error correction described in Sect. 4. The key issue is how to determine parameters required in the encoding. Once these parameters are obtained, the encoding itself can be easily performed. In order to determine these parameters, machine learning, which is a kind of supervised learning based on error correction in a SAR-ADC [15], is introduced in the present study. Good parameter sets can predict calibrated data with small generalization errors (errors for unlearned data) by learning a small number of obtained data.

In learning mode (Learning Enable = 1 in Fig. 6), an analog voltage $V_{in,ideal}$ corresponding to an ideal ADC output code $D_{out,ideal}$ is input to the ADC. The voltage $V_{in,ideal}$ must be eligible to have precision of the target ADC resolution. As a result, MSB-side output D_U and LSB-side output D_L are obtained for the ideal code $D_{out,ideal}$. By repeating a similar process, an index set T for labels indicating a sufficient number of training data can be prepared. The k-th training data $D^{(k)} = (D^{(k)}_{out,ideal}, D^{(k)}_U, D^{(k)}_L), k \in T$ has the same bit configuration as in Eq. (3).

$$\begin{aligned} D^{(k)}_U &= (D^{(k)}_{U,N_U-1}, \ldots, D^{(k)}_{U,0}) \\ D^{(k)}_L &= (D^{(k)}_{L,N_L-1}, \ldots, D^{(k)}_{L,0}) \end{aligned} \tag{17}$$

$$D^{(k)}_{out,ideal} = (D^{(k)}_{out,ideal,N_{out}-1}, \ldots, D^{(k)}_{out,ideal,0})$$

In the present study, an ensemble of all sets of training data $D^{(k)}$ is denoted as D_T.

In order to predict the ideal code $D^{(k)}_{out,ideal}$ from the obtained $D^{(k)}_U$ and $D^{(k)}_L$ based on the training data ensemble D_T, the following correction function $h_{\boldsymbol{w}}(D^{(k)}_U, D^{(k)}_L)$ is used.

$$\begin{aligned} h_{\boldsymbol{w}}(D^{(k)}_U, D^{(k)}_L) =& \sum_{i=0}^{N_U-1} 2^{i+N_{L2}} D^{(k)}_{U,i} \left(1+e_i+\sum_{j=i+1}^{N_U-1} f_{i,j} D^{(k)}_{U,j}\right) \\ &+ \sum_{i=1}^{2^{N_{L1}+N_{L2}}-1} h_i p^{(k)}_i + \sum_{i=0}^{N_{L3}-1} 2^{i-N_{L3}} D^{(k)}_{L,i}(g_i+1) \\ &+ \delta, \end{aligned} \tag{18}$$

where $D^{(k)}_{U,i}$ denotes the i-th digit of $D^{(k)}_U$, and $p^{(k)}_i$ is the i-th bit of a $(2^{N_L-N_{L3}}-1)$-bit thermometer code, which corresponds to the binary code $(D^{(k)}_{L,N_L-1}, \ldots, D^{(k)}_{L,N_{L3}})$ or a decimal expression of $D^{(k)}_{L,D}$ given by

$$D^{(k)}_{L,D} = \sum_{i=N_{L3}}^{N_L-1} 2^{i-N_{L3}} D^{(k)}_{L,i}. \tag{19}$$

The first and second terms on the right-hand side of Eq. (18) are corrections for the SAR-ADC mode and re-quantized output in the SF-ADC mode, respectively. The third term is contributed from the N_{L3} bits of the fractional part in the SF-ADC mode. Appendix A shows the derivation of Eq. (18). The parameters determined in the learning mode are $\boldsymbol{e}$, $\boldsymbol{f}$, $\boldsymbol{h}$, $\boldsymbol{g}$, and δ, which are denoted as a set of parameters $\boldsymbol{w}$ for the correction function $h_{\boldsymbol{w}}(D^{(k)}_U, D^{(k)}_L)$ and are defined as follows:

$$\boldsymbol{w} = (\boldsymbol{e}^{\mathsf{T}}, \boldsymbol{f}^{\mathsf{T}}, \boldsymbol{h}^{\mathsf{T}}, \boldsymbol{g}^{\mathsf{T}}, \delta)^{\mathsf{T}}. \tag{20}$$

$$\boldsymbol{e} = (e_0, \ldots, e_{N_U-1})^{\mathsf{T}}, \tag{21}$$

$$\begin{aligned} \boldsymbol{f} = (&f_{0,1}, \ldots, f_{0,N_U-1}, f_{1,2}, \ldots, f_{1,N_U-1}, \ldots, \\ &f_{N_U-3,N_U-2} \ldots, f_{N_U-2,N_U-1})^{\mathsf{T}}, \end{aligned} \tag{22}$$

$$\boldsymbol{h} = (h_1, \ldots, h_{2^{N_{L1}+N_{L2}}-1})^{\mathsf{T}}, \tag{23}$$

$$\boldsymbol{g} = (g_0, \ldots, g_{N_{L3}-1})^{\mathsf{T}}, \tag{24}$$

where the superscript T indicates transpose of a matrix. After determining the set of parameters $\boldsymbol{w}$, in conversion mode (Learning Enable = 0 in Fig. 6), corrected ADC output $D_{out} = (D_{out,N_{out}-1}, \ldots, D_{out,0})$ can be obtained from D_U and D_L for an analog input V_{in} by using $D_{out,D} = h_{\boldsymbol{w}}(D_U, D_L)$, where

$$D_{out,D} = \sum_{i=0}^{N_{out}-1} 2^i D_{out,i}. \tag{25}$$

The error of the correction function from the ideal code, $E_{\boldsymbol{w}}(D^{(k)})$, is defined by

$$E_{\boldsymbol{w}}(D^{(k)}) = \left| D^{(k)}_{out,ideal,D} - h_{\boldsymbol{w}}(D^{(k)}_U, D^{(k)}_L) \right|, \tag{26}$$

$$D^{(k)}_{out,ideal,D} = \sum_{i=0}^{N_{out}-1} 2^i D^{(k)}_{out,ideal,i}, \tag{27}$$

and the set of parameters $\boldsymbol{w}$ can be determined by solving the following minimization problem:

$$\min_{\boldsymbol{w}} \sum_{k \in T} E_{\boldsymbol{w}}(D^{(k)})^2 + c_{r1}\|\boldsymbol{e}\|_2^2 + c_{r1}\|\boldsymbol{f}\|_2^2 + c_{r2}\|\boldsymbol{g}\|_2^2 + c_{r3}\|\boldsymbol{h}\|_2^2, \tag{28}$$

where $\|\cdot\|_2$ is the Euclidean norm, c_{r1}, c_{r2}, and c_{r3} are regularization constants, and their related terms are regularization terms to avoid exceedingly large values of e_i, $f_{i,j}$, g_i, and h_i.

