IN SITU AUTOMATIC ANALOG CIRCUIT CALIBRATION AND OPTIMIZATION

A Dissertation

by

SANGHOON LEE

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Edgar Sánchez-Sinencio
Committee Members,	Sebastian Hoyos
	Jiang Hu
	Erick Moreno-Centeno
Head of Department,	Miroslav M. Begovic

December 2020

Major Subject: Electrical Engineering

ABSTRACT

As semiconductor technology scales down, the variations of active/passive device characteristics after fabrication are getting more and more significant. As a result, many circuits need more accuracy margin to meet minimum accuracy specifications over huge process-voltage-temperature (PVT) variations. Although, overdesigning a circuit is sometimes not a feasible option because of excessive accuracy margin that requires high power consumption and large area. Consequently, calibration/tuning circuits that can automatically detect and compensate the variations have been researched for analog circuits to make better trade-offs among accuracy, power consumption, and area.

The first part of this research shows that a newly proposed in situ calibration circuit for a current reference can relax the sharp trade-off between the temperature coefficient accuracy and the power consumption of the current reference. Prototype chips fabricated in a 180 nm CMOS technology generate 1 nA and achieve an average temperature coefficient of 289 ppm/°C and an average line sensitivity of 1.4 %/V with no help from a multiple-temperature trimming. Compared with other state-of-the-art current references that do not need a multiple-temperature trimming, the proposed circuit consumes at least 74% less power, while maintaining similar or higher accuracy.

The second part of this research proves that a newly proposed multidimensional in situ analog circuit optimization platform can optimize a Tow-Thomas bandpass biquad. Unlike conventional calibration/tuning approaches, which only handle one or two frequency-domain characteristics, the proposed platform optimizes the power consumption, frequency-, and time-domain characteristics of the biquad to make a better trade-off between the accuracy and the power consumption of the biquad. Simulation results show that this platform reduces the gain-bandwidth product of op-amps in the biquad by 80% while reducing the standard deviations of frequency- and time-domain characteristics by 82%. Measurement results of a prototype chip fabricated in a 180 nm CMOS technology also show that this platform can save maximum 71% of the power consumption of the biquad while the biquad maintains its frequency-domain characteristics: Q, ω_O, and the gain at

ω_O.

DEDICATION

To my parents, who have encouraged me in pursuing my dream since I was a kid.

To my wife, Narae Yoon, who is always next to me with love.

To my daughter, Julia Taehee Lee, who makes me smile.

ACKNOWLEDGMENTS

During the time I worked on my Ph.D. degree, I learned a lot more than I had expected thanks to my advisor, Dr. Edgar Sánchez-Sinencio, and my friends. Especially, Dr. Sánchez gave me an opportunity to have leadership in my research and taught me how to make an innovation with a strong initiative. Two projects I led were enjoyable journeys thanks to his support and patience. I believe that having a good attitude is as important as having an in-depth knowledge for my future career. Dr. Sánchez showed me what a good attitude is as a researcher and a teacher. I would like to appreciate it.

I also would like to thank my committee members: Dr. Sebastian Hoyos, Dr. Jiang Hu, and Dr. Erick Moreno-Centeno. Especially, Dr. Hu gave me many valuable comments while we were working together for an NSF project. In addition, I thank Ms. Ella Gallagher, who was always willing to help me with Dr. Sánchez.

Many people contributed to two projects in this dissertation. Stephen Heinrich-Barna and Keith Kunz initiated the current reference calibration project. Kyoohyun Noh thoroughly reviewed my manuscript for the current reference project. All circuit optimization team members (Congyin Shi, Jiafan Wang, Adriana Sanabria Borbon, and Hatem Osman) did not hesitate giving me good comments while we were working together as a team. Especially, Congyin Shi provided a die photograph and measurement results for our manuscript. I am grateful to all of them.

My life in College Station would have been less delightful without my good friends: Alfredo Costilla Reyes, Chulhyun Park, Haewoong Yang, Hyun-Myung Woo, Johan Estrada Lopez, Joseph Samy Riad, Kyoohyun Noh, Myung Seok Shim, Sangjin Han, Sangmin Kim, and Sungjoon Yoon. I will never forget the coffee chats and happy hours with you.

Last but not least, I would like to express my sincere gratitude to my family. From elementary school years to high school years, my parents gave me countless opportunities to learn more about science and engineering. Thanks to the support, my curiosity could grow, leading to my Ph.D. degree. My parents-in-law, who have a full amount of positive energy, always encouraged me with

the energy. My wife, Narae Yoon, was the biggest supporter of my Ph.D. degree. She is always in my heart. My precious daughter, Julia Taehee Lee, has been and always will be the joy of my life.

CONTRIBUTORS AND FUNDING SOURCES

NOMENCLATURE

%RMSE	Percent Root-Mean-Square Error
ADC	Analog-to-Digital Converter
ALU	Arithmetic Logic Unit
ATE	Automatic Test Equipment
BIST	Built-In Self-Test
BJT	Bipolar Junction Transistor
CTAT	Complementary to Absolute Temperature
CUO	Circuit Under Optimization
CUT	Circuit Under Test
DNL	Differential Nonlinearity
ESG	Excitation Signal Generator
FPGA	Field Programmable Gate Array
GA	Genetic Algorithm
GBW	Gain Bandwidth Product
HVRVT	High-Voltage Regular-Threshold
IC	Integrated Circuit
IDAC	Current Digital-to-Analog Converter
INL	Integral Non-Linearity
IoT	Internet of Things
LSB	Least Significant Bit
LTI	Linear Time-Invariant
MOSFET	Metal-Oxide-Semiconductor Field-Effect Transistor

MSB	Most Significant Bit
MVT	Medium-Threshold
NM	Nelder-Mead Method
NMOS	N-type Metal-Oxide-Semiconductor
ORA	Output Response Analyzer
PDF	Probability Density Function
PMOS	P-type Metal-Oxide-Semiconductor
PS	Pattern Search
PTAT	Propotional To Absolute Temperature
PVT	Process-Voltage-Temperature
RVT	Regular-Threshold
SA	Simulated Annealing
SNR	Signal-to-Noise Ratio
SS	Sensitivity Search
THD	Total Harmonic Distortion
ZTC	Zero Temperature Coefficient

TABLE OF CONTENTS

Page

x

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

As semiconductor technology scales down, the variations of circuit characteristics are getting more and more significant [1, 2]. To have both of a high yield and enough performance better than required specifications given by customers, the most intuitive approach is designing a circuit that has much better performance than the minimum requirement as Fig. 1.1 shows. In Fig. 1.1, larger z_1 and z_2 mean better performance. The location of an actual operating point of the designed circuit can be anywhere enclosed by the dotted line in Fig. 1.1 when there are process-voltage-temperature (PVT) variations. Since the design point has enough margins for z_1 and z_2 circuit characteristics, an actual operating point of the designed circuit can always be in the region of acceptable performance regardless of PVT variations. However, the overdesigned circuit normally requires sacrificing other circuit characteristics that are less important for a certain application. For example, the most important circuit characteristics of a high-precision voltage or current reference circuit are the temperature coefficient and the line sensitivity of the circuit. Consequently, a circuit designer sometimes sacrifices the area and the power consumption of the reference circuit to achieve a low temperature coefficient and a small line sensitivity regardless of PVT variations.

Since achieving excessive performance that has a huge margin is sometimes not feasible, many research have been conducted to make a better trade-off among circuit characteristics. One common approach for the purpose is utilizing an in situ automatic calibration/tuning circuit. Fig. 1.2(a) shows the benefit of the approach. When we place our design point at A, an actual operating point of the designed circuit (A') can be located outside of the region of acceptable performance due to PVT variations. However, the actual operating point can move to A thanks to the calibration/tuning circuit that evaluates the circuit characteristic z_1 and compensates the difference between the required z_1 value and the actual z_1 value. Therefore, we do not need to excessively sacrifice other circuit characteristics that are less important for a certain application because the large margin for z_1 is not needed anymore. This approach generally makes a better trade-off among circuit characteristics when the area and the power consumption overheads of the calibration/tuning circuit are
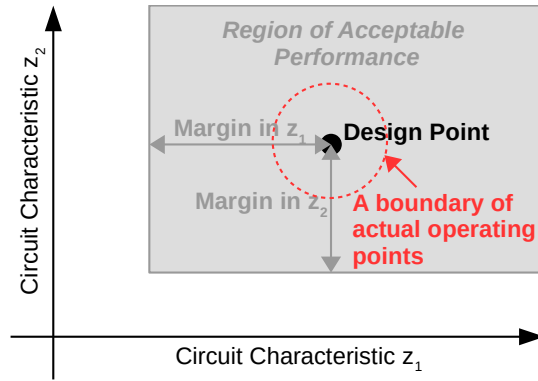
Figure 1.1: Design point with margins.
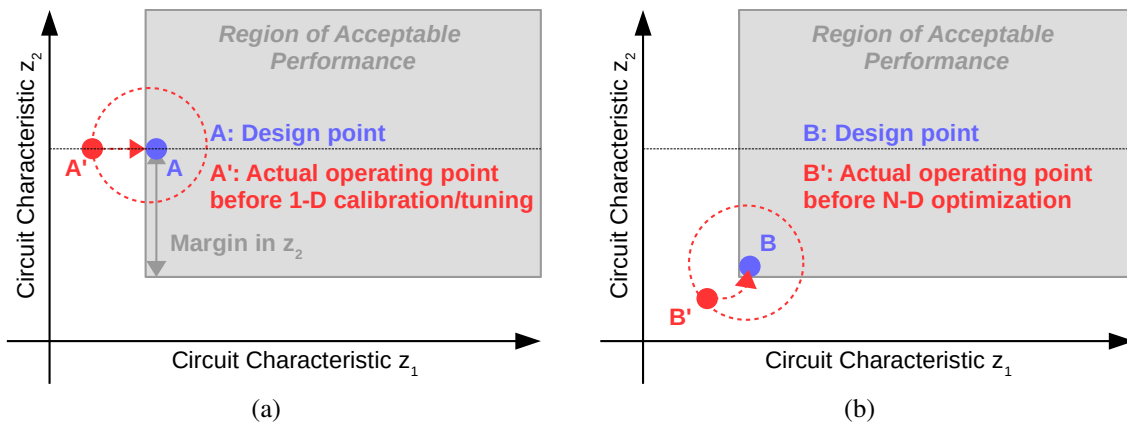
Figure 1.2: New design point when a 1-dimension calibration/tuning or a N-dimension optimization is employed. (a) 1-dimension calibration/tuning case. (b) N-dimension optimization case.

minimized.

Even though researchers have researched various calibration/tuning circuits for many analog circuits for a long time, the previous research have two limitations. First, there are still several analog circuits that have not been calibrated/tuned by an in situ automatic calibration/tuning circuit. For example, an in situ automatic calibration circuit for a current reference, to the best of the author's knowledge, has not been reported. Second, in many cases, calibration/tuning techniques that have been researched so far deal with only one circuit characteristic, leading to a suboptimal circuit. Fig. 1.2(a) shows the limitation clearly. The design point A still needs a large margin for

z_2 since the calibration/tuning circuit in the example only compensates the deviation of z_1. As a result, we still need to sacrifice less important circuit characteristics to some extent.

This dissertation investigates two approaches that overcome the aforementioned limitations. The first research [3] proposes an in situ automatic calibration circuit for an ultralow-power current reference. Since the calibration circuit detects a current generated from the ultralow-power current reference and compensates its deviations over PVT variations, the generated current can be accurate. The power consumption overhead of the calibration circuit is small because unnecessary circuits are powered off after each calibration. Therefore, the calibration circuit can relax the tight trade-off between the accuracy and the power consumption of the current reference.

The second research [4] proposes a built-in self-test and in situ analog circuit optimization platform. Since the platform optimizes multiple competing circuit characteristics simultaneously unlike conventional calibration/tuning circuits, margins for the multiple circuit characteristics can be greatly reduced. Consequently, a designer can choose the design point at the bottom left corner of the region of acceptable performance (*B*) in Fig. 1.2(b), resulting in a better trade-off among circuit characteristics. Turning on the platform only when it is needed can relax the power consumption overhead of the platform. We can mitigate the area overhead as well by utilizing the platform for multiple linear time-invariant (LTI) analog circuits on the same chip.

The rest of this dissertation is structured as follows. Chapter 2 introduces previous research on current reference circuits. Chapter 3 discusses the newly proposed in situ automatic calibration circuit for a current reference in detail. Previous research on in situ analog circuit optimization are summarized in Chapter 4. Chapter 5 elaborates the newly proposed built-in self-test and in situ analog circuit optimization platform. Finally, future works are discussed in Chapter 6.

2. A SURVEY ON CURRENT REFERENCE CIRCUITS

2.1 Background

2.1.1 Temperature-Dependent Physical Quantities

Since current generated from a current reference circuit is a function of physical quantities, understanding the temperature dependence of the physical quantities is essential to evaluate the temperature dependence of the generated current. There are five physical quantities that are widely used in current reference circuits to generate a stable current: the thermal voltage (V_T), the base-emitter voltage of a bipolar junction transistor (V_BE), the threshold voltage of a metal-oxide-semiconductor field-effect transistor (V_TH), the electron/hole mobility (μ), and the resistances of various resistors (R).

At room temperature, the thermal voltage, which equals to kT/q, is approximately 26 mV. The following equation shows the temperature coefficient of V_T.

$$V_{T,TC} = \frac{1}{V_T}\frac{\partial V_T}{\partial T} = \frac{1}{T}. \tag{2.1}$$

As (2.1) shows, the thermal voltage is a proportional-to-absolute-temperature (PTAT) quantity and has a temperature coefficient of 3333 ppm/°C at room temperature.

According to [5], the temperature variation of V_BE is given by

$$\frac{\partial V_{BE}}{\partial T} = \frac{V_{BE} - 2.5V_T - 1.12}{T}. \tag{2.2}$$

When V_BE equals to 600 mV, the temperature variation of V_BE is -2 mV/°C at room temperature, leading to a temperature coefficient of -3333 ppm/°C. Note that V_BE cannot be arbitrary small. If V_BE equals to 170 mV, the temperature variation of V_BE is -3.38 mV/°C at room temperature. If we ignore the temperature dependency of the temperature variation, V_BE will be close to 0 mV at 77 °C, which is not acceptable for many circuits.

The temperature variation of V_{TH} is also approximately -2 mV/°C [6]. Consequently, the temperature coefficient of V_{TH} is -6667 ppm/°C when V_{TH} equals to 300 mV at room temperature.

The temperature dependency of the electron/hole mobility can be expressed as

$$\mu = \mu_0 (T_0/T)^n. \tag{2.3}$$

In (2.3), T_0 is a reference temperature, and μ_0 is the mobility at the reference temperature, and n equals to 1.5. The temperature coefficient of the mobility can be written as

$$\mu_{TC} = -1.5/T. \tag{2.4}$$

As a result, μ_{TC} is -5000 ppm/°C at room temperature.

The temperature coefficient of the resistance of a resistor can be positive or negative depending on the type of the resistor. The temperature coefficient also depends on a process technology. In many cases, a resistor has a temperature coefficient of several hundreds or thousands ppm/°C.

2.1.2 Two Types of Computations to Get a Temperature-Independent Current

The physical quantities discussed in Section 2.1.1 have temperature coefficients much higher than 100 ppm/°C. Accordingly, it is hard to generate a stable current that has a temperature coefficient comparable to 100 ppm/°C by relying on only one physical quantity. More viable way is utilizing multiple physical quantities and allowing their temperature coefficients to cancel each other.

Two types of computations can cancel the temperature coefficients of physical quantities. The first type is addition/subtraction. If we assume that there are two physical quantities, A and B, then the generated current (I) from a current reference circuit can be

$$I = c_1 A \pm c_2 B. \tag{2.5}$$

In (2.5), c_1 and c_2 are temperature-independent constants. The temperature coefficient of I is given

by

$$I_{TC} = \frac{1}{I}\frac{\partial I}{\partial T} = \frac{c_1 A}{c_1 A \pm c_2 B}A_{TC} \pm \frac{c_2 B}{c_1 A \pm c_2 B}B_{TC}. \tag{2.6}$$

Here A_{TC} and B_{TC} are the temperature coefficients of A and B, respectively. As (2.6) shows, I_{TC} can be zero when we select an appropriate ratio between $c_1 A$ and $c_2 B$. However, I_{TC} in (2.6) can be sensitive over process variation when $c_1 A$ and $c_2 B$ do not have a correlation over the process variation, which is generally true for many circuits. Consequently, a multiple temperature trimming is required to obtain a small I_{TC} over process variation even when A_{TC} and B_{TC} are less sensitive for the process variation.

The second type of computation to cancel the temperature coefficients of physical quantities is multiplication/division. When we use a multiplication, I can be expressed as

$$I = cAB, \tag{2.7}$$

where c is a temperature-independent constant. (2.7) results in

$$I_{TC} = A_{TC} + B_{TC}. \tag{2.8}$$

On the contrary, a division ($I = cA/B$) leads to

$$I_{TC} = A_{TC} - B_{TC}. \tag{2.9}$$

(2.8) and (2.9) show that I_{TC} can be zero when the absolute values of A_{TC} and B_{TC} are the same. If A_{TC} and B_{TC} are less sensitive over process variation, we can obtain a stable I_{TC} with no help from a multiple temperature trimming. However, when A_{TC} is different from B_{TC}, there is no design flexibility to compensate the difference because c cannot make any change for I_{TC}.

In summary, the two types of computations show the trade-off between design flexibility and robustness. Although an addition/subtraction gives us a design flexibility to make I_{TC} small, I_{TC} is less robust over process variation. A multiplication/subtraction can guarantee a stable I_{TC} while it

6
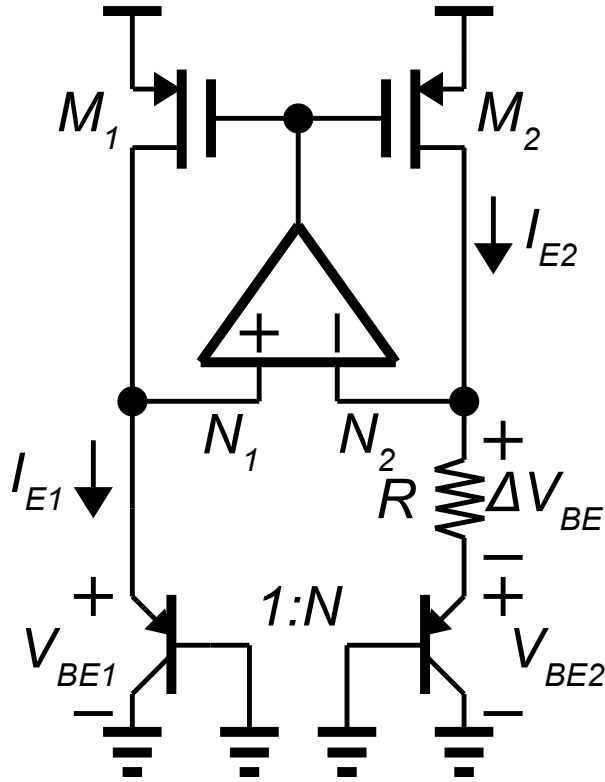
Figure 2.1: An example of ΔV_{BE}-based current references.

provides less design freedom.

2.2 Four Types of Current Reference Circuits

2.2.1 ΔV_{BE}-based Current References

One of the conventional approaches to generate a stable current is utilizing the difference between two V_{BE} as Fig. 2.1 shows [7]. The collector current of a bipolar junction transistor (I_{C}) is

$$I_C = I_S e^{V_{BE}/V_T}. \tag{2.10}$$

Thus,

$$V_{BE} = V_T \ln\left(I_C/I_S\right). \tag{2.11}$$

When we assume that the base current of each bipolar junction transistor (BJT) is negligible, V_{BE1} in Fig. 2.1 is

$$V_{BE1} = V_T \ln\left(I_{E1}/I_S\right). \tag{2.12}$$

Here, I_{E1} is the emitter current of the left BJT. Since the emitter area of the right BJT is N times larger than that of the left BJT, V_{BE2} is

$$V_{BE2} = V_T \ln\left(I_{E2}/(NI_S)\right). \tag{2.13}$$

If we assume that there is no mismatch between M_1 and M_2, I_{E1} equals to I_{E2}, considering that the op-amp (A_1) matches the voltage of node N_1 with the voltage of N_2. Consequently, the voltage difference between the two terminals of the resistor is

$$\Delta V_{BE} = V_T \ln N. \tag{2.14}$$

Therefore, the current flowing through the resistor is

$$I = \frac{V_T \ln N}{R}. \tag{2.15}$$

The temperature coefficient of I is

$$I_{TC} = V_{T,TC} - R_{TC}. \tag{2.16}$$

Even though a designer can choose a resistor that has a temperature coefficient close to $V_{\text{T,TC}}$, the circuit in Fig. 2.1 cannot provide a design flexibility to make I_{TC} zero. This is a natural consequence because (2.15) uses a division.

Modified versions of the circuit in Fig. 2.1 have been proposed to provide more design freedom [8–10]. Fig. 2.2(a) and Fig. 2.2(b) show the circuits proposed in [8] and [9], respectively. In
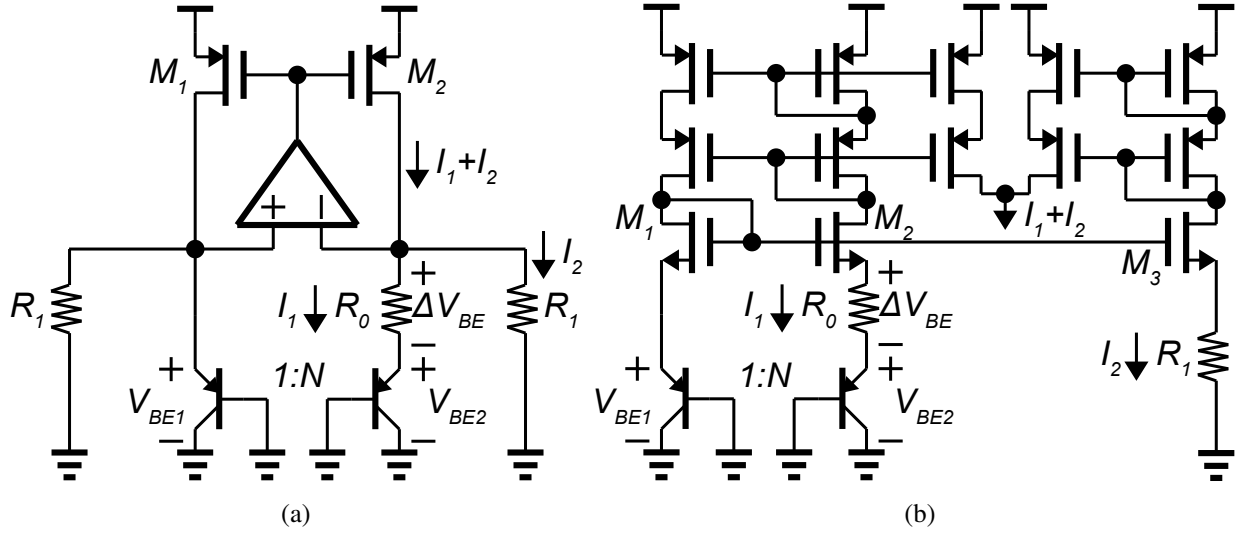
Figure 2.2: ΔV_{BE}-based current references that can provide a design flexibility. (a) utilizes an op-amp to generate ΔV_{BE}, whereas (b) uses a self-biased current mirror for the same purpose.

Fig. 2.2(a), the generated current is

$$I = I_1 + I_2 = \frac{V_T \ln N}{R_0} + \frac{V_{BE1}}{R_1}. \tag{2.17}$$

Thus, I_{TC} is

$$\begin{aligned} I_{TC} &= \frac{I_1}{I_1 + I_2} I_{1,TC} + \frac{I_2}{I_1 + I_2} I_{2,TC} \\ &= \frac{I_1}{I_1 + I_2} (V_{T,TC} - R_{0,TC}) + \frac{I_2}{I_1 + I_2} (V_{BE1,TC} - R_{1,TC}). \end{aligned} \tag{2.18}$$

I_{TC} can be zero only when $I_{1,\mathrm{TC}}$ and $I_{2,\mathrm{TC}}$ have opposite polarities. At a certain temperature, we can choose an appropriate ratio between I_1 and I_2 by using design parameters (N, R_0, and R_1) to obtain zero I_{TC}. However, the ratio cannot be maintained for a wide temperature range because of the opposite polarities of $I_{1,\mathrm{TC}}$ and $I_{2,\mathrm{TC}}$. As a result, I_{TC} can be zero only at a specific temperature. In addition, the ratio cannot be maintained over process variation as well due to the fact that ΔV_{BE} is not sensitive over process variation unlike V_{BE}. Therefore, process variation can significantly

change both of I and I_{TC}.

The circuit in Fig. 2.2(b) has the same design equation with the circuit in Fig. 2.2(a). Accordingly, the circuit has the same limitations we discussed in the previous paragraph. The difference between the two circuits comes from the two circuit implementations of the same design equation. The circuit in Fig. 2.2(b) utilizes a self-biased current mirror instead of an op-amp to generate ΔV_{BE}, resulting in the reduced power consumption of the circuit. In addition, M_2 and M_3 generate I_2 in (2.17) by assigning V_{BE} to R_1.

ΔV_{BE}-based current references are not widely used for generating a current smaller than 1 µA because of two reasons. First, as discussed in Section 2.1.1, V_{BE} should be higher than 170 mV at room temperature. This means that R_1 in Fig. 2.2(a) and Fig. 2.2(b) should be larger than 170 MΩ to generate a current smaller than 1 nA, leading to an excessively large area. Second, $V_{BE,TC}$ is large and sensitive to V_{BE} variation when V_{BE} is small as (2.2) shows. Due to the large $V_{BE,TC}$, I_2 in (2.18) should be much smaller than I_1, resulting in an even larger R_1.

Note that when I_C is small, the ratio between I_C and I_B (β_F) is not constant anymore and decreases significantly [6]. Accordingly, we cannot approximate I_E to I_C, and the temperature variation of β_F should be considered in (2.16) and (2.18).

2.2.2 Beta-multiplier-based Current References

Another type of conventional current references is based on a beta multiplier. Fig. 2.3 shows one of beta-multiplier-based current references. When the circuit starts its operation, the source voltage of M_2 is close to 0 V. Consequently, the drain current ratio between M_2 and M_1 is approximately K. Since the current mirror that consists of M_3 and M_4 matches I_1 with I_2, the drain currents of M_1 and M_2 keep increasing until the source voltage of M_2 reaches the voltage that sets the drain current ratio between M_1 and M_2 as 1. At the operating point, I_1 equals to I_2.

When all MOSFETs are in the saturation region, I_1 and I_2 can be expressed as below.

$$I = I_1 = I_2 = \frac{2}{\mu_n C_{ox} W_N / L_N} \frac{1}{R_S^2} \left(1 - \frac{1}{\sqrt{K}} \right). \tag{2.19}$$
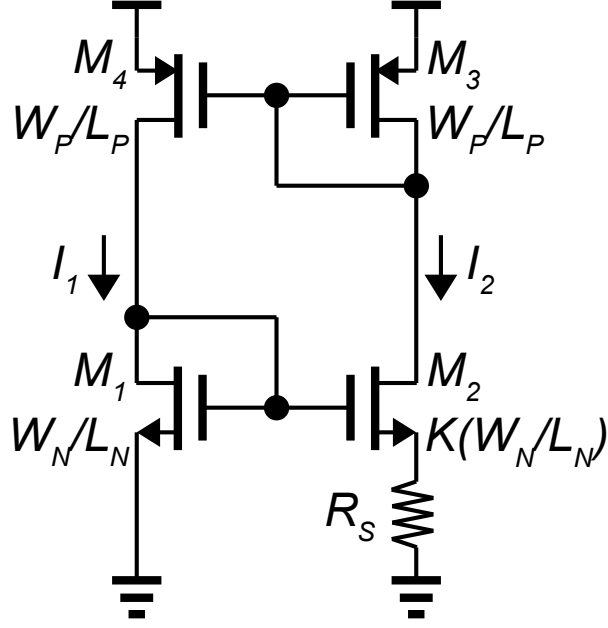
Figure 2.3: An example of beta-multiplier-based current references.

Thus, I_{TC} is

$$I_{TC} = -\mu_{n,TC} - 2R_{S,TC}. \tag{2.20}$$

(2.19) and (2.20) show two limitations of the circuit in Fig. 2.3. First, the circuit is not appropriate for generating a nA-range current. Let $K' = 1 - 1/\sqrt{K}$. Since $\mu_n C_{ox} W_N / L_N = 2I/V_{OV1}^2$, we can rewrite (2.19) as

$$R_S = \frac{V_{OV1}}{I} \sqrt{K'}. \tag{2.21}$$

Here, V_{OV1} means the overdrive voltage of M_1. When V_{OV1} equals to 100 mV, K equals to 2, and I equals to 1 nA, R_S will be 54 MΩ, which takes a huge area. Additionally, channel lengths of M_1~M_4 should be long enough to guarantee that all MOSFETS are in the saturation region while generating a nA-range current, resulting in a large area as well. The second limitation is that there is no design flexibility that can set I_{TC} as zero. This is because μ_n and R_S are multiplied in (2.19).

Various circuits have been proposed to overcome the first limitation by improving the circuit shown in Fig. 2.3 [11–13]. Instead of using R_S, [11] utilizes a PTAT floating voltage source inserted between the gates of M_1 and M_2 as Fig. 2.4(a) shows. In Fig. 2.4(a), if we assume that there is no
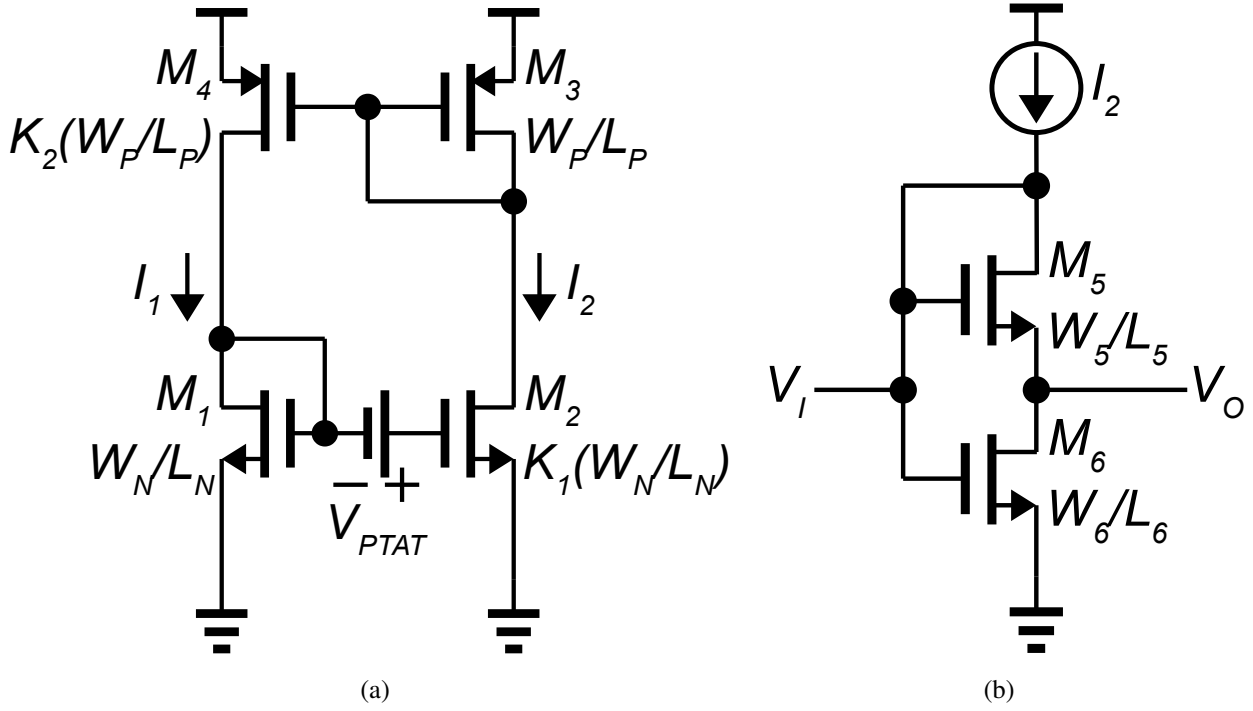
Figure 2.4: An example of beta-multiplier-based current references that do not use a resistor. (a) Schematic of the current reference. (b) Schematic of the PTAT voltage source.

threshold voltage mismatch between M_1 and M_2, I_2 is

$$I_2 = \mu_n C_{ox} K_1 \frac{W_N}{L_N} \frac{V_{PTAT}^2}{(1 - \sqrt{m})^2}, \tag{2.22}$$

where m equals to $K_1 K_2$, and V_{PTAT} is the voltage of the PTAT voltage source. Since I_2 is proportional to V^2_{PTAT}, a small V_{PTAT} results in a small I_2 while all MOSFETs in Fig. 2.4(a) are in the saturation region. Fig. 2.4(b) represents the circuit implementation of the PTAT floating voltage source. The following equation relates the input voltage (V_{I}) and the output voltage (V_{O}) of the floating voltage source by assuming that the MOSFETs in Fig. 2.4(b) are in the subthreshold region.

$$V_O = V_I + nV_T ln(\frac{W_6/L_6}{W_5/L_5}). \tag{2.23}$$

12

Consequently,

$$V_{PTAT} = nV_T ln(\frac{W_6/L_6}{W_5/L_5}).$$ (2.24)

From (2.22) and (2.24), $I_{2,\text{TC}}$ is

$$I_{2,TC} = \mu_{n,TC} + 2V_{T,TC}.$$ (2.25)

Even though the circuit in Fig. 2.4 can generate a nA-range current while consuming small area, it still does not have a design flexibility for $I_{2,\text{TC}}$.