In the present study, the above minimization problem was solved by Bayesian linear regression [15], [23], as described in Appendix B. In this approach, only a set of parameters $\boldsymbol{e}$ and $\boldsymbol{f}$, which is related to D_U from the SAR-ADC mode, has a continuous degree of freedom. A set of parameters $\boldsymbol{h}$ and $\boldsymbol{g}$ related to D_L from the SF-ADC mode can be determined focusing on monotonic re-quantization characteristics for the residue error in the SAR-ADC mode. The computation for the minimization problem is based on matrix operations, as described in Appendix B. The advantage of this approach is its suitability for incremental learning. As shown in Fig. A. 1, from the stored matrices ($\boldsymbol{\mu}_{T,\ell-1}$ and $\boldsymbol{S}_{T,\ell-1} \in \mathfrak{R}^{\frac{1}{2}N_U(N_U+1)+N_{L3}+2^{N_L-N_{L3}}}$) and additional training data sets, the renewed parameters $\boldsymbol{w}$ are obtained as their mean $\boldsymbol{\mu_T}$ and variance-covariance matrix $\boldsymbol{S_T}$. The size of the stored matrices depend only on the bit configuration (N_U, N_L, N_{L3}) and not on the number of training data sets, which is also useful for practical application.

For practical calibration involving limited time and an easy-to-use test environment, test data can be incomplete or missing. The validity and efficiency of complete-data-based methods cannot be guaranteed in such a case. Although a multiple-imputation method based on the expectation maximization was proposed in order to supplement the circuit calibration and to guide the verification process with the information obtained through the monitoring process, this method requires die-level process monitoring circuits [24]. The calibration techniques used in the present study do not require die-level process monitoring. In addition, compared to a simple digital-domain calibration technique [5], the proposed technique can realize more complicated error correction.

6. Behavior-Level Simulation Results

In order to verify the effectiveness of the proposed method, behavior-level simulations were carried out using MATLAB, where the full-scale voltage is $V_{FS} = 0.5$ V and the final total precision of the AD conversion is 18-bit (1 LSB is approximately 1.91 μV). Although the present study considers the multi-channel AFE shown in Fig. 1, we herein focus on one-channel operation with the highest conversion speed. The oversampling ratio of the SSA-ADC is $OSR = 500$. In the present study, $n = 12$, $d = 5$, and $V_{REF} = V_{CM} = 0.25$ V in Fig. 8. Since the present study was conducted prior to transistor-level simulation, parasitic capacitances were not considered in the simulation. The capacitance elements in

Table 3 Simulation conditions and parameters for the 18-bit ADC.

Sampling Frequency	500 kHz
Signal Bandwidth	500 Hz
Input Signal Frequency	164.99 Hz
Input Differential Amplitude	0.5 V
N	511
N_S	8
Full Scale Range	0.5 V
Buffer Gain	2
C_u	1.2 pF
δ_{C_u}/C_u	0.003
Comparator Input Offset σ_{off}	3 mV (standard deviation)
Comparator IR Noise	120 $\mu\mathrm{V_{rms}}$
Buffer Output-referred Noise	100 $\mu\mathrm{V_{rms}}$
Comparator Input Dead Zone	± 60 μV
Number of Training Data Sets	14,000

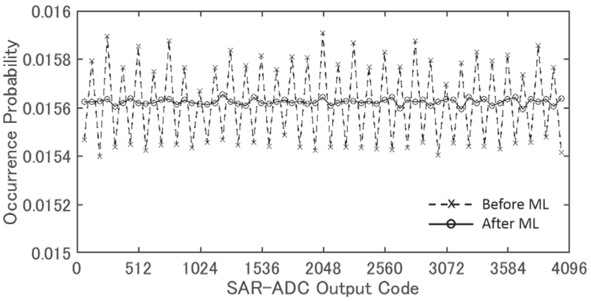


Fig. 9 Distribution of D_U (output in the SAR-ADC mode) before machine learning (ML) and upper 12-bit of corrected code D_{out} after ML for a full-range ramp analog input. (The sums of occurrence probabilities for 64 continuous D_U codes are plotted for clarity.)

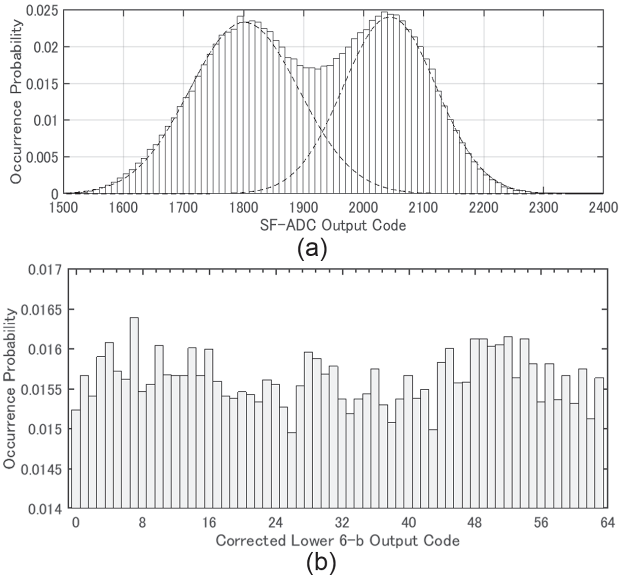


Fig. 10 Distributions of (a) D_L (output in the SF-ADC mode) before machine learning (ML) with fitted Gaussian curves and (b) corrected lower 6-bit output code after ML for a full-range ramp analog input.

Fig. 8 are assumed to be implemented with parallel connection of unit capacitances C_u. The mismatches of unit capacitances are considered to be $\delta_{C_u}/C_u = 0.003$, where δ_{C_u} is the standard deviation of capacitances C_u. This value is smaller than that used in a previous study on a 15-bit SAR-ADC [15] because of the larger unit capacitance in the present study ($C_u = 1.2$ pF). Although the dynamic digital threshold technique is used in this simulation, the non-linearity of capacitance was not considered because it depends on the details of the fabrication process [20]. The input offset standard deviation of the comparators in the SF-ADC is set to 3 mV, based on the assumption that offset calibration has been implemented. In the present study, the number of comparators is set to $N = 511$. The capacitor mismatch, comparator offset, and white noise are assumed to follow Gaussian distributions in the present study. The parameters used in the present study are listed in Table 3. The kT/C thermal noise generated at sampling was neglected because it is estimated to be 7.3 $\mu\mathrm{V_{rms}}$ and to be much smaller than the buffer IR noise (50 $\mu\mathrm{V_{rms}}$). The influence of this thermal noise in the learning mode on the tuning parameters $\boldsymbol{w}$ was relaxed by Bayesian linear regression [15], [23], as described in Appendix B. Moreover, the OSR oversampling and low-pass filtering described later herein can reduce the influence of thermal noise. Flicker noise in the comparators was not considered in the present study because of a lack of practical transistor-level design, as described above.

In order to determine the parameters of $\boldsymbol{w}$, initial learning with 2,000 training data sets and incremental learning steps was carried out. The number of training data sets per incremental learning was 800. Two-step selection of training data was adopted for 800 regions of (D_U, D_L), as described in detail in Appendix B. The first step selects 10 sets of training data per region at random, and the second step extracts one of set of training data per region based on error statistics, resulting in 800 sets of training data for one incremental learning iteration. The default number of incremental learning steps was 15, which means learning a total of 14,000 sets of training data (only 5.3% of 18-bit output code patterns).

In order to clarify the generalization error for the proposed SSA-ADC, transient simulation for full-range ramp analog input was carried out, and the uniformity of the occurrence probability was investigated. Figure 9 shows the distribution of D_U, which is 12-bit output in the SAR-ADC mode, and the upper 12-bit codes of D_{out} were corrected according to the machine learning results. Calibration of machine learning can significantly improve linearity. Figure 10 shows the distribution of D_L, which is 12-bit output in the SF-ADC mode, and the lower 6-bit codes of D_{out} obtained by machine learning. The original distribution of D_L is composed of two Gaussian distributions, each mean of which corresponds to typical residual errors at the last conversion of the SAR-ADC mode. By machine learning, the linearity of the lower 6-bit codes of D_{out} can be significantly improved. In simulation of both training and evaluating the proposed ADC, the white noise described above is taken into account. Therefore, there are inevitable errors even in reproduction of the same learned analog input. Fortunately, these errors can be significantly reduced by low-pass filtering, as demonstrated later herein.

Figure 11 shows the conversion error from the ideal code for the last 800 training data, which were obtained in the learning mode. Using Eq. (26), the error is defined as

$$e_{\boldsymbol{w}} = \frac{1}{|T|}\sum_{k\in T} E_{\boldsymbol{w}}(D^{(k)}), \tag{29}$$

where $|T|$ is the number of training data sets. Machine learning reduces the conversion error to approximately ±30 LSB, which corresponds to approximately the IR noise level of the buffer ($\pm 50\,\mu\mathrm{V}_{rms}$). In order to evaluate the effect of the dynamic digital threshold technique, the same simulation was carried out with and without this technique. As shown in Fig. 11, the dynamic digital threshold technique has little effect. This is attributed to the significant error compensation by the machine learning. However, the dynamic digital threshold technique is still useful to ensure that the residual error in the last conversion step of the SAR-ADC mode is within the input range of the SF-ADC block.

The output code density test [25] was carried out for a full-range single-tone signal input (frequency: 164.99 Hz). In order to obtain static characteristics with 99% confidence at 0.2-bit accuracy, the number of sampling points was set to 10^{24}. Figures 12(a) and 12(b) show the differential nonlinearity (DNL) error in the output code calibrated by machine learning. Figure 12(c) shows the integral nonlinearity (INL) error of the calibrated output code, which is calculated using the end-point method. The large INL error originates from the reduced output range of the capacitor DAC with $C_{C,p(n)} = C_u$ shown in Fig. 8 [19]. When the code range is limited from 2^6 to $2^{18}-2^6-1$, which corresponds to 0.4998 V for $V_{FS} = 0.5$ V, the INL can be improved, as shown in Fig. 12(d).

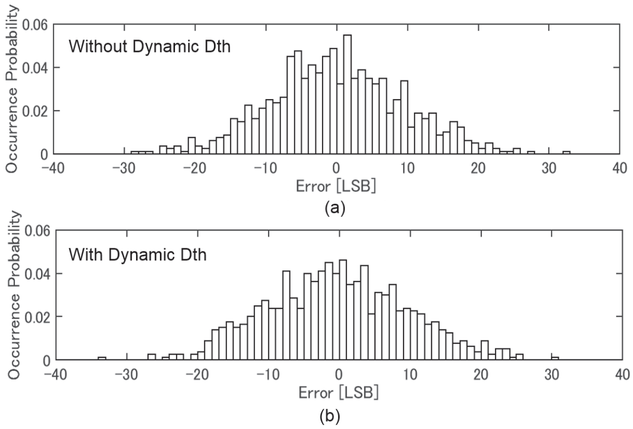


Fig. 11 Conversion error (a) without and (b) with the dynamic digital threshold technique for the last 800 sets of training data after incremental machine learning (total number of training data sets: 14,000).

In order to investigate the signal-to-noise and distortion ratio (SNDR) in the proposed SSA-ADC, transient simulation for full-range single-tone signal input (frequency: 164.99 Hz) was carried out with a sampling frequency of 500 kHz, which is oversampled ADC operation, as described in Sect. 1. Figure 13 shows FFT spectra of the ADC outputs corrected by machine learning with and without low-pass filtering (bandwidth: 500 Hz). As a part of the digital signal processing block shown in Fig. 1, the sixth-order Butterworth low-pass filter (LPF) was used for the filtering of the ADC outputs. This is because noise and error reduction by signal bandwidth elimination have a signifi-

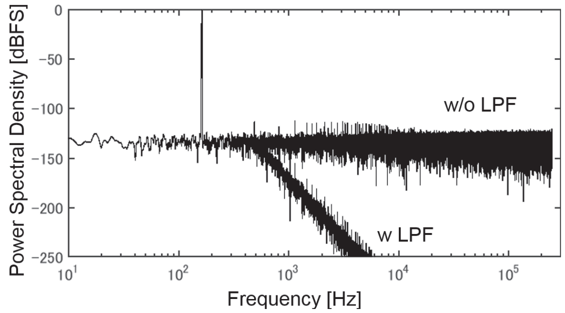


Fig. 13 Power spectral density for the SSA-ADC output with and without low-pass filtering.