Unlike [11], [12] employs a high-voltage MOSFET to remove R_S. On the other hand, [13] uses two different body biases for a current mirror in a beta multiplier for the same purpose with [12]. Since the circuits proposed in [12] and [13] generate a μA-range current, it is hard to justify that the circuits are better than the circuit in Fig. 2.3.

Circuits proposed in [14, 15] improve the design freedom of the circuit in Fig. 2.3 to obtain the better temperature coefficient of the generated current. [14] replaces R_S with a linear-region MOSFET and assigns a PTAT voltage to the drain of the linear-region MOSFET. Since a bandgap voltage reference generates the PTAT voltage, the temperature coefficient of the PTAT voltage can be designed to minimize the temperature coefficient of the generated current. [15] improves the design flexibility by adding one more diode-connected MOSFET to the source of M_1 in Fig. 2.3. Unfortunately, both of [14] and [15] do not provide measurement results.

If M_1 and M_2 in Fig. 2.3 are in the subthreshold region and if we ignore nonideal effects such as the finite resistance seen at the drain, the drain-induced barrier lowering, and the body effect of the MOSFETs, the generated current is given by

$$I = I_1 = I_2 = \frac{nV_T ln(K)}{R_S},$$ (2.26)

where n is the slope factor. Thus,

$$I_{TC} = V_{T,TC} - R_{S,TC}.$$ (2.27)

The numerator in (2.26) is 27 mV at room temperature when n and K equal to 1.5 and 2, respectively. As a result, to generate a 1 nA current, R_S should be 27 MΩ, which is 50% smaller than the resistance calculated from (2.21). However, when M_1 and M_2 are in the subthreshold region, the beta multiplier is more sensitive to mismatch variations. If we assume that there is a threshold voltage mismatch between M_1 and M_2 (ΔV_{TH}), (2.26) can be rewritten as

$$I = \frac{\Delta V_{TH} + nV_T ln(K)}{R_S}.$$ (2.28)

If $nV_T ln(K)$ is 27 mV, ΔV_{TH} and $nV_T ln(K)$ can be comparable, resulting in decreased accuracy for I and I_{TC}. On the other hand, when M_1 and M_2 are in the saturation region, ΔV_{TH} can be negligible if ΔV_{TH} is much smaller than V_{GS2}, which is generally true since M_2 is in the saturation region. Therefore, in many cases, $nV_T ln(K)$ in (2.28) should be as large as 100 mV as [16] shows, leading to an excessively large area for R_S.

[17–22] replace R_S with a MOSFET in the triode or the saturation region. The currents generated from the circuits in [17, 19–21] all have the following relation.

$$I \propto \mu V_T^2.$$ (2.29)

Thus,

$$I_{TC} = \mu_{TC} + 2V_{T,TC}.$$ (2.30)

Since (2.30) does not have any design freedom, [22] exploits a threshold voltage difference between two transistors that have different sizes to add more design freedom in (2.30). However, if the threshold voltage difference (17 mV) is comparable to the mismatch variations of the threshold voltages, the temperature coefficient of the generated current from the circuit proposed in [22] can be several hundreds ppm/°C as the measurement results in [22] show.

2.2.3 Smartly Biased Current References

When a MOSFET is in the saturation region, there is a V_{GS} bias voltage that sets the temperature coefficient of the MOSFET drain current as zero [23]. The bias voltage is called "zero temperature coefficient (ZTC) bias voltage" and can be derived from the square law shown below.

$$I = \frac{1}{2}\mu C_{OX}\frac{W}{L}\left(V_{GS} - V_{TH}\right)^2.$$

(2.31)

The temperature coefficient of I is

$$I_{TC} = \mu_{TC} - \frac{2V_{TH}}{V_{GS} - V_{TH}}V_{TH,TC}.$$

(2.32)

Since μ_{TC} and $V_{TH,TC}$ are negative values, I_{TC} can be zero when the following equation holds.

$$V_{GS} = \left(\frac{2V_{TH,TC}}{\mu_{TC}} + 1\right)V_{TH}.$$

(2.33)

When V_{TH} is given by

$$V_{TH} = V_{TH0} - \gamma T,$$

(2.34)

the temperature coefficient of V_{TH} is

$$V_{TH,TC} = \frac{-\gamma}{V_{TH}}.$$

(2.35)

From (2.4), (2.34), and (2.35), we have

$$V_{GS} = V_{TH0} + \frac{1}{3}\gamma T.$$

(2.36)

As (2.36) shows, ideally V_{GS} should be a PTAT voltage to set I_{TC} as zero at all temperature. Smartly biased current references generate the ZTC bias voltage and assign the bias voltage to an NMOS or a PMOS current generator. [24] utilizes a bandgap voltage reference to generate a ZTC voltage.

15

On the other hand, [25] uses a nanowatt PTAT voltage generator for the same purpose.

One drawback of this approach is that ZTC bias voltages are sensitive to process variation. In other words, the generated V_{GS} should have the same process variation with V_{TH} as (2.33) shows. [25] proposes an all-PMOS PTAT voltage generator to generate a ZTC bias voltage that tracks the V_{TH} variation of a PMOS current generator. [26] utilizes a threshold voltage monitoring circuit that outputs V_{TH0} of a MOSFET as a ZTC bias voltage generator. If the threshold voltage of a current generator is the same with the threshold voltage of a MOSFET monitored, the current generator can always have an appropriate bias voltage regardless of process variation.

The other drawback of this approach is that a MOSFET that is biased to generate a stable current should be in the saturation region. As a result, generating a current smaller than 1 nA is challenging when we consider the following equation.

$$V_{OV} = V_{GS} - V_{TH} = \sqrt{\frac{2I}{\mu C_{OX} W/L}}. \tag{2.37}$$

As (2.37) shows, W/L for generating 1 nA should be 1000 times smaller than W/L for 1 μA when both cases have the same V_{OV}. To avoid this issue, the circuit proposed in [27] uses a subthreshold MOSFET as a current generator and produces a gate bias voltage for the current generator. Even though the circuit generates a pA-range current with a small area, it has relatively low accuracy because of the exponential relationship between V_{GS} and the generated current.

2.2.4 Division-based Current References

Division-based current references generate a stable current by assigning a voltage to a resistor. Fig. 2.5 shows one example of division-based current references [28, 29]. [28] utilizes a PTAT voltage generator that consists of four subthreshold MOSFETs to generate V_{C}. On the other hand, [29] uses floating gates to generate a temperature-independent V_{C} and a temperature-insensitive resistance for R in Fig. 2.5. I can be written as

$$I = \frac{V_C}{R}. \tag{2.38}$$
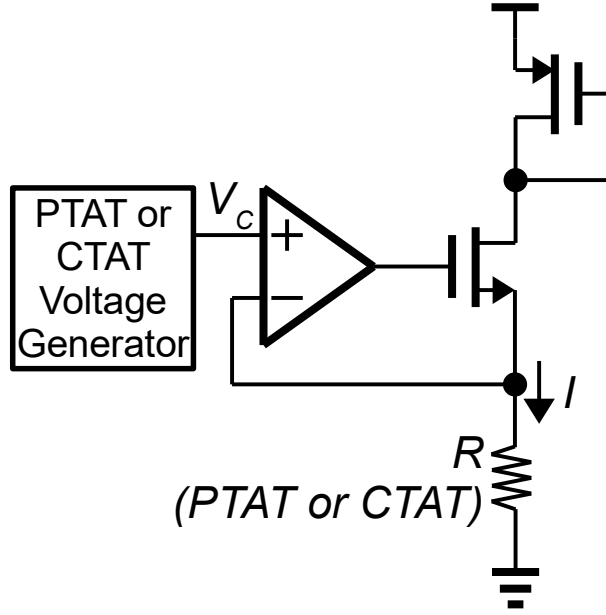
Figure 2.5: An example of division-based current references.

Thus,

$$I_{TC} = V_{C,TC} - R_{TC}. \tag{2.39}$$

Even though (2.38) includes only a division, division-based current references have enough design freedom because a designer can choose $V_{C,TC}$ while designing the compensation voltage generator in Fig. 2.5. One drawback of the circuits in [28, 29] is that R should be very large when the circuits generate a small current. For example, if V_C is 100 mV, R should be 100 MΩ to generate 1 nA, resulting in an excessively large area.

The circuit proposed in [30] reduces R to minimize the area of the circuit by generating a small temperature-independent voltage. Since $V_{C,TC}$ is close to zero, I_{TC} cannot be zero anymore. [31, 32] utilize MOSFET gate leakages to obtain large resistance with small area.

2.3 Performance Comparison Among Current Reference Types

Fig. 2.6 shows the measured performance of various current references and their types. In Fig. 2.6, "ΔVBE" means ΔV_{BE}-based current references. "BM-Sat" and "BM-Sub" indicate beta-multiplier-based current references in the saturation region and in the subthreshold region, respec-
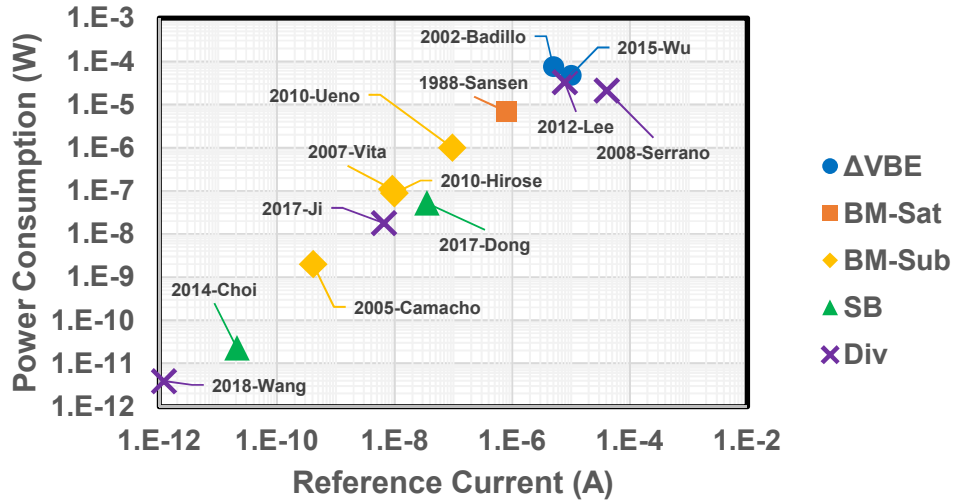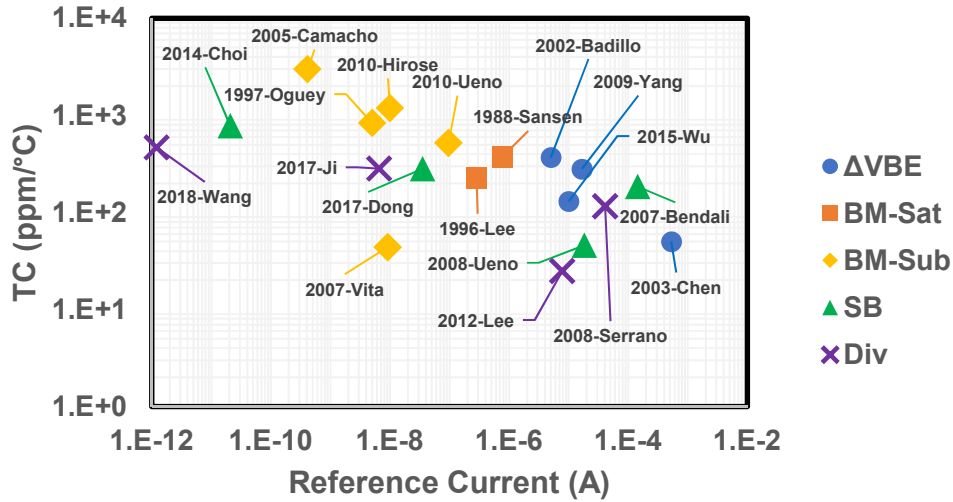
Figure 2.6: Performance of five current reference types. (a) Generated reference current versus total power consumption. (b) Generated reference current versus temperature coefficient.

tively. "SB" and "Div" stand for smartly biased current references and division-based current references, respectively. Note that the number of current references shown in Fig. 2.6(a) is smaller than the number of current references shown in Fig. 2.6(b). The reason for the difference is that some articles do not report the power consumption of the current reference each article proposes.

The ΔV_{BE}-based current references and the beta-multiplier-based current references in the saturation region generate currents comparable to or higher than 1 µA. On the other hand, the beta-multiplier-based current references in the subthreshold region, the smartly biased current ref-

18

erences, and the division-based current references can generate nA-range or pA-range currents. Especially, the smartly biased current references and the division-based current references have better temperature coefficient than others in many cases as Fig. 2.6(b) shows when we compare the references with other types of current references that provide comparable amounts of current. In addition, if a current reference provides a small amount of current, the reference tends to consume less power as Fig. 2.6(a) indicates. Therefore, circuit techniques that utilize smart biasing or division are better choices than others to generate a small and accurate current over temperature variations with low power consumption.

2.4 Conclusion

This chapter categorizes current reference circuits into four types and discusses the characteristics of each type. This chapter also summarizes recent research on each type of current references and shows a research trend. Smart biasing circuit techniques and division-based circuit techniques are more promising than other circuit techniques since the two techniques can generate a smaller and more accurate current over temperature variations with lower power consumption than others.

3.   AN ULTRALOW-POWER HIGH-ACCURACY CURRENT REFERENCE USING

AUTOMATIC CALIBRATION*

3.1 Motivation

A current reference circuit provides a stable bias current for many analog and mixed-signal circuits despite process-voltage-temperature (PVT) variations. As a result, on-chip fully-integrated current references have been researched to reduce a bill of materials and a form factor. Recently, researchers pay attention to on-chip fully-integrated current references that can generate a small current in a range of pA or nA because the Internet of Things (IoT) applications commonly require a small bias current for ultralow-power circuits.

One challenge that makes designing the current references hard is a trade-off between the power consumption and the accuracy of a current reference. Fig. 3.1 clearly demonstrates the trade-off, using selected low-power current references that generate currents smaller than 1 μA [20, 22, 25, 30–33]. Each axis of Fig. 3.1 represents one of three key characteristics of a current reference: the amount of a generated current from a current reference, the temperature coefficient of the generated current, and the total power consumption of the current reference. In Fig. 3.1(a), the power consumption for generating a reference current tends to decrease as the reference current decreases. However, a lower-power current reference results in a larger temperature coefficient as Fig. 3.1(b) shows. In short, low power consumption compromises the accuracy of a current reference. One way to relax the trade-off is utilizing a multiple-temperature trimming as [32] and [30] show. However, a multiple-temperature trimming requires an additional post-fabrication test setup, leading to an increased cost and a decreased production throughput.

[3] proposes a current reference that relaxes the tight trade-off with no help from a multiple-temperature trimming. In the current reference, an automatic calibration circuit periodically corrects the current generated from an ultralow-power current generator. After the calibration is fin-
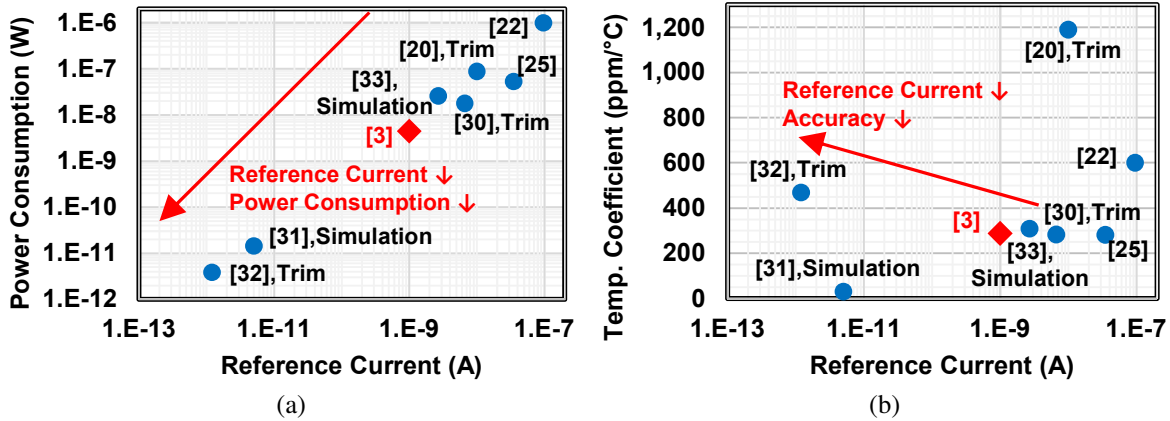
Figure 3.1: Performance of various current references. (a) Generated reference current versus total power consumption. (b) Generated reference current versus its accuracy over temperature variation. ©2020 IEEE.

ished, all circuit blocks for the calibration are powered off. If the temperature and the supply voltage of the ultralow-power current generator change slowly, the calibration process does not need to be activated frequently. Therefore, the proposed circuit can provide an accurate current while consuming low average power.