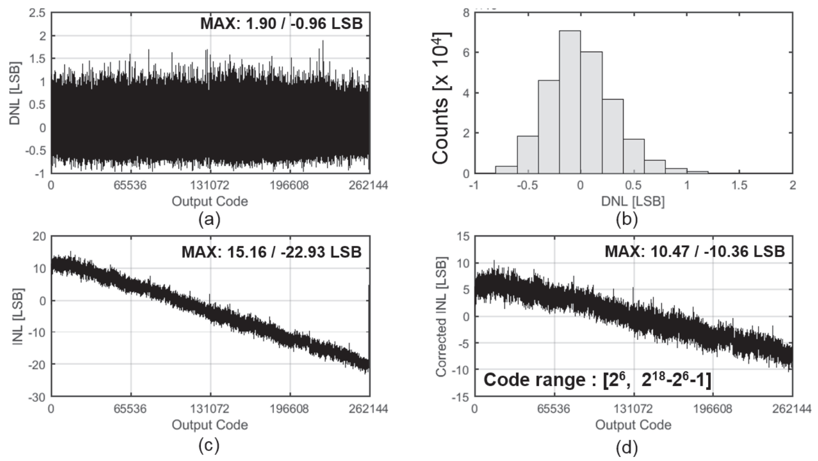


Fig. 12 (a) DNL error, (b) distribution of DNL error, (c) INL error, and (d) corrected INL error of the output code calibrated after incremental machine learning (total number of training data sets: 14,000).

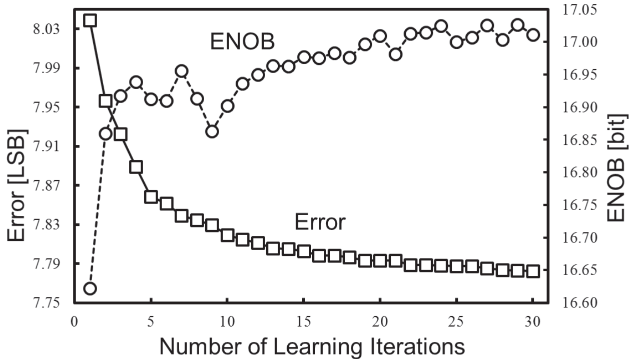


Fig. 14 Learning curve for the SSA-ADC.

cant effect on the SNDR. The number of sampling points was 2^{19} for accuracy. As shown in this figure, the filtering achieves an SNDR of 104.3dB and an effective number of bits (ENOB) of approximately 17.0 bits for a bandwidth of 500 Hz ($OSR = 500$). If N_S is doubled ($N_S = 16$) in order to enhance the ENOB, similar simulation results achieved an ENOB of 17.5 bits. Similarly, ENOB values of 18.0 and 18.4 bits are predicted for $N_S = 32$ and 64, respectively. Based on the stochastic signal detection [14], [17], doubling N_S can enhance ENOB by approximately 0.5 bits. Usually, the digital processing block shown in Fig. 1 decimates after filtering according to the signal bandwidth, i.e., down-sampling is performed for efficient data transfer. The signal down-sampled by 128 was also verified, and no significant degradation was observed.

Figure 14 shows the learning curve for the ENOB and the error obtained in the above simulations. The figure reveals that incremental learning beyond 15 times provides little improvement. In other words, a relatively small amount of training data can provide generalization.

7. Conclusions

In the present paper, we proposed an SSA-ADC for a novel low-power high-resolution AFE. In SAR-ADC mode for MSB-side bits, an SF-ADC block is used for the dynamic digital threshold technique. The SF-ADC block calibrates some error originating from the internal capacitor DAC to ensure that the residual error in the last conversion step of this mode is within the input range of the SF-ADC block. Configuration data for the dynamic digital threshold configurations were prepared preliminarily using a minimal amount of test input data and were used to generate the optimal threshold dynamically for each conversion in the SAR-ADC mode. These operations minimize the number of operations and the register size to store configuration data. For the residual error after SAR-ADC operation, which can be smaller than thermal noise, the SF-ADC uses the statistical characteristic of noise to obtain LSB-side bits. The SF-ADC output for the residual signal is combined with the SAR-ADC output to obtain high-precision ADC output data using a machine learning technique based on Bayesian linear regression. This technique compensates for capacitance mismatch and SF-ADC non-linearity. Behavior-level simulation results on a 18-bit SSA-ADC revealed its effectiveness.

Since the present feasibility study only provides behavior-level predictions, extensive future research on practical circuit implementation is necessary. However, we expect that the present study will establish new solutions to serious problems on ADCs in advanced device technologies.

Acknowledgments

The present study was supported by the Adaptable and Seamless Technology Transfer Program through Target-driven R&D (A-STEP) of the Japan Science and Technology Agency (JST). The authors would also like to thank Proassist, Ltd. for providing technical support.

References

[1] P. Schönle, F. Schulthess, S. Fateh, R. Ulrich, F. Huang, T. Burger, and Q. Huang, "A DC-Connectable Multi-Channel Biomedical Data Acquisition ASIC with Mains Frequency Cancellation," 2013 Proc. ESSCIRC (ESSCIRC), pp.149–152, 2013.

[2] S. Fateh, P. Schonle, L. Bettini, G. Rovere, L. Benini, and Q. Huang, "A Reconfigurable 5-to-14 bit SAR ADC for Battery-Powered Medical Instrumentation," IEEE Trans. Circuit Syst. I, vol.62, no.11, pp.2685–2694, Nov. 2015.