This chapter discusses in detail about the current reference proposed in [3]. Section 3.2 introduces the system-level architecture of the proposed current reference. Section 3.3 elaborates the building blocks of the current reference. The accuracy of the proposed current reference is analyzed in Section 3.4. A calibration algorithm and a calibration time frame are discussed in Section 3.5. Current-providing mechanisms for load circuits are given in Section 3.6. Section 3.7 presents the measurement results of prototype chips. Finally, Section 3.8 makes a conclusion.

3.2 System-Level Architecture

The proposed current reference has three modes of operation: calibration mode, normal mode, and sleep mode. In the calibration mode, the proposed circuit calibrates a small and inaccurate current (I_{REF2}) using a relatively large and accurate current (I_{REF1}). In Fig. 3.2(a), the leakage-based current digital-to-analog converter (IDAC) generates I_{REF2}, while the high-power low-temperature-coefficient on-chip current reference provides I_{REF1}. The calibration mode is activated when fol-
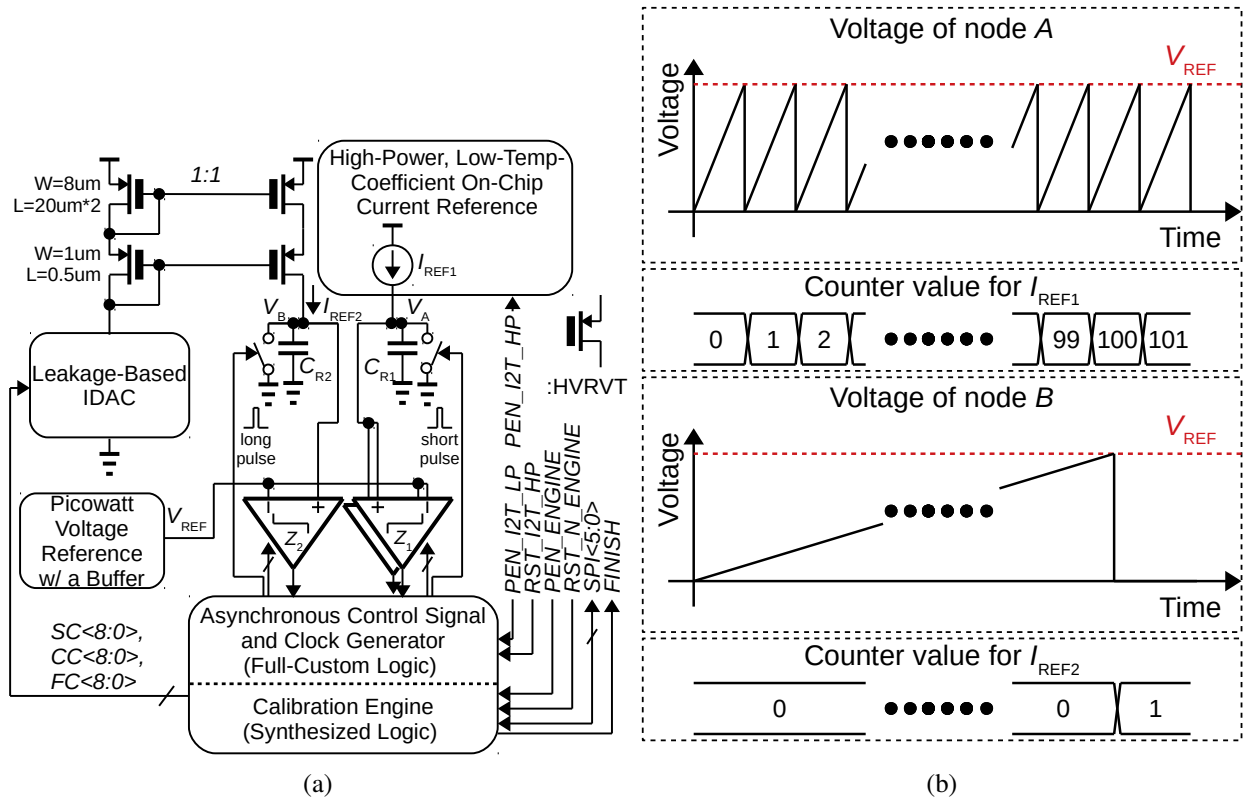
Figure 3.2: System-level architecture. (a) Block diagram. (b) Example waveforms of selected signals when I_{REF1}:I_{REF2}=100.5:1. ©2020 IEEE.

lowing conditions are met: the active-high power-enable signals (*PEN_I2T_HP*, *PEN_I2T_LP*, and *PEN_ENGINE*) are high, *RST_I2T_HP* is low, and *RST_N_ENGINE* is high. In these signal names, *I2T_HP* means a current-to-time converter for the high-power current reference providing I_{REF1}, whereas *I2T_LP* means a current-to-time converter for the low-power current generator (IDAC) supplying I_{REF2}. Additionally, *ENGINE* means the calibration engine in Fig. 3.2(a). In the calibration mode, the proposed circuit first converts the two currents to time signals by charging two capacitors: C_{R1} and C_{R2}, where C_{R1}=C_{R2}=1.79 pF. Since I_{REF1} is larger than I_{REF2}, the node voltage at A (V_A) rises more quickly than the node voltage at B (V_B) as Fig. 3.2(b) shows. When V_A crosses V_{REF}, a rising edge appears at the output of an auto-zeroed comparator Z_1. The full-custom logic detects the rising edge and subsequently produces various digital signals asynchronously: a pulse signal to discharge C_{R1}, a clock signal for the calibration engine, and other necessary signals

to control Z_1. The calibration engine counts the rising and the falling edges of the clock signal derived from I_{REF1}. Once C_{R1} is reset, I_{REF1} charges the capacitor again, and the proposed circuit repeats the charging and discharging process until V_B crosses V_{REF}. A counter in the full-custom logic counts the rising edge that appears at the output of the left-side comparator Z_2, and the full-custom logic resets C_{R2}. Since the engine operates based on the clock generated from I_{REF1}, the engine detects the change in the counter value for I_{REF2} when the next rising edge of Z_1 comes. Therefore, in the case of Fig. 3.2(b), the engine calculates the measured current ratio between I_{REF1} and I_{REF2} as 101:1 by comparing the two counter values. The ratio can be considered the outcome of a rounding-up because the actual ratio between the two currents is 100.5:1. When the engine obtains the measured current ratio, it assigns to the IDAC a better input code that makes I_{REF2} closer to I_{REF1}/N based on its algorithm by changing the inputs of the IDAC (*SC<8:0>*, *CC<8:0>*, and *FC<8:0>*). Since the IDAC consists of three separate arrays of binary-weighted leakage sources, the three inputs control the leakage sources of each array in the IDAC. Afterwards, the engine releases the reset signal for C_{R2}. After trying multiple input code values for the IDAC, the engine assigns 1'b1 to *FINISH* to indicate that the calibration process is complete. Additionally, it fixes a final code value for IDAC. Through *SPI<5:0>*, the code value can be read or be written for testing purposes. Furthermore, *SPI<5:0>* can set the target integer ratio between I_{REF1} and I_{REF2} (N).

After the calibration ends, the proposed circuit enters the normal mode. In the normal mode, all circuits are powered off, except the IDAC, the current mirror, and the picowatt voltage reference in Fig. 3.2(a), by assigning 1'b0 to *PEN_I2T_HP*, *PEN_I2T_LP*, and *PEN_ENGINE*. Since retention flops [34] can maintain the input digital code for the IDAC, the IDAC can provide I_{REF2} through the current mirror for other analog circuits as a reference/bias current, which equals to I_{REF1}/N. Also, the normal mode has significantly lower power consumption than the calibration mode thanks to small I_{REF2} around 1 nA and a power gating technique [34] that minimizes leakages of digital logics. If the other circuits do not need the reference/bias current anymore, the proposed circuit moves to the sleep mode when all the *PEN* signals and *RST_N_DIGENGINE* are 1'b0, which sets the input of IDAC as 27'b0. In the sleep mode, the entire circuit consumes the lowest energy.
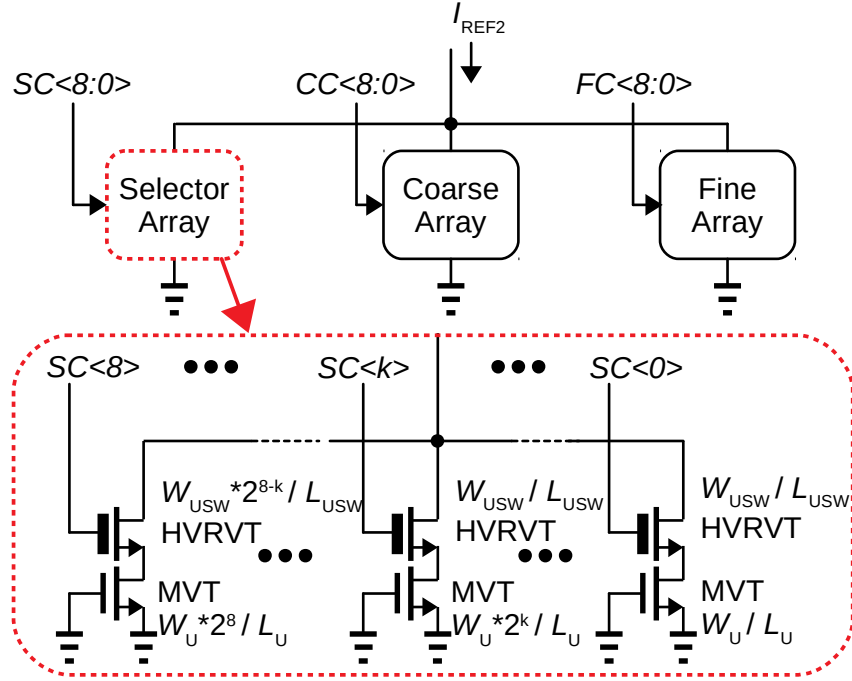
Figure 3.3: Schematic of the leakage-based IDAC. ©2020 IEEE.

The calibration mode is turned on periodically to compensate IDAC's output current deviation caused by temperature and supply voltage variations. However, the proposed circuit remains energy efficient, since it spends most of its time in the normal mode or in the sleep mode with low power consumption, whereas the calibration mode lasts for only a short time. In summary, the proposed circuit creates a small and accurate current by relaxing the trade-off between the accuracy and the power consumption of a current reference.

3.3   Building Blocks

3.3.1   Leakage-Based IDAC

The leakage-based IDAC in Fig. 3.3 consists of three arrays: a selector array, a coarse array, and a fine array. Each array has a 9-bit digital input and produces a current as an output. The selector array generates a wide-range current, whereas the coarse and the fine arrays provide sufficient granularity in the range. Accordingly, the selector array has the largest one-least-significant-bit current (I_{LSB}). The I_{LSB} of the coarse array is the second largest, and the I_{LSB} of the fine array is the

smallest. In the wide-range selector array, each medium-threshold (MVT) NMOS transistor that has zero voltage difference between its gate and source supplies a large leakage current. A high-votlage regular-threshold (HVRVT) NMOS transistor is selected as a switch above each MVT NMOS transistor because its high threshold voltage allows only a small leakage when it is off. Since the k-th MVT NMOS transistor from the right has a 2^k times larger width than the width of the rightmost transistor, the current controlled by $SC<k>$ is 2^k times larger than the current handled by $SC<0>$. Consequently, the input to each array should be a binary-coded decimal to get ideally-constant I_{LSB} from the minimum input to the maximum. Unlike the leakage current suppliers, the switches have a constant width up to the k-th switch from the right, whereas the p-th switch from the right has a width equals to $W_{USW} \times 2^{p-k}$, where p>k. Thanks to the constant width, the area of the selector array can be reduced. The other arrays have the same structure as the selector array. Meanwhile, to generate leakage currents that have various scales, they use different types of NMOS transistors as leakage sources: regular-threshold (RVT) NMOS transistors for the coarse array and HVRVT NMOS transistors for the fine array.

The sizes of the leakage sources in the three arrays are determined based on four requirements. First, the maximum current of the selector array should be larger than the current that the IDAC provides (I_{REF1}/N). Second, the I_{LSB} of the selector array should be smaller than the maximum current of the coarse array. Third, the I_{LSB} of the coarse array should be smaller than the maximum current of the fine array. Lastly, the I_{LSB} of the fine array should be smaller than 0.1 % of I_{REF1}/N to minimize a quantization error. The second and the third requirements are intended for calibration accuracy through current redundancies between the arrays. When 1-LSB increase in the input code affects more than two arrays simultaneously, the IDAC has negative differential nonlinearity (DNL) smaller than -1 due to the redundancies. For example, the current generated from a setting, $SC<8:0>$=9'h000 and $CC<8:0>$=$FC<8:0>$=9'h1FF, is larger than the current produced from a setting, $SC<8:0>$=9'h001 and $CC<8:0>$=$FC<8:0>$=9'h000. The redundancy prevents calibration accuracy degradation by canceling positive DNL that might occur from current mismatches between the arrays. Mismatches between the leakage sources in an array can be alleviated by

25

utilizing large area for the sources.

Two factors determine the k value and the sizes of the HVRVT switches. If k=0 for an array and there is no mismatch between transistors in the array, V_{DS} of on-state switches are uniform because the resistance of an on-state switch is scaled in inverse proportion to the amount of the current flowing through the switch. As a result, the array has small DNL due to the uniform drain voltage of its leakage sources. In other words, k can be increased to reduce area as long as the value does not harm the second, the third, and the forth design criteria for leakage sources. Additionally, the sizes of the switches are decided to have a total leakage current of an array smaller than the 1-LSB current of the array when all switches in the array are off. The transistor sizes are shown as follows: W_U=4 um, L_U=5 um, W_{USW}=3 um, L_{USW}=0.35 um, and k=5 for the selector array; W_U=0.6 um, L_U=19.9 um, W_{USW}=0.22 um, L_{USW}=5 um, and k=6 for the coarse array; W_U=4 um, L_U=4 um, W_{USW}=0.22 um, L_{USW}=20 um, and k=6 for the fine array.

3.3.2   Current-to-Time Converters

The same two auto-zeroed [35] unbalanced-current-starved-inverter-based comparators convert I_{REF1} in Fig. 3.2(a) to a time signal. Fig. 3.4(a) shows the two comparators. Each comparator has two operation phases: a sampling phase and a tracking phase. In the sampling phase, the comparator samples the switching threshold voltages [36] of the two unbalanced current-starved inverters (V_{ST1}, V_{ST2}). C_1 stores the voltage difference between the reference (V_{REF}) and the switching threshold of the first inverter because V_{SP1}=V_{ST1}. In addition, C_2 keeps V_{SP1}-V_{SP2}, where V_{SP2}=V_{ST2}. In the tracking phase, the comparator acts as an offset-canceled comparator. When V_{IN} reaches V_{REF}, the input (and the output) voltage of the first inverter ideally equals to V_{ST1} regardless of PVT variations because of the voltage sampled in C_1. Likewise, the input (and the output) voltage of the second inverter is ideally V_{ST2} when V_{IN} reaches V_{REF}. Therefore, the switching threshold of the comparator is V_{REF} regardless of V_{ST1} and V_{ST2}. Each inverter in the comparator consumes 400 nA and has a 32-dB DC gain (27 °C, TT) at its switching threshold. Two current mirrors generate the bias voltages for each comparator. The high-power accurate current reference in Fig. 3.2(a) provides a bias current for the current mirrors. The hysteresis buffer [36] in the

26
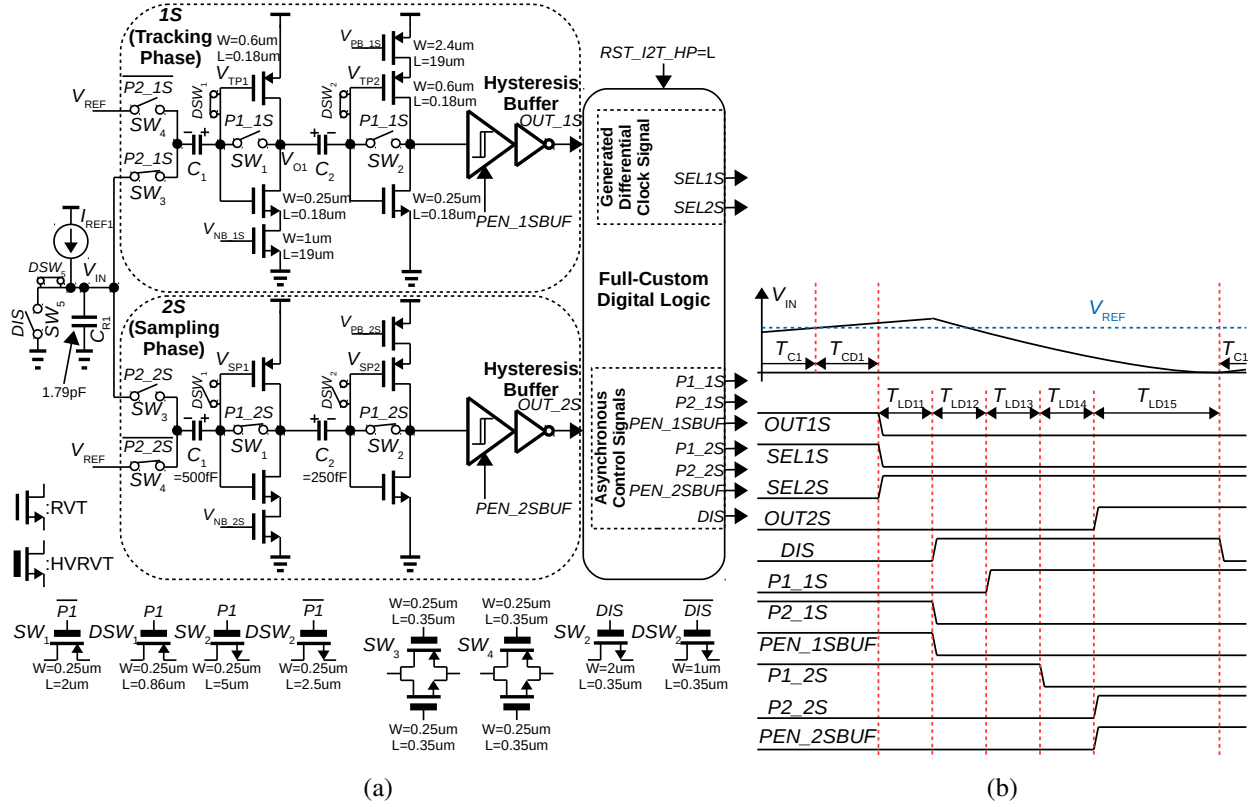
Figure 3.4: Current-to-time converter for I_{REF1}. (a) Schematic. (b) Timing diagram. ©2020 IEEE.

comparator suppresses possible glitches caused by noise.

Each current-starved inverter in Fig. 3.4(a) has only one NMOS (or PMOS) current source. Accordingly, V_{ST1} is higher than V_{REF}, whereas V_{ST2} is lower than V_{ST1}. These high and low switching thresholds ensure that V_{TP1} is always higher than the ground and V_{TP2} is always lower than the supply voltage in the tracking phase, although V_{IN} and V_{O1} start from the ground and near the supply voltage respectively. For example, a balanced current-starved inverter that has a PMOS current source (W=2.4 μm, L=19 μm) and an NMOS current source (W=1 μm, L=19 μm) can have a switching threshold ranging from 515 mV to 863 mV at 80 °C in the FF corner case when the supply voltage is 1.5 V and when there are bias current mismatches smaller than ±10 % between the two current sources. With the same temperature, corner case, supply voltage, and bias current mismatch conditions, the unbalanced current-starved inverter has a switching threshold ranging from 1.12 V to 1.13 V. Since V_{REF} is 610 mV in the FF corner case, the balanced current-starved

27

inverter samples -95 mV on C_1 in the sampling phase when the switching threshold of the inverter is the minimum. On the other hands, the unbalanced current-starved inverter stores 510 mV on C_1. Accordingly, the maximum voltage differences between the two nodes of SW_1 in the tracking phase are 1.60 V and 0.99 V for the balanced and unbalanced current-starved inverters respectively. Although the unbalanced current-starved inverter uses a PMOS-only switch for SW_1, the balanced inverter cannot use the same switch because of its high on-resistance when the switching threshold of the inverter is the lowest. Instead, a transmission gate (W=0.25 μm, L=0.5 μm for both of transistors) is utilized considering that the switching threshold of the inverter has high variability. The length of the transmission gate is selected to closely match the on-resistance of the transmission gate (1.81 MΩ) with that of the PMOS-only switch (1.74 MΩ) when both of the inverters are in the sampling phase at 27 °C in the TT corner case. Both switches have the minimum width to minimize the capacitance seen at the drain or source of each transistor. Unlike the unbalanced current-starved inverter, the balanced inverter does not have dummy switches (DSW_1 and DSW_2) because a PMOS in the transmission gate can cancel the charge injection of an NMOS in the transmission gate without dummy switches. Under this circumstance, the unbalanced current-starved inverter has 65.6 fA leakage through SW_1 when the tracking phase starts, whereas the balanced inverter has 26.8 pA leakage.

Due to the small leakage, the unbalanced-starved-inverter-based comparator can have a small offset after the auto zeroing. The offset can be quantified by measuring the difference between two voltages: sampled voltage on C_1 (and C_2) in the sampling phase; actual voltage on C_1 (and C_2) in the tracking phase when the input of the frist (and the second) inverter in the comparator reaches the switching threshold of the inverter. The comparator based on the unbalanced inverters in the current-to-time converter for I_{REF1} has 67 μV and -10 μV offsets on C_1 and C_2 respectively (30 °C, FF). However, the comparator based on the balanced inverters has a -462 μV offset on C_1 and a 496 μV offset on C_2. The total input-referred offsets of the comparators are 67 μV and -450 μV respectively considering that the offset on C_2 is divided by the gain of the first inverter. Temperature variations of the offsets are 111 μV and 94 μV respectively from -20 °C to 80 °C.

If the current-to-time converter for I_{REF2} utilizes the same comparators, the effect of the leakage on the comparator offset can be more prominent because the comparators stay in the tracking phase longer than the comparators in the current-to-time converter for I_{REF1}. The balanced-starved-inverter-based comparator in the converter for I_{REF2} has a -2.9 mV offset (80 °C, FF). Since the temperature variation of the offset is 2.9 mV and V_{REF} is 610 mV, the temperature-dependent offset can increase the temperature coefficient of I_{REF2} by at most 47.5 ppm/°C. However, the unbalanced-starved-inverter-based comparator has a 290 µV offset (80 °C, FF). The temperature variation of the offset is 102 µV, resulting in at most 1.7 ppm/°C temperature coefficient degradation of I_{REF2}.

Another benefit of employing the unbalanced current-starved inverter is to provide a large bandwidth with a fixed current budget. When the unbalanced inverter is loaded with a replica inverter, it has a 2.7-MHz 3-dB bandwidth at its switching threshold (27 °C, TT, 1.8 V), whereas the balanced inverter has a 17-KHz bandwidth with the same current budget (400 nA). This is because the balanced inverter has a large output resistance due to the cascoded transistors. When the supply voltage is 1.8 V and there is no bias current mismatch, the worst delay of the balanced-starved-inverter-based comparator is 4.24 µs (80 °C, FF), whereas that of the unbalanced-starved-inverter-based comparator is 225 ns (-20 °C, SS). Due to the long delay of the balanced-starved-inverter-based comparator, moving to the sampling phase from the tracking phase is delayed after V_{IN} exceeds V_{REF}. As a result, the input and the output voltages of each inverter deviate largely from the switching threshold of the inverter, resulting in huge settling errors on C_1 and C_2 in the sampling phase. When the current-to-time converter for I_{REF1} utilizes the two types of inverters, the worst settling error of the unbalanced inverter is 25.2 µV on C_1 (80 °C, SF). The temperature variation of the settling error is 21.5 µV, resulting in at most 0.35 ppm/°C temperature coefficient degradation of I_{REF1}. However, the balanced inverter shows a 40.5 mV settling error on C_1 in the worst case (-20 °C, FF), and its temperature variation is 11.3 mV, leading to at most 185 ppm/°C temperature coefficient degradation of I_{REF1}.

The full-custom digital logic in Fig. 3.4(a) generates a differential clock signal from the outputs of the comparators and provides asynchronous control signals for the comparators. Fig. 3.4(b)

represents the timing diagram of the asynchronous control signals. When SW_5 is disconnected, I_{REF1} charges C_{R1}, and V_{IN} rises from the ground. After a charging time (T_{C1}), V_{IN} crosses V_{REF}, and the upper-side comparator (*1S*) flips its output after the delay of the comparator (T_{CD1}). A 1-bit counter in the full-custom logic detects the falling edge of *OUT_1S* and changes the polarities of its outputs, *SEL1S* and *SEL2S*. After a short logic delay T_{LD11}, SW_5 starts discharging C_{R1}, and SW_3 in *1S* breaks the connection between C_{R1} and C_1. Additionally, the full-custom logic sets *PEN_1SBUF* as 1'b0 to power off the hysteresis buffer in *1S*. *1S* is in the sampling phase after an additional delay T_{LD12} because SW_1 and SW_2 are enabled. A non-overlapping signal generator produces two signals, *P1_2S* and *P2_2S*, with delays T_{LD13} and T_{LD14}. As a result, *2S* is in the tracking phase after $T_{LD13}+T_{LD14}$. At the same time, the full-custom logic enables the hysteresis buffer in *2S*. In short, the full-custom logic completes the process that interchanges the phases of operation between *1S* and *2S* in $T_{LD11}+T_{LD12}+T_{LD13}+T_{LD14}$. However, the full-custom logic needs a long delay (T_{LD15}) before charging C_{R1} again because discharging C_{R1} takes longer than $T_{LD12}+T_{LD13}+T_{LD14}$. Unlike the other delays, which static logic gates generate, a dynamic logic provides T_{LD15} by discharging its MOS capacitor with a constant rate using a current from the high-power accurate current reference in Fig. 3.2(a). Once the full-custom logic disconnects SW_5, the same process is repeated. The only difference is that *1S* and *2S* start from the sampling phase and the tracking phase respectively. Accordingly, the 1-bit counter in the full-custom logic detects the falling edge of *OUT_2S* when V_{IN} goes across V_{REF}. Afterwards, the full-custom logic first changes the control signals for *2S* to move *2S* to the sampling phase. *1S* enters the tracking phase later.

The current-to-time converter for I_{REF2} shown in Fig. 3.5 utilizes the same type of the comparator used for I_{REF1}. All devices in the comparator for I_{REF2} have the same sizes with the devices in the comparator for I_{REF1} except the sizes of the dummy switches (DSW_1 and DSW_2) in Fig. 3.4(a). There are two major differences between the two converters. First, the comparator in the converter for I_{REF2} is controlled by *PRESET* and *RST* from the calibration engine in Fig. 3.2(a). Second, the converter for I_{REF2} uses only one comparator because it does not need to generate a clock signal.
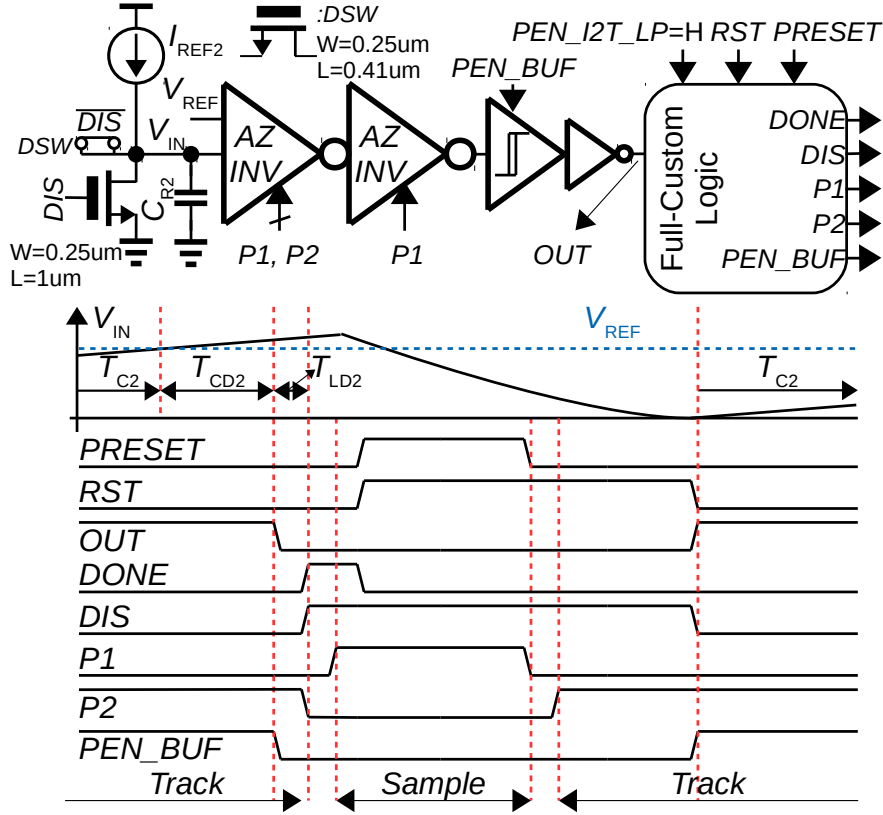
Figure 3.5: Current-to-time converter for I_{REF2}. ©2020 IEEE.

The operation of the converter is synchronized with the clock generated from the converter for I_{REF1}. When V_{IN} crosses V_{REF}, the comparator drives *OUT* to 1'b0 after T_{CD2}. A 1-bit counter activated by the falling edge of *OUT* assigns 1'b1 to its output, *DONE* after T_{LD2}. In addition, the full-custom logic in the converter generates control signals to discharge C_{R2} and to move the comparator to the sampling phase. In the next clock cycle, the calibration engine receives the *DONE* signal and changes *PRESET* and *RST* to 1'b1, which resets *DONE*. After the calibration engine computes the ratio between the two currents (I_{REF1} and I_{REF2}) and assigns a better input code to the IDAC, the engine changes *PRESET* to 1'b0, which moves the comparator to the tracking phase. In the next clock cycle, the engine sets *RST* as 1'b0, and I_{REF2} starts charging C_{R2} again.

3.3.3 Other Building Blocks

The high-power low-temperature-coefficient current reference in Fig. 3.2(a) has the same system-level architecture proposed in [28]. However, the reference has four major differences, compared to the circuit introduced in [28]. First, the reference utilizes a P+ poly resistor instead of a N+ poly resistor. Since the P+ poly resistor in the 180 nm technology has a lower temperature coefficient and a larger sheet resistance than the N+ poly resistor, the reference can generate a stable 100 nA current, while consuming small area. Second, the reference uses a picowatt complementary-to-absolute-temperature (CTAT) voltage generator to provide the compensation voltage for the resistor. The CTAT voltage generator has the same circuit structure proposed in [37]. The dimension of each MOS is tuned to generate a CTAT voltage [38]. Since the CTAT voltage generator has a line sensitivity better than 0.47 %/V even in the worst corner case (SF, 80°C), the reference does not need the bandgap regulator utilized in [28]. Third, the reference uses a self-biased one-stage op-amp to assign the CTAT voltage to the resistor. A leakage source provides a bias current for the op-amp. The op-amp has a 60-dB DC gain and consumes 1.2 nA at 27 °C in a power-on mode. Lastly, the reference has a power-off mode, where the entire circuit consumes 92.3 pA at 27 °C in the worst corner case (FF). The picowatt voltage reference proposed in [37] provides V_{REF} in Fig. 3.2(a), which is 557.6 mV in the TT corner case. The voltage buffer for the voltage reference is implemented by utilizing a one-stage op-amp. The high-power accurate current reference in Fig. 3.2(a) supplies a bias current for the op-amp.

3.4 Accuracy Analysis

There are six nonidealities in the proposed current reference as below.

1. Leakages that create a current difference between I_{REF1} (or I_{REF2}) and the actual current that charges C_{R1} (or C_{R2}).

2. Comparator offsets in the current-to-time converters.

3. Nonzero V_{IN} in Fig. 3.4 and Fig. 3.5 when I_{REF1} and I_{REF2} start to charge C_{R1} and C_{R2}

respectively.