[3] P. Harpe, E. Cantatore, and A. van Roermund, "An oversampled 12/14b SAR ADC with noise reduction and linearity enhancements achieving up to 79.1 dB SNDR," IEEE International Solid-State Circuits Conference, pp.194–195, 2014.

[4] K. Uyttenhove and M.S. J. Steyaert, "Speed-power-accuracy tradeoff in high-speed CMOS ADCs," IEEE Trans. Circuits Syst. II, vol.49, no.4, pp.280–287, April 2002.

[5] J.-Y. Um, Y.-J. Kim, E.-W. Song, J.-Y. Sim, and H.-J. Park, "A digital-domain calibration of split-capacitor DAC for a differential SAR ADC without additional analog circuits," IEEE Trans. Circuits Syst. I, vol.60, no.11, pp.2845–2856, Nov. 2013.

[6] T. Sundström and A. Alvandpour, "Utilizing process variations for reference generation in a flash ADC," IEEE Trans. Circuits Syst. II, vol.56, no.5, pp.364–368, May 2009.

[7] S. Weaver, B. Hershberg, P. Kurahashi, D. Knierim, and U.K. Moon, "Stochastic flash analog-to-digital conversion," IEEE Trans. Circuits Syst. I, vol.57, no.11, pp.2825–2833, Nov. 2010.

[8] H. Ham, T. Matsuoka, J. Wang, and K. Taniguchi, "Design of a 500-MS/s stochastic signal detection circuit using a non-linearity reduction technique in a 65-nm CMOS process," IEICE Electronics Express, vol.8, no.6, pp.353–359, March 2011.

[9] H. Ham, T. Matsuoka, J. Wang, and K. Taniguchi, "Design of a high-speed-sampling stochastic flash analog-to-digital converter utilizing device mismatch," Electronics and Communications in Japan, vol.96, no.1, pp.51–62, Jan. 2013.

[10] Y. Hirai, S. Yano, and T. Matsuoka, "A delta-sigma ADC with stochastic quantization," IPSJ Trans. System LSI Design Methodology, vol.8, pp.123–130, Aug. 2015.

[11] J.J. Collins, C.C. Chow, and T.T. Imhoff, "Stochastic resonance without tuning," Nature, vol.376, no.6537, pp.236–238, July 1995.

[12] P.C. Gailey, A. Neiman, J.J. Collins, and F. Moss, "Stochastic resonance in ensembles of nondynamical elements: The role of internal noise," Phys. Rev. Lett., vol.79, no.23, pp.4701–4704, Dec. 1997.

[13] P. Carbone and M. Caciotta, "Stochastic-flash analog-to-digital conversion," IEEE Trans. Instrum. Meas., vol.47, no.1, pp.65–68, Feb. 1998.

[14] H. Ham, T. Matsuoka, and K. Taniguchi, "Application of noise-enhanced detection of subthreshold signals for communication systems," IEICE Trans. Fundamentals, vol.E92-A, no.4, pp.1012–1018, April 2009.

[15] T. Kurata, K. Tatsumi, T. Tanino, Y. Hirai, T. Matsuoka, and S. Tani, "Incremental learning for a calibration of a high-precision SAR-ADC by using the Bayesian linear regression," ISCIE Trans. Institute of Systems, Control and Information Engineers, vol.29, no.2, pp.76–85, Feb. 2016 (in Japanese).

[16] K. Tatsumi, T. Matsuoka, and S. Tani, "Incremental learning for a calibration of a high-precision SAR-ADC by using the inverse calibration and Bayesian regression," ISCIE Trans. Institute of Systems, Control and Information Engineers, vol.29, no.3, pp.136–142, March 2016 (in Japanese).

[17] T. Matsuoka, J. Wang, T. Kihara, H. Ham, and K. Taniguchi, "Low-voltage wireless analog CMOS circuits toward 0.5 V operation," IEICE Trans. Fundamentals, vol.E93-A, no.2, pp.356–366, Feb. 2010.

[18] S. Weaver, B. Hershberg, and U.K. Moon, "Digitally synthesized stochastic flash ADC using only standard digital cells," IEEE Trans. Circuits Syst. I, vol.61, no.1, pp.84–91, Jan. 2014.

[19] A. Agnes, E. Bonizzoni, P. Malcovati, and F. Maloberti, "A 9.4-ENOB 1V 3.8 μW 100kS/s SAR ADC with time-domain comparator," IEEE International Solid-State Circuits Conference, pp.246–247, 2008.

[20] S. Haenzsche, S. Henker, and R. Schüffny, "Modelling of capacitor mismatch and non-linearity effects in charge redistribution SAR ADCs," Mixed Design of Integrated Circuits and Systems, pp.300–305, 2010.

[21] A. Bannon, C.P. Hurrell, D. Hummerston, and C. Lyden, "An 18 b 5 MS/s SAR ADC with 100.2 dB Dynamic Range," IEEE Symosium on VLSI Circuits, pp.1–2, 2014.

[22] T. Asano, Y. Hirai, S. Tani, S. Yano, I. Jo, and T. Matsuoka, "An offset distribution modification technique of stochastic flash ADC," IEICE Electronics Express, vol.13, no.6, p.20160115, March 2016.

[23] C.M. Bishop, Pattern recognition and machine learning, Springer, pp.1–177, 2006.

[24] A. Zjajo, M.J. Barragan, and J.P. de Gyvez, "Digital adaptive calibration of multi-step analog to digital converters," Journal of Low Power Electronics, vol.8, no.2, pp.182–196, April 2012.

[25] J. Doernberg, H.-S. Lee, and D.A. Hodges, "Full-Speed Testing of A/D Converters," IEEE J. Solid-State Circuits, vol.19, no.6, pp.820–827, Dec. 1984.