4. Delays of the comparators and the digital logics in the current-to-time converters.

5. Nonzero temperature coefficients of V_{REF}, C_{R1}, and C_{R2}.

6. A finite resolution of IDAC (quantization error).

For a convenient analysis on how the nonidealities impact on the accuracy of the proposed current reference, the first three nonidealities are referred to V_{REF}. Consequently, V_{REF} in Fig. 3.4 and Fig. 3.5 can be rewritten as $V_{REF}+\alpha$ and $V_{REF}+\beta$ respectively. α is a V_{REF}-referred error that comes from the first three nonidealities in the current-to-time converter for I_{REF1}. β is also a V_{REF}-referred error from the same nonidealities in the current-to-time converter for I_{REF2}. If we substitute $V_{REF}+\beta$ with V_{REF2}, $V_{REF}+\alpha=V_{REF2}+\gamma$, where $\gamma=\alpha-\beta$. The analysis can start from the following equation, which can be established after the calibration engine finishes its algorithm.

$$T_{C2} + T_{CD2} + T_{LD2} + Q = N \times (T_{C1} + T_{CD1} + T_{LD1}). \tag{3.1}$$

T_{C1} and T_{C2} are time periods for I_{REF1} and I_{REF2} to charge C_{R1} and C_{R2} from ground to $V_{REF2}+\gamma$ and V_{REF2} respectively. T_{CD1} and T_{LD1} are the delay of the comparators and the summation of the logic delays ($\sum T_{LD1i}$) respectively in the current-to-time converter for I_{REF1}. T_{CD2} and T_{LD2} are the delay of the comparator and the clock-to-q delay of the 1-bit counter respectively in the current-to-time converter for I_{REF2}. N is a target ratio between I_{REF1} and I_{REF2}. Q is a time error that comes from a finite resolution of IDAC. From (3.1), (3.2) can be derived by using two relations: $T_{C1}=C_{R1}(V_{REF2}+\gamma)/I_{REF1}$ and $T_{C2}=C_{R2}V_{REF2}/I_{REF2}$.

$$I_{REF2} \sim \frac{I_{REF1}}{N} \left(1 - \frac{\gamma}{V_{REF2}} - \frac{T_{D1} - T_{D2}/N - Q/N}{T_{C1} - \epsilon} \right). \tag{3.2}$$

In (3.2), $T_{D1}=T_{CD1}+T_{LD1}$, $T_{D2}=T_{CD2}+T_{LD2}$, and $\varepsilon=C_{R1}\gamma/I_{REF1}$.

The error ranges in Table 3.1 can simplify (3.2). According to Table 3.1, the maximum value of α is 5.71 mV, whereas the minimum value of β is 1.69 mV. Consequently, the upper bound of γ

Table 3.1: Ranges of Errors From the Circuit Nonidealities (-20 °C ~ 80 °C; TT, FF, SS, FS, and SF corner cases with mismatches; 1.8 V). ©2020 IEEE.

Error	Min	Max
Charging current error in the current-to-time converter for I_{REF1} (V_{REF}-referred)	3.29 mV	5.66 mV
Comparator offset in the current-to-time converter for I_{REF1} (V_{REF}-referred)	-137 μV	120 μV
Error from nonzero V_{IN} when I_{REF1} starts to charge C_{R1} (V_{REF}-referred)	-113 μV	-69 μV
Charging current error in the current-to-time converter for I_{REF2} (V_{REF}-referred)	2.10 mV	4.08 mV
Comparator offset in the current-to-time converter for I_{REF2} (V_{REF}-referred)	-388 μV	183 μV
Error from nonzero V_{IN} when I_{REF2} starts to charge C_{R2} (V_{REF}-referred)	-15 μV	41 μV
T_{D1}	120 ns	256 ns
T_{D2}	375 ns	1073 ns
T_{D1}/T_{C1} @ 30 °C	0.015	0.020
Q (I_{LSB} of IDAC < 0.1 % of a target current)	0	0.001*T_{C2}

is 4.01 mV, which is 0.78 % of V_{REF} when V_{REF} is the minimum (461.8 mV in the SF corner case). Therefore, T_{C1}-ε can be approximated by T_{C1} because the upper bound of ε/T_{C1} (=γ/(V_{REF2}+γ)) is 0.78 % considering that V_{REF2}+γ is larger than V_{REF}. From (3.2), the temperature coefficient of I_{REF2} can be derived with an equation $I_{REF2,TC}$=(1/I_{REF2})×($\partial I_{REF2}/\partial T$). Afterwards, the derived equation can be simplified as below in that T_{D1}/T_{C1} is much larger than (T_{D2}+Q)/NT_{C1} when N=100 as Table 3.1 shows.

$$
\begin{aligned}
I_{REF2,TC} = {} & (1 - 2T_{D1}/T_{C1} - \gamma/V_{REF2}) \times I_{REF1,TC} \\
& - (1/T_{C1})(\partial(T_{D1} - T_{D2}/N - Q/N)/\partial T) \\
& - (1/V_{REF2})(\partial\gamma/\partial T) \\
& + (\gamma/V_{REF2})V_{REF2,TC} \\
& + (T_{D1}/T_{C1})((V_{REF2} + \gamma)_{TC} + C_{R1,TC}).
\end{aligned}
\tag{3.3}
$$

T is temperature. $I_{REF1,TC}$, $V_{REF2,TC}$, (V_{REF2}+γ)$_{TC}$, and $C_{R1,TC}$ are the temperature coefficients of

Table 3.2: Temperature Coefficient Ranges ($\mu \pm 3\sigma$) of the Circuit Nonidealities. ©2020 IEEE.

Items	TT (ppm/°C)		FF (ppm/°C)		SS (ppm/°C)		FS (ppm/°C)		SF (ppm/°C)	
	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
Charging current error in the current-to-time converter for I_{REF1}	3.81	3.90	4.29	4.65	3.47	3.54	3.39	3.52	3.75	4.14
Comparator offset in the current-to-time converter for I_{REF1}	0.99	2.02	1.38	2.24	0.87	1.65	0.76	1.47	1.10	2.65
Error from nonzero V_{IN} when I_{REF1} starts to charge C_{R1}	0.48	0.56	0.43	0.53	0.55	0.64	0.43	0.49	0.56	0.67
Charging current error in the current-to-time converter for I_{REF2}	2.25	4.26	1.11	10.40	2.31	2.78	1.90	5.12	2.36	3.65
Comparator offset in the current-to-time converter for I_{REF2}	3.17	3.62	4.03	5.49	2.77	3.00	2.59	2.93	3.95	4.58
Error from nonzero V_{IN} when I_{REF2} starts to charge C_{R2}	0.01	0.04	0.08	0.14	0.04	0.07	0.03	0.06	-0.01	0.02
$(1/T_{C1}) \times (\partial T_{D1}/\partial T)$	1.23	2.80	0.85	3.20	0.81	3.47	1.59	3.50	2.93	6.08
$(1/NT_{C1}) \times (\partial T_{D2}/\partial T)$	2.17	2.66	2.78	4.20	2.44	2.68	1.96	2.34	2.39	3.07
$(1/NT_{C1}) \times (\partial Q/\partial T)$	0.00	10.00	0.00	10.00	0.00	10.00	0.00	10.00	0.00	10.00
Summation of the above temperature coefficients (upper bound)	**14.12**	**29.87**	**14.95**	**40.84**	**13.26**	**27.83**	**12.65**	**29.43**	**17.04**	**34.85**
V_{REF}	26.64	30.44	23.33	27.69	38.82	42.33	28.14	32.64	13.68	23.34
$V_{REF2}+\gamma$ ($V_{REF}+\alpha$, upper bound)	31.92	36.93	29.43	35.11	43.70	48.16	32.72	38.12	19.09	30.79
V_{REF2} ($V_{REF}+\beta$, upper bound)	32.08	38.36	28.55	43.71	43.93	48.17	32.67	40.75	19.99	31.58

I_{REF1}, V_{REF2}, $V_{REF2}+\gamma$, and C_{R1} respectively.

The accuracy of the proposed current reference can be analyzed based on (3.3). According to Table 3.1, the maximum value of T_{D1}/T_{C1} is 0.02. Furthermore, the upper bound of γ/V_{REF2} is 0.0078 as discussed earlier. As a result, the minimum bound of the first term in (3.3) is 95.2 % of $I_{REF1,TC}$. (3.3) can be further simplified by using the simulation results in Table 3.2, which summaries the temperature coefficient ranges of each nonidealities based on 50 Monte-Carlo mismatch simulations at each temperature and each corner case when the supply voltage is 1.8 V. In Table 3.2, an addition of the temperature coefficients of the first three items to the temperature coefficient of V_{REF} results in the upper bound of the temperature coefficient of $V_{REF2}+\gamma$, provided that the temperature variations do not cancel each other. The upper bound of the temperature coefficient of V_{REF2} can be obtained in the same way by adding the temperature coefficients of the V_{REF}-referred nonidealities in the current-to-time converter for I_{REF2} to the temperature coefficient of V_{REF}. Although the maximum value of $V_{REF2,TC}$ is 48.17 ppm/°C, the value is scaled by γ/V_{REF} in (3.3), leading to 0.38 ppm/°C. Likewise, the maximum values of $(T_{D1}/T_{C1}) \times (V_{REF2}+\gamma)_{TC}$ and

$(T_{D1}/T_{C1}) \times (C_{R1,TC})$ are 0.96 ppm/°C and -0.59 ppm/°C respectively. Accordingly, the last two terms in (3.3) can be neglected. The second and the third terms in (3.3) are the major sources of the error that makes a difference between $I_{REF1,TC}$ and $I_{REF2,TC}$. Although the two terms have explicit negative signs, $I_{REF2,TC}$ can be larger than $I_{REF1,TC}$, depending on the polarities of $\partial I_{REF1}/\partial T$, $\partial T_{D1}/\partial T$, $\partial T_{D2}/\partial T$, $\partial Q/\partial T$, and $\partial \gamma/\partial T$ at each temperature. To calculate an upper bound of the difference between $I_{REF1,TC}$ and $I_{REF2,TC}$, the absolute values of the second and the third terms are added to the first term. For the same reason, the second and the third terms are evaluated by adding the temperature coefficients of related items in Table 3.2. $I_{REF2,TC}$ can be degraded up to 40.84 ppm/°C, compared to $I_{REF1,TC}$ when the 1-LSB current of IDAC is less than or equal to 1 pA.

3.5 Calibration Algorithm and Time Frame

As discussed in Section 3.2, the calibration engine selects a better code for the leakage-based IDAC based on its algorithm. The algorithm used in this paper is a binary search algorithm. Since the algorithm requires only 27 iterations when the IDAC has a 27-bit input, the engine can finish the calibration quickly while it does not sacrifice the accuracy of the calibration. Fig. 3.6(a) illustrates the binary search algorithm using a simple example, where the IDAC only has a 3-bit input. The horizontal axis represents the 3-bit input of the IDAC, whereas the vertical axis stands for the output current of the IDAC (I_{REF2}). To prove that the algorithm works well even when there are current redundancies between arrays as explained in Section 3.3.1, it is assumed that there is a current overlap between the most significant bit (MSB) and the last of the bits in the example.

The algorithm starts from the MSB and tries 1'b1 for the MSB. In this example, it assigns 3'b100 to the IDAC and waits for 100 clock cycles, which take around 2 ms, to settle the output current of the IDAC. Afterwards, the algorithm changes *PRESET* and *RST* in Fig. 3.5 to 1'b0. A counter in the calibration engine counts the rising and the falling edges of the clock until *DONE* in Fig. 3.5 equals to 1'b1 or until the value of the clock counter reaches $N+4$. The algorithm regards the value of the clock counter as a measured ratio between I_{REF1} and I_{REF2} (R_{MEAS}). If R_{MEAS} is larger than N, the algorithm sets the bit that it focuses on as 1'b1. Otherwise, it fixes the bit as 1'b0. In this example, I_{REF2} is smaller than I_{REF1}/N, which means R_{MEAS} is larger than N. For
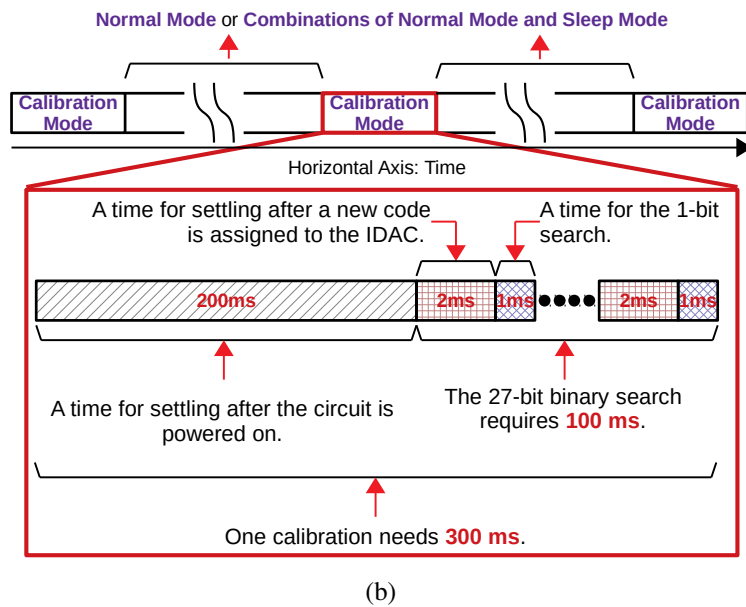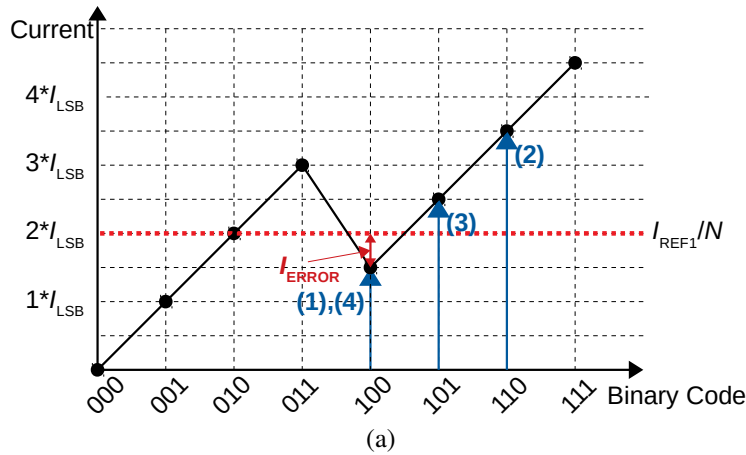
Figure 3.6: Operation of the automatic calibration. (a) Algorithm (binary search). (b) Time frame. ©2020 IEEE.

that reason, the algorithm chooses 1'b1 for the MSB. Subsequently, the algorithm tries 1'b1 for the next MSB and assigns 3'b110 to the IDAC. After following the same process, the algorithm obtains R_{MEAS} smaller than N. In the end, it sets the second MSB as 1'b0 and moves to the next MSB. The algorithm repeats the same process until it determines the least significant bit (LSB). In this example, the finial code value is 3'b100.

In Fig. 3.6(a), the intermediate calibration result converges to the ideal value (I_{REF1}/N) as the algorithm proceeds. Even when R_{MEAS} equals to N, the algorithm keeps reducing its intermediate

calibration result to get better accuracy instead of stopping the process. Since the actual current ratio (R_{ACT}) is always smaller than or equal to R_{MEAS} as mentioned in Section 3.2, I_{REF2} should be reduced until R_{ACT} is larger than N. Therefore, the intermediate calibration result converges to one of the points that have a difference between I_{REF1}/N and I_{REF2} (I_{ERROR}) smaller than I_{LSB}. The current overlap between the MSB and the LSBs does not affect the accuracy of the calibration as Fig. 3.6(a) shows.

Fig. 3.6(b) shows a time frame for the operation of the proposed current reference. When the proposed circuit moves to the calibration mode, all circuit blocks for the calibration are powered on. It takes time for all node voltages and currents to settle down before the calibration starts. Simulation results show that 200 ms are enough to acquire a settling accuracy better than 0.33 % even in the worst case (SS, -20 °C). Since a calibration error caused by the finite settling accuracy is smaller than 1 pA in simulations, the actual calibration can start after 200 ms. Since each chip can have a different clock frequency due to the process variations of I_{REF1}, the required calibration time varies by chip. In the TT corner case, it takes approximately 3 ms for the algorithm to determine the correct value for one bit. Therefore, the entire calibration spends about 81 ms (3 ms×27). Even in the worst corner case (SS), the algorithm completes the entire calibration within 100 ms. Overall, one calibration takes 300 ms or shorter. Based on the calibration time needed, the total power consumption of the proposed circuit can be calculated like below when the calibration mode is activated every 5 minutes and when the proposed circuit is in the normal mode between two calibration trials.

$$\text{Power consumption of the proposed current reference} = x/1000 + y \times 999/1000. \qquad (3.4)$$

x is the power consumption in the calibration mode, and y is the power consumption in the normal mode. As (3.4) indicates, the power consumption in the calibration mode is reduced by 1000 times. It can go down further if the calibration mode is activated less frequently than every 5 minutes.
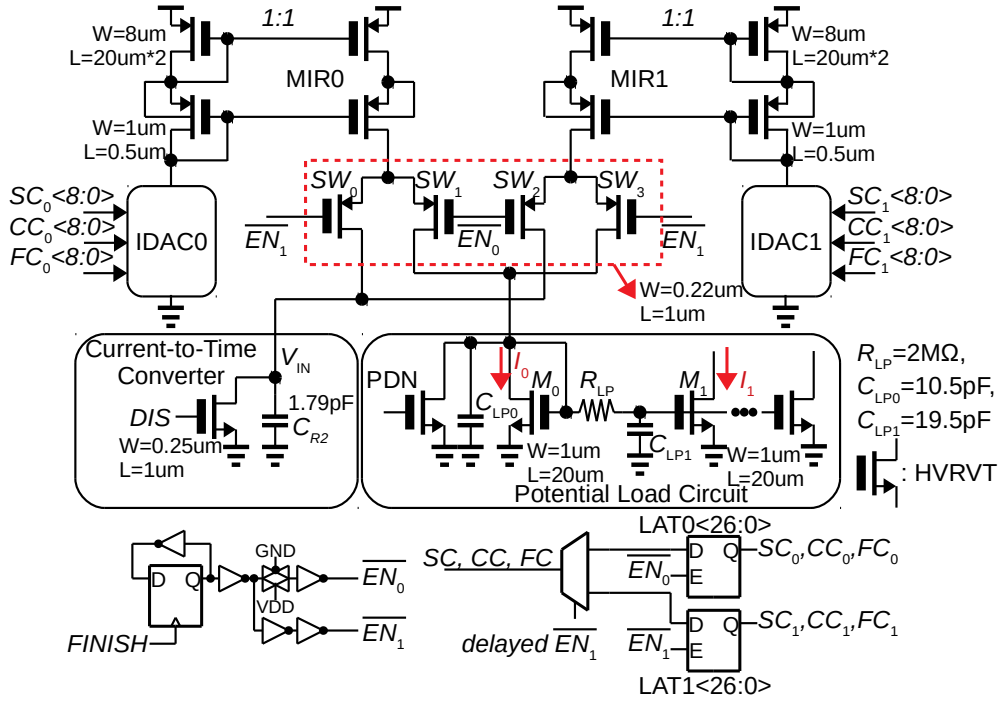
Figure 3.7: Current-providing mechanism for always-on load circuits. ©2020 IEEE.

3.6 Current-Providing Mechanism

When an application needs a continuous current with no interruption for always-on circuits, two IDACs (IDAC0 and IDAC1), two current mirrors (MIR0 and MIR1), and four switches ($SW_0 \sim SW_3$) can be utilized for the purpose as Fig. 3.7 shows. Since the load circuit is always on, the digital power-down signal (PDN) is always 1'b0 except when the load circuit is turned on for the first time. If EN_0=1'b1, IDAC0 supplies a current to the potential load circuit through MIR0, whereas IDAC1 provides a current for the current-to-time converter through MIR1. Since a 27-bit latch (LAT0) holds the input signals of IDAC0 (SC_0, CC_0, and FC_0), I_0 can be supplied continuously in the sleep mode and in the calibration mode of the current reference. In the calibration mode, the calibration engine calibrates the current of IDAC1 by controlling the inputs of IDAC1 through the demux and LAT1. When the calibration is complete, the engine assigns 1'b1 to *FINISH*. Accordingly, EN_0 and EN_1 change their polarities. As a result, IDAC1 takes over the role of supplying a current to the potential load circuit. The input signals of IDAC0 are set as 9'd0 when

39

the proposed current reference goes to the sleep mode again.

C_{LP0} and a low-pass filter that consists of R_{LP} and C_{LP1} in Fig. 3.7 can minimize the effect of the switching activities of $SW_0 \sim SW_3$ on the accuracy of I_1 as proposed in [39]. From the simulation results, the worst-case peak-to-peak fluctuation of I_1 is 0.9 fA (SS, -20 °C) when a replica of the diode-connected PMOS devices in MIR0 is connected to the drain of M_1. The calibration activities for IDAC0 or IDAC1 also do not affect the accuracy of I_1 significantly. When 1 nA from IDAC0 or IDAC1 charges C_{R2} during 1 ms and when the full-custom logic in Fig. 3.2(a) discharges C_{R2} to ground after 1 ms, the worst-case peak-to-peak fluctuation of I_1 is 0.2 fA (TT, 80 °C). If the input code value of IDAC0 or IDAC1 changes from the minimum code to the maximum code or from the maximum to the minimum in the calibration mode, the worst-case peak-to-peak fluctuation of I_1 is 35.0 fA (FS, 80 °C).

If the proposed current reference provides a current for duty-cycled circuits, IDAC1, MIR1, SW_2, SW_3, and the logics on the bottom of Fig. 3.7 are not necessary. Instead, a d-type retention flip flop can enable SW_1 and disable SW_0 by storing 1'b1 on its output at a rising edge of *FINISH*. Since the flip flop maintains its output until *RST_N_DIGENGINE* goes low, SW_1 can connect MIR0 to the duty-cycled load circuit in the normal mode. Afterwards, *PDN* can be set as 1'b0 to wake up the load circuit. The worst-case 0.1 % settling time of I_1 is 29.1 ms (SS, -20°C). The calibration mode can be activated during the time when *PDN* is 1'b1. If there is a ±10 mV variation on the supply voltage, the worst-case fluctuation of I_1 at 1 Hz is ±0.6 pA (SS, -20 °C) when an ac ground is connected to the drain of M_1. From 1 Hz to 100 MHz, the worst-case maximum variation of I_1 is ±10.5 pA. The integrated noise of I_1 from 1 Hz to 100 KHz is 1.8 pArms in the worst case (FF, 80 °C). 86 % of the integrated noise power comes from the thermal noise of M_1.

3.7   Measurement Results

3.7.1   Static Accuracy

Prototype chips are fabricated using a 180 nm CMOS technology. 10 chips, which come from three different wafers in one lot, are tested. Each chip is evaluated in a temperature range having
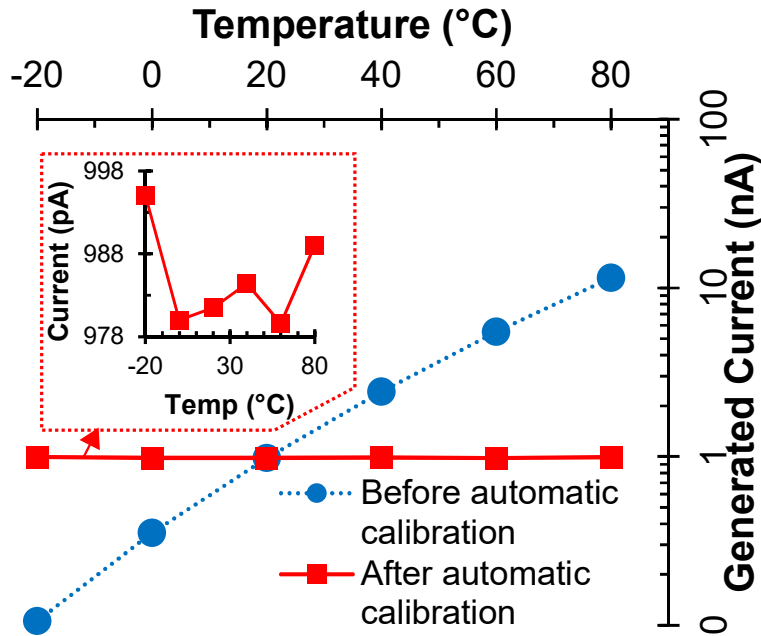
Figure 3.8: Effect of the automatic calibration on the generated reference current. ©2020 IEEE.

20 °C steps from -20 °C to 80 °C when the supply voltage is 1.8 V. When the current reference is in the normal mode after calibration, it is assumed that the temperature of the circuit is stable enough to evaluate the static accuracy of the current reference at each temperature. The dynamic accuracy of the current reference is discussed in Section 3.7.2. The static accuracy of each chip is also measured at three different supply voltages when the temperature is 20 °C: 1.5 V, 1.8 V, and 2.0 V. 32 automatic calibrations at each temperature and each supply voltage lead to a noise-averaged calibration result, suppressing calibration noise, which is a part of dynamic accuracy of the current reference. For the testing purpose, a Field Programmable Gate Array (FPGA) board initiates each calibration by releasing the reset signals (*RST_I2T_HP* and *RST_N_ENGINE*) in Fig. 3.2(a). When the board receives the finish signal (*FINISH*) from a chip being tested, the board reads a digital code for the IDAC and initiates the next automatic calibration after 1 ms. After 32 calibrations, a median code among the first 31 records is selected. The generated reference current is measured after the median code is externally assigned to the IDAC by the FPGA board.

Fig. 3.8 shows a generated reference current measured from one sample chip. When the automatic calibration is not activated at each temperature, the generated current shows 1160 % variation
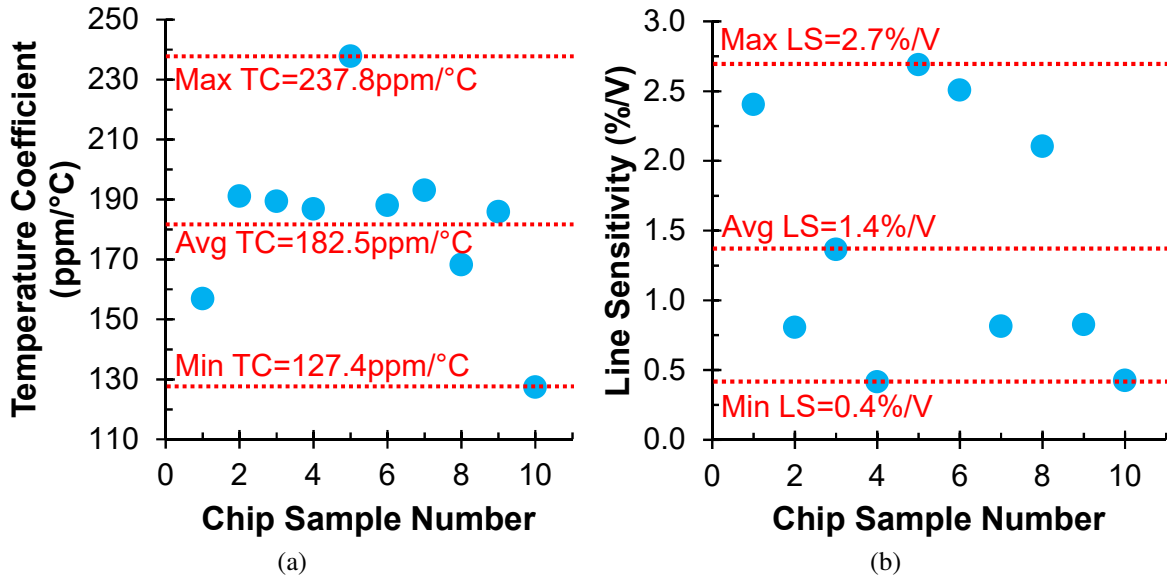
Figure 3.9: Accuracy of the generated reference current after the automatic calibration when *N*=86. (a) Accuracy over process and temperature variation. (b) Accuracy over process and supply voltage variation. ©2020 IEEE.

from the current at 20 °C. The reason for the large variation is that the input code for the IDAC is not adjusted depending on the temperature change. However, the generated reference current only has 1.57 % variation from the current at 20 °C after the automatic calibration is performed at each temperature. Therefore, this result proves that the entire calibration circuit works effectively from -20 °C to 80 °C.

The accuracy of the generated reference currents (I_{REF2}) over PVT variation can be checked from Fig. 3.9(a) and Fig. 3.9(b). All measured results in the figures are obtained without a chip-to-chip (and wafer-to-wafer) trimming. Each dot represents a temperature coefficient or a line sensitivity of each chip when the automatic calibration is activated at each temperature and supply voltage. On average, the generated reference current has 1.825 % variation in the 100 °C temperature range, which is equivalent to 182.5 ppm/°C. All temperature coefficients from 10 chips are in the range from 127.4 ppm/°C to 237.8 ppm/°C. The average temperature coefficient of I_{REF1} is 156.1 ppm/°C. The average difference between $I_{REF1,TC}$ and $I_{REF2,TC}$ is 26.4 ppm/°C, which is smaller than the upper bound analyzed in Section 3.4. The average line sensitivity of the generated
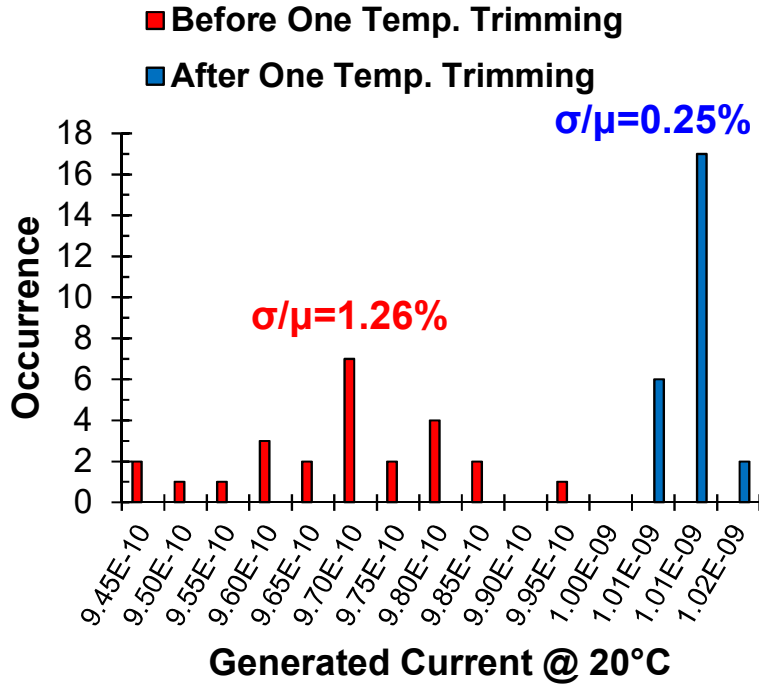
Figure 3.10: Auto-calibrated reference current spread of 25 sample chips at 20 °C before and after room-temperature trimming. ©2020 IEEE.

reference current is 1.4 %/V, which is 0.7 % variation from 1.5 V to 2.0 V. All line sensitivities from the 10 chips are spread in the region from 0.4 %/V to 2.7 %/V.

Fig. 3.10 represents the statistics of measured reference currents (I_{REF2}) at 20 °C when the automatic calibration is activated at the temperature. As the red bars show, 25 chips have ± 3.78 % variation ($\pm 3\sigma/\mu$) when N is 86 for all chips and when no chip-to-chip (or wafer-to-wafer) trimming is applied. However, the spread decreases to ± 0.75 % (blue bars) if each chip is trimmed at room temperature. The trimming is accomplished with the largest N, which makes the auto-calibrated reference current larger than 1 nA. Since the trimming changes only N and since the auto-calibrated reference current monotonically increases as N decreases, we can finish the trimming process quickly without using complex algorithms. Among 25 chips, 10 chips are fully evaluated from -20 °C to 80 °C. As Fig. 3.11(a) and Fig. 3.11(b) indicate, the trimming process does not profoundly affect the temperature coefficient of the reference current. In addition, the proposed circuit can generate various amounts of a reference current by changing N as Fig. 3.12
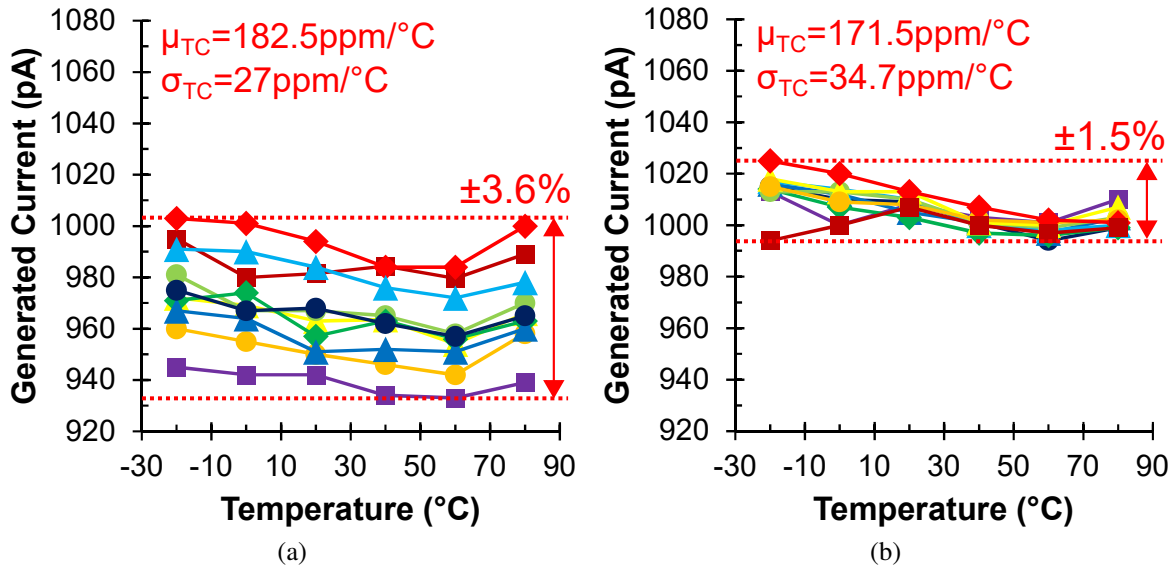
Figure 3.11: Auto-calibrated reference current before and after room-temperature trimming. (a) Spread of 10 sample chips from -20 °C to 80 °C before room-temperature trimming. (b) Spread of 10 sample chips from -20 °C to 80 °C after room-temperature trimming. ©2020 IEEE.

shows.