Appendix A: Derivation of Eq. (18)

Based on physical phenomena in the SAR-ADC mode, which is expressed by Eq. (5), error correction for the output in the SAR-ADC mode is considered. At the last conversion step of the SAR-ADC mode, all digital inputs D_i are determined. Since $V_{DAC} \approx 0$ at this time, $V_{in}(= V_{in,p} - V_{in,n})$ can be given by

$$\begin{aligned}V_{in} \approx\; & \frac{2^{d+1}}{2^n - 1 + \sum_{i=0}^{n-1} \varepsilon_i + (2^{d+1} - 1)\beta} \\ & \times \left[\sum_{i=0}^{n-1} 2^{i-d} \left\{ \left(\left(1 + \varepsilon_i + \beta\, u \left(d - i + \frac{1}{2} \right) \right) \right) D_i V_{REF} \right.\right. \\ & - \left(\varepsilon_i + \beta\, u \left(d - i + \frac{1}{2} \right) \right) \frac{V_{REF}}{2} \\ & + \frac{1}{2} \left(\Delta\varepsilon_i + \frac{\Delta\alpha}{\alpha} + \Delta\beta\, u \left(d - i + \frac{1}{2} \right) \right) \\ & \times (V_{CM} + \Delta V_{in,REF}) \Big\} - \frac{2^n - 1}{2^{d+1}} V_{REF} \\ & \left. + \frac{C_{p,eff,0}}{C_u} \left(\varepsilon_{p,eff} + \frac{\Delta\alpha}{\alpha} \right) V_{CM} \right]. \qquad \text{(A· 1)}\end{aligned}$$

In the ideal case without capacitance errors or parasitic capacitances, $\alpha = 2^{d+1}/(2^n - 1)$, $\beta = 0$, and input voltage V_{in} can be given by

$$V_{in} \approx \left(\frac{2}{2^n - 1} \sum_{i=0}^{n-1} 2^i D_i - 1 \right) V_{REF} \qquad \text{(A· 2)}$$

The factor of $2/(2^n - 1)$ is different from the ideal factor $1/2^{n-1}$, which originates from split capacitance $C_{C,p(n)}$ with unit capacitance C_u used from the viewpoint of fabrication error [19]. In the present study, this gain error can also be taken into account in error correction.

In order to calibrate the ADC, only ideal output code for experimentally obtained output code $D_{i,m}$ can be used as training code $D_{i,t}$. As such, Eq. (A· 1) is modified to obtain a simple model, as follows:

$$\begin{aligned}V_{in} &\approx \left(\frac{1}{2^{n-1}} \sum_{i=0}^{n-1} 2^i D_{i,t} - 1 \right) V_{REF} \\ &\approx \left[\frac{1}{2^{n-1}} \sum_{i=0}^{n-1} 2^i D_{i,m} \left(1 + e_i + \sum_{j=i+1}^{n-1} f_{i,j} D_{j,m} \right) - 1 \right] V_{REF},\end{aligned} \qquad \text{(A· 3)}$$

where e_i is related to the coefficient of D_i in Eq. (A· 1), and $f_{i,j}$ introduces other errors into the coefficient of $D_{i,m}$. In successive comparisons from the MSB code, errors generated in the upper bit selection influence errors in the lower bit. Thus, only a set of $f_{i,j}$ $(i < j)$ is used. In order to reduce the number of learned parameters, higher-order terms in $D_{j,m}$ were neglected. This simple model is used in the correction function of Eq. (18).

The second term of the correction function given by Eq. (18) originated from the non-linearity characteristics of the SF-ADC. In other words, each bit of the thermometer code of the SF-ADC output has a different weight. Moreover, $\boldsymbol{h}$ corresponds to the array of these weights. The average of the N_{L3} bits of the fractional part is taken into account as the third term in the correction function given by Eq. (18).

Appendix B: Minimization Problem Using Bayesian Linear Regression

Here, the minimum problem expressed in Eq. (28) can be solved easily. By extending the previous study [15], the optimal values of the tuning parameters ($\boldsymbol{w}^*$) for the correction function expressed by Eq. (18) can be obtained analytically as follows:

$$\boldsymbol{w}^* = \left(\boldsymbol{\Phi}_T^{\mathsf{T}} \boldsymbol{\Phi}_T + \boldsymbol{I}_c \right)^{-1} \boldsymbol{\Phi}_T^{\mathsf{T}} \boldsymbol{d}_T, \qquad \text{(A· 4)}$$

where

$$\begin{aligned}
&\boldsymbol{\phi}(D_U, D_L)\\
&\quad = \Big(2^{N_{L2}} D_{U,0}, \ldots, 2^{N_{out}-1} D_{U,N_U-1},\\
&\qquad 2^{N_{L2}} D_{U,0} D_{U,1}, \ldots,\ 2^{N_{L2}} D_{U,0} D_{U,N_U-1},\\
&\qquad 2^{N_{L2+1}} D_{U,1} D_{U,2}, \ldots,\ 2^{N_{out}-1} D_{U,N_U-2} D_{U,N_U-1},\\
&\qquad p, 2^{N_{L3}-1} D_{L,N_{L3}-1}, \ldots, 2^0 D_{L,0}, 1\Big)^{\mathsf{T}} \qquad (\mathrm{A\cdot 5})
\end{aligned}$$

$$\boldsymbol{\Phi}_T = (\boldsymbol{\phi}(D_U^{(1)}, D_L^{(1)}), \ldots, \boldsymbol{\phi}(D_U^{(K)}, D_L^{(K)}))^{\mathsf{T}}, \qquad (\mathrm{A\cdot 6})$$

$$\begin{aligned}
\boldsymbol{I}_c &= \mathrm{diag}\,\{c_{r1}, \ldots, c_{r1}, c_{r3}, \ldots, c_{r3}, c_{r2}, \ldots, c_{r2}, 0\}\\
&\in \Re^{\frac{1}{2} N_U (N_U+1) + N_{L3} + 2^{N_L - N_{L3}}}, \qquad (\mathrm{A\cdot 7})
\end{aligned}$$

$$\boldsymbol{d}_T = (D_{out,ideal,D}^{(1)} - D_{U,D}^{(1)}, \ldots, D_{out,ideal,D}^{(K)} - D_{U,D}^{(K)})^{\mathsf{T}}, \qquad (\mathrm{A\cdot 8})$$

$$D_{U,D}^{(k)} = \sum_{i=0}^{N_U-1} 2^{i+N_{L2}} D_{U,i}^{(k)}. \qquad (\mathrm{A\cdot 9})$$

Here, K is the number of the training data sets, and diag indicates the elements of the diagonal matrix $\boldsymbol{I}_c$, p is a $2^{N_L - N_{L3}}$-bit thermometer code, and $p^{(k)}$ corresponds to the binary code $(D_{L,N_L-1}^{(k)}, \ldots, D_{L,N_{L3}}^{(k)})$ or a decimal expression of $D_{L,D}^{(k)}$ given by Eq. (19). However, the reliability of the training data can be degraded due to thermal noise, insufficient test signal accuracy, and external disturbance. In order to relax this problem, Bayesian linear regression [15], [23] is used for the correction function $h_{\boldsymbol{w}}(D_U, D_L)$ in the present study.

The tuning parameters $\boldsymbol{w}$ in $h_{\boldsymbol{w}}(D_U, D_L)$ follow a Gaussian distribution $\mathcal{N}(\boldsymbol{w}|\, 0, \boldsymbol{\Sigma}_c)$ as prior probability distribution, where $\boldsymbol{\Sigma}_c^{-1} = \boldsymbol{I}_c$, and δ in Eq. (20) is assumed to follow a uniform distribution. In addition, the corrected data D_{out} is assumed to obey the Gaussian distribution $\mathcal{N}\left(D_{out,D}|\, h_{\boldsymbol{w}}(D_U, D_L), \beta_{out}^{-1}\right)$. The value of β_{out} determines the variance of the corrected data. The posterior probability of $\boldsymbol{w}$ in the case of obtaining the training data ensemble D_T, $p\,(\boldsymbol{w}|\, D_T)$, is most likely given by

$$p(\boldsymbol{w}|\, D_T) = \mathcal{N}(\boldsymbol{w}|\, \boldsymbol{\mu}_T, \boldsymbol{S}_T), \qquad (\mathrm{A\cdot 10})$$

$$\boldsymbol{\mu}_T = \beta_{out}\, \boldsymbol{S}_T\, \boldsymbol{\Phi}_T^{\mathsf{T}}\, \boldsymbol{d}_T, \qquad (\mathrm{A\cdot 11})$$

$$\boldsymbol{S}_T^{-1} = \boldsymbol{\Sigma}_c^{-1} + \beta_{out}\, \boldsymbol{\Phi}_T^{\mathsf{T}}\, \boldsymbol{\Phi}_T, \qquad (\mathrm{A\cdot 12})$$

where $\boldsymbol{\Phi}_T$ has $\boldsymbol{\phi}(D_U^{(k)}, D_L^{(k)})^{\mathsf{T}}$, $k \in T$ in its row, as shown in Eq. (A·6), and the vertical vector $\boldsymbol{d}_T$ has $D_{out,ideal,D}^{(k)} - D_{U,D}^{(k)}$, $k \in T$ in its element, as shown in Eq. (A·8). In this case, the expected probability distribution of D_{out} for a given set of D_U and D_L is given by

$$\begin{aligned}
&p\,(D_{out,D}|\, D_T)\\
&\quad = \mathcal{N}\left(D_{out,D}|\, \boldsymbol{\mu}_T^{\mathsf{T}} \boldsymbol{\phi}(D_U, D_L) + D_{U,D}, \sigma_{out}^2\right), \qquad (\mathrm{A\cdot 13})
\end{aligned}$$

$$\sigma_{out}^2 = \beta_{out}^{-1} + \boldsymbol{\phi}(D_U, D_L)^{\mathsf{T}} \boldsymbol{S}_T \boldsymbol{\phi}(D_U, D_L). \qquad (\mathrm{A\cdot 14})$$

The standard deviation σ_{out} can give the uncertainty of the expected average of $D_{out,D}$ for training data ensemble D_T and can be used as an indicator to select D_U and D_L in incremental learning to refine the correction function [15].

Figure A·1 shows the procedure for incremental learning, based on the previous work [15]. The index set T for labels indicating a training data set in the ensemble D_T is divided into $T_0, T_1, \ldots, T_{\ell_{max}}$ as $T = \cup_{k=0}^{\ell_{max}} T_k$. In Fig. A·1, $\boldsymbol{S}_{T_\ell}$ and $\boldsymbol{\mu}_{T_\ell}$ are given by Eqs. (A·12) and (A·11), respectively, for $T = \cup_{k=0}^{\ell} T_k$. As shown in Fig. A. 1, the calculations for them use the stored matrices ($\boldsymbol{\mu}_{T,\ell-1}$ and $\boldsymbol{S}_{T,\ell-1}$) and additional training data sets [15]. The size of the stored matrices depend only on the bit configuration (N_U, N_L, N_{L3}).

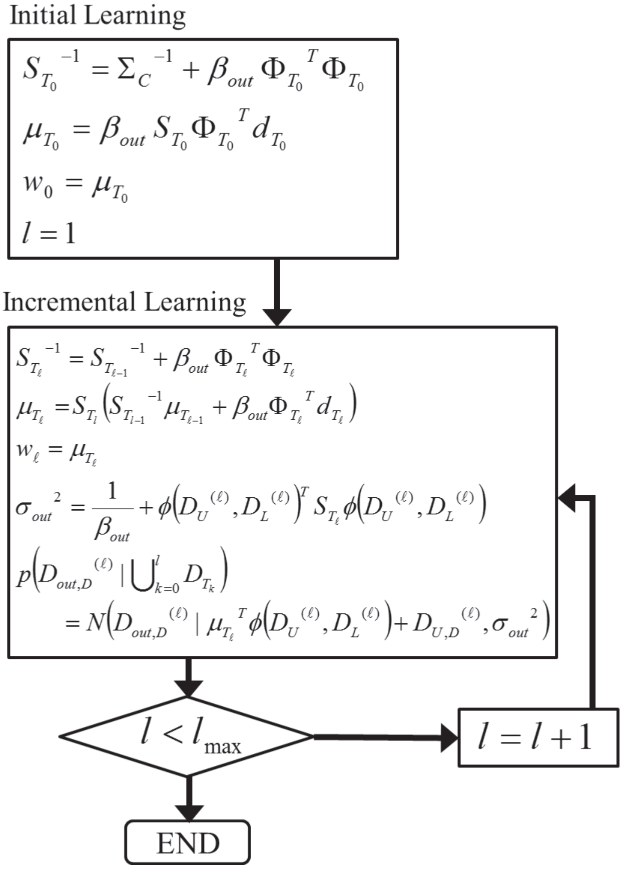

Initial Learning
$S_{T_0}{}^{-1} = \Sigma_C{}^{-1} + \beta_{out}\, \Phi_{T_0}{}^T \Phi_{T_0}$
$\mu_{T_0} = \beta_{out}\, S_{T_0} \Phi_{T_0}{}^T d_{T_0}$
$w_0 = \mu_{T_0}$
$l = 1$
Incremental Learning
$S_{T_\ell}{}^{-1} = S_{T_{\ell-1}}{}^{-1} + \beta_{out}\, \Phi_{T_\ell}{}^T \Phi_{T_\ell}$
$\mu_{T_\ell} = S_{T_l}\left(S_{T_{l-1}}{}^{-1} \mu_{T_{\ell-1}} + \beta_{out} \Phi_{T_\ell}{}^T d_{T_\ell}\right)$
$w_\ell = \mu_{T_\ell}$
$\sigma_{out}{}^2 = \dfrac{1}{\beta_{out}} + \phi\left(D_U{}^{(\ell)}, D_L{}^{(\ell)}\right)^T S_{T_\ell} \phi\left(D_U{}^{(\ell)}, D_L{}^{(\ell)}\right)$
$p\left(D_{out,D}{}^{(\ell)} \mid \bigcup_{k=0}^{l} D_{T_k}\right)$
$= N\left(D_{out,D}{}^{(\ell)} \mid \mu_{T_\ell}{}^T \phi\left(D_U{}^{(\ell)}, D_L{}^{(\ell)}\right) + D_{U,D}{}^{(\ell)}, \sigma_{out}{}^2\right)$
$l < l_{\max}$
$l = l + 1$
END

Fig. A·1 Procedure of incremental learning.

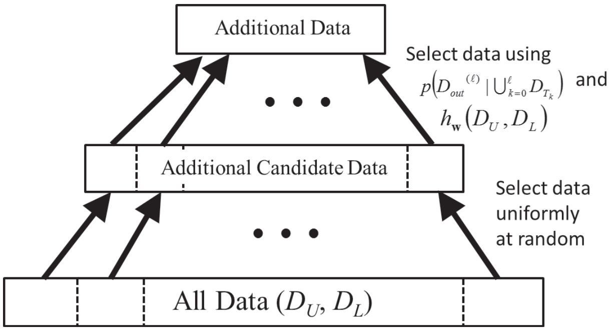

Additional Data
Select data using
$p\left(D_{out}{}^{(\ell)} \mid \cup_{k=0}^{\ell} D_{T_k}\right)$ and
$h_{\mathrm{w}}(D_U, D_L)$
Additional Candidate Data
Select data uniformly at random
All Data (D_U, D_L)

Fig. A·2 Selection flow of training data for incremental learning.

As a training data selection method, the two-step selection shown in Fig. A·2 is uniformly applied to training data with a large error. Note that D_U and D_L cannot be assigned directly due to noise and errors. By using the approximation $D_{out,ideal} \approx h_{\boldsymbol{w}}(D_U.D_L)$, adequate incremental training data sets can be obtained [16]. For the sake of simplicity, other selection techniques proposed in [16] are not used in the present study.

Sadahiro Tani received B.S. and M.S. degrees in electronic engineering from Osaka University, Osaka, Japan, in 1979 and 1981, respectively. In 2004 he received a Ph.D. degree in information systems engineering from Osaka University. During 1981–2012, he worked for Sharp Corporation, Nara, Japan. During 2014–2016, he was a project researcher at Osaka University. His current research includes CMOS mixed signal circuits. Dr. Tani is a member of the IEICE and the IEEE.

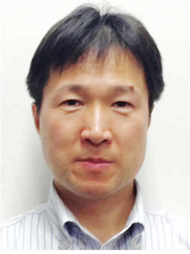
Toshimasa Matsuoka received B.S., M.S., and Ph.D. degrees in electronic engineering from Osaka University, Osaka, Japan, in 1989, 1991, and 1996, respectively. During 1991–1998, he worked at the Central Research Laboratories, Sharp Corporation, Nara, Japan, where he was engaged in research and development related to deep-submicron CMOS devices and ultra-thin gate oxides. Since 1999, he has been working at Osaka University, where he is an Associate Professor. His current research includes CMOS analog/RF circuits and device modeling. Dr. Matsuoka is a member of the Japan Society of Applied Physics, the IEICE, and the IEEJ.

Yusaku Hirai received B.S. and M.S. degrees in electronic engineering from Osaka University, Osaka, Japan, in 2013, and 2015, respectively. He is now working for THine Electronics Inc., where he is developing high-speed serial interfaces.

Toshifumi Kurata received an M.S. degree in electric engineering from Osaka University, Osaka, Japan, in 2015. He is now working for Yamaha Corporation.

Keiji Tatsumi received a B.S. degree in engineering from Kyoto University, Kyoto, Japan in 1993, and an M.S. degree in information science from Nara Institute of Science and Technology, Nara, Japan in 1995. In 2006, he received a Ph.D. degree in informatics from Kyoto University. Since 1998, he has been working at Osaka University, where he is an Associate Professor. His current research includes metaheuristics for global optimization and machine learning. Dr. Tatsumi is a member of the Operations Research Society of Japan, the ISCIE, and the SICE.

Tomohiro Asano received B.S. and M.S. degrees in electronic engineering from Osaka University, Osaka, Japan, in 2014 and 2016, respectively.



Masayuki Ueda received a B.S. degree in physics from Konan University, Hyogo, Japan in 1987. During 1987–2014, he worked for Renesas System Design Co., Ltd. (formerly NEC IC Microcomputer Systems, Ltd.), Japan, where he was engaged mainly in the design of analog circuits. Since 2014, he has been working at SPChange, LLC, Japan.

Takatsugu Kamata received a B.S. degree in electrical engineering from Tokyo City University in 1987. During 1987–2001, he worked for Motorola Inc., where he was engaged in the design of RF/IF ICs for cellular phones. During 2001–2010, he worked for RfStream Corp., where he carried out research and development related to TV tuner ICs. In 2010, he received a Ph.D. degree from Osaka University, Japan. Currently he is working for SPChange, LLC. Since 2014, he has been also a Special Appointed Associate Professor in Osaka University. His current research includes an implantable medical device, wireless transceiver and contact-less power circuit/system. Dr. Kamata is a member of the IEICE and IEEE.