3.7.2 Dynamic Accuracy

In case the proposed current reference is in the normal mode for a long time (e.g. 25 minutes), changes in temperature and supply voltage during the time can affect the accuracy of the generated current. Two factors can cause temperature variations: ambient temperature change and on-die thermal conduction. According to [40], the average diurnal air temperature range in the United States in 2012 is 13.5 °C. If we assume that the temperature change occurs for 12 hours from early morning to late afternoon, the rate of the temperature change is approximately 0.1 °C per 5 minutes. In Nevada, where there are many deserts, diurnal air temperature ranges do not normally exceed 20 °C as [41] shows. Even when daily record high and low occur on the same day, diurnal temperature ranges are around 33 °C [42], which is approximately 0.1 °C change for 2.5 minutes. However, indoor temperature variation rates are much smaller than the rate of 0.1 °C per 5 minutes. [43] shows daily indoor temperature profiles during two seasons in three different climate regions. In the cooling season (from May to September for Florida homes and from June to

44
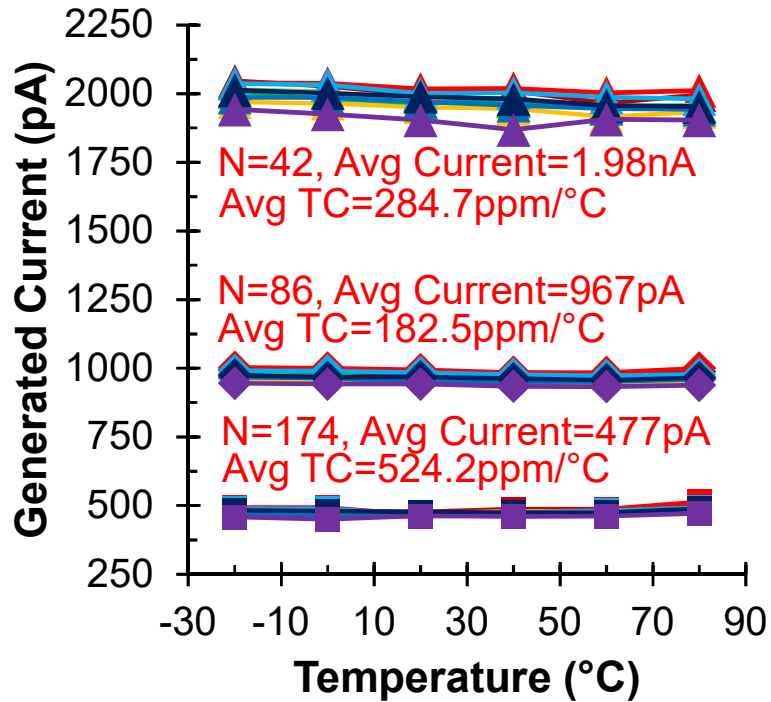
Figure 3.12: Generated reference current for various N (N=42, N=86, N=174). ©2020 IEEE.

August for New York homes), the temperature variation rate is 0.06 °C and 0.07 °C per 25 minutes for Florida and New York homes respectively. The rate increases to approximately 0.1 °C per 25 minutes during the heating season for New York and Oregon/Washington homes. Therefore, the proposed current reference can experience 0.1 °C temperature change for 25 minutes when it is used for indoor low-power electronics such as smart thermostats. For low-power mobile applications, the reference can be exposed to on average 0.1 °C temperature change for 5 minutes. In the extreme cases mentioned before, the same temperature change can occur for 2.5 minutes.

Fig. 3.13 shows the measured dynamic accuracy of the current reference when there are temperature variations at two extreme starting temperatures: -20 °C and 80 °C. The red arrows represent calibration noise around -20 °C, whereas the green arrows show maximum current deviations caused by the temperature variations at -20 °C in the normal mode. The y-axis value of each data point is defined by the difference between the current at each time point and the current at time 0. When the calibration cycle is 5 minutes to compensate 0.1 °C ambient temperature change per 5 minutes for mobile applications, the average maximum current deviation is 0.5 % at -20 °C
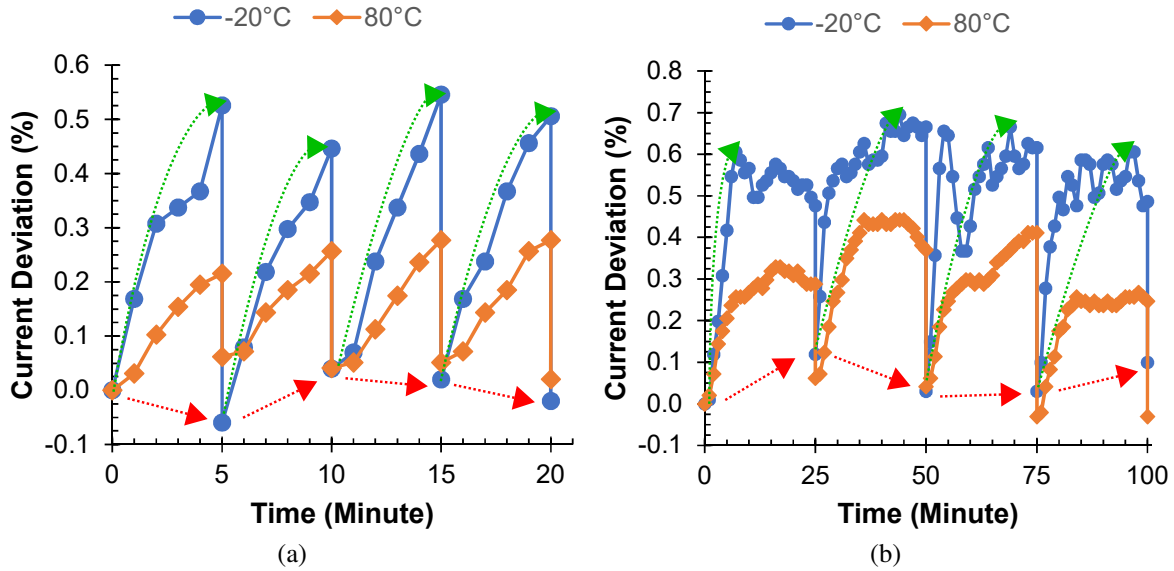
Figure 3.13: Accuracy of the current reference when there are ambient temperature variations. A red arrow represents calibration noise at -20 °C, whereas a green arrow shows the maximum current deviation in each normal mode at -20 °C. (a) 0.1 °C temperature change during 5 minutes and 5-minute calibration cycle. (b) 0.1 °C temperature change during 25 minutes and 25-minute calibration cycle. ©2020 IEEE.

as Fig. 3.13(a) represents. The deviation is equivalent to 50 ppm/°C accuracy degradation, if it is assumed that the same deviation occurs from -20 °C to 80 °C although the actual deviation is smaller than 0.5 % at higher temperatures (0.22 % at 80 °C). If the calibration cycle is reduced to 2.5 minutes for extreme outdoor applications that have 0.1 °C ambient temperature change per 2.5 minutes, the maximum deviation is 0.36 % at -20 °C. As Fig. 3.13(b) shows, the maximum deviation can increase up to 0.6 % at -20 °C when the calibration cycle is 25 minutes for indoor electronics that are exposed to 0.1 °C ambient temperature change per 25 minutes.

Fig. 3.14 quantifies the effect of calibration noise on the dynamic accuracy of the current reference. The horizontal axis represents the deviations of the auto-calibrated reference currents from an average current calculated at each temperature. The vertical axis shows the number of deviations out of 101 calibration trials. The worst deviation is ±0.23 % at 80 °C considering ±3 standard deviations obtained from the measured statistic in Fig. 3.14. Therefore, even if the worst deviation occurs at all temperatures, the temperature coefficient of the generated current increases to only
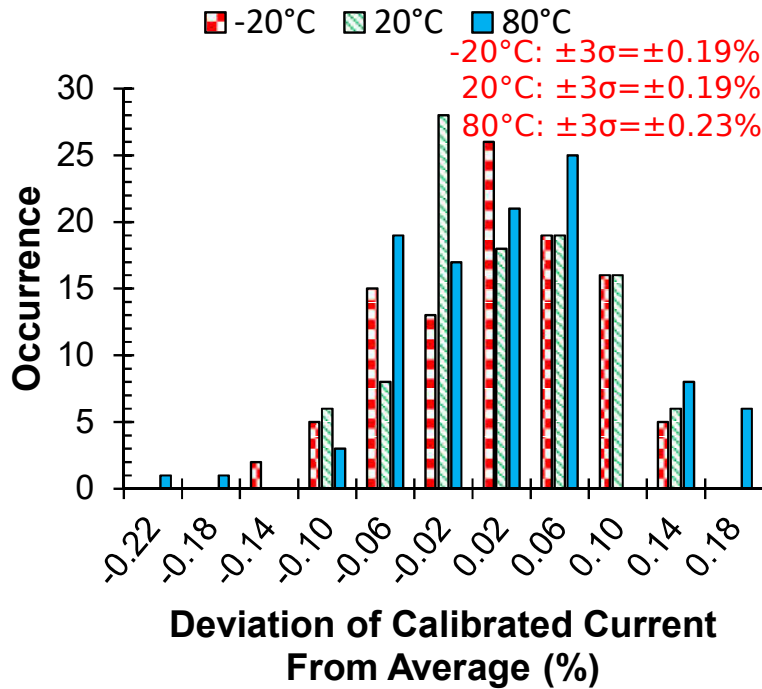
Figure 3.14: Deviation of the reference currents generated by 101 calibration trials at three different temperatures. ©2020 IEEE.

46 ppm/°C.

The other factor that contributes to temperature variations of the current reference is on-die thermal conduction. To prove that the thermal conduction effect is not significant, the following experiment is performed at two extreme temperatures: -20 °C and 80 °C. In the experiment, an FPGA board generates on-die heat by maintaining the calibration mode of the current reference continuously for 25 minutes instead of moving the reference to the normal mode periodically. Since the building blocks of the reference, excluding IDAC, consume μW-range power in the calibration mode, the effect of μW-range thermal conduction can be observed if we compare the current of IDAC recorded before the 25-minute calibration with the current obtained after the calibration by assigning the same input code for IDAC. There are approximately 0.3 pA and 0.1 pA differences between the two currents at -20 °C and 80 °C respectively. Therefore, integrating the proposed reference with other low-power systems such as [44] does not significantly degrade the accuracy of the reference when the low-power systems consume nW power in their active modes.

47

Table 3.3: Power Consumption of the Current Reference at Three Temperatures. ©2020 IEEE.

Temp.	Power in Normal Mode	Power in Calibration Mode	Total Average Power (25-min Cycle Calibration)	Total Average Power (5-min Cycle Calibration)	Total Average Power (2.5-min Cycle Calibration)
-20 °C	3.00 nW	5.04 µW	4.3 nW (reference: 4.01 nW, wake-up timer: 0.25 nW*, counter & logics: 0.02 nW**)	8.3 nW (reference: 8.04 nW, wake-up timer: 0.25 nW*, counter & logics: 0.02 nW**)	13.4 nW (reference: 13.08 nW, wake-up timer: 0.25 nW*, counter & logics: 0.02 nW**)
23 °C	3.09 nW	5.30 µW	**4.5 nW (reference: 4.15 nW, wake-up timer: 0.28 nW*, counter & logics: 0.03 nW**)**	**8.7 nW (reference: 8.38 nW, wake-up timer: 0.28 nW*, counter & logics: 0.03 nW**)**	**14.0 nW (reference: 13.7 nW, wake-up timer: 0.28 nW*, counter & logics: 0.03 nW**)**
80 °C	6.20 nW	6.50 µW	8.5 nW (reference: 7.50 nW, wake-up timer: 0.63 nW*, counter & logics: 0.34 nW**)	13.7 nW (reference: 12.7 nW, wake-up timer: 0.63 nW*, counter & logics: 0.34 nW**)	20.2 nW (reference: 19.2 nW, wake-up timer: 0.63 nW*, counter & logics: 0.34 nW**)

* Estimated from [46] ** Simulation

The effect of supply voltage variations on the accuracy of the current reference in the normal mode is not remarkable thanks to the large output impedance of each leakage source in Fig. 3.3. Measurement results show that there are 0.32 % and 0.37 % reference current variations in the normal mode at -20 °C and 80 °C respectively when the supply voltage changes from 1.85 V to 1.75 V. If a 1-cm^2 full-cell 8-layer Li-Ion microbattery proposed in [45] provides the 1.8 V supply voltage for the proposed current reference, the battery can maintain the supply voltage accurately at least up to 1 mAh/cm^2 as Fig. 3(e) in [45] shows. Therefore, in this scenario, the supply voltage variation effect is negligible because the supply voltage drop is much smaller than 100 mV for more than 19 years considering that the reference and its wake-up timing circuits consume 5.8 nA when the calibration mode is activated every 5 minutes.

3.7.3 Power Consumption

Since this work does not integrate circuits that control the timing of the calibration and is tested with an FPGA board, the total power consumption of the whole system is estimated based on a wake-up timer proposed in [46]. When the calibration mode is activated every 25 minutes

or shorter, the timing of each mode can be controlled by the wake-up timer, an 18-bit ripple counter, 30 logic gates, and two flip flops. The following elaborates the operation of the timing circuits. Since the timer generates a 90-Hz clock, the value of the counter reaches 18'h20F58 from 18'h00000 after 25 minutes. At the moment, logic gates and a flip flop assign 1'b1 to *PEN_I2T_HP*, *PEN_I2T_LP*, and *PEN_ENGINE* in Fig. 3.2(a). Logic gates and another flip flop give a calibration start signal to the current reference by disabling *RST_I2T_HP* and by toggling *RST_N_DIGENGINE* from 1'b1 after 200 ms (18 clock cycles). The flip flops maintain their values until the calibration engine in Fig. 3.2(a) assigns 1'b1 to *FINISH*. Since the *FINISH* signal resets the 18-bit counter as well, the current reference moves back to the normal mode and stays in the mode for 25 minutes again. When the supply voltage is 1.5 V, the timer consumes approximately 280 pW at room temperature according to [46]. Simulation results show that other logics, including the counter and the flip flops, consume 27.2 pW in the nominal corner case (TT). Therefore, the total power consumption overhead needed for the timing control is 307.2 pW at room temperature. The estimate of the area overhead is 0.063 mm^2 (0.057 mm^2 for the wake-up timer and 0.006 mm^2 for other logics).

Table 3.3 summarizes the measured power consumption of the proposed current reference in two modes at three temperatures when the supply voltage is 1.5 V. The total average power consumption at each temperature is estimated based on three scenarios, where the calibration mode is activated every 25 minutes, 5 minutes, or 2.5 minutes. In the normal mode, I_{REF2} in Fig. 3.2(a) is close to 1 nA. Accordingly, the power consumption of the circuit in the normal mode is larger than 3 nW. Although the proposed circuit consumes 5.30 μW in the calibration mode at 23 °C, the total estimated average power consumption is 4.5 nW when the automatic calibration is initiated every 25 minutes for indoor low-power applications. The average power consumption is drastically reduced from the power consumption in the calibration mode because the proposed circuit stays in the normal mode most of the time as discussed in Section 3.5.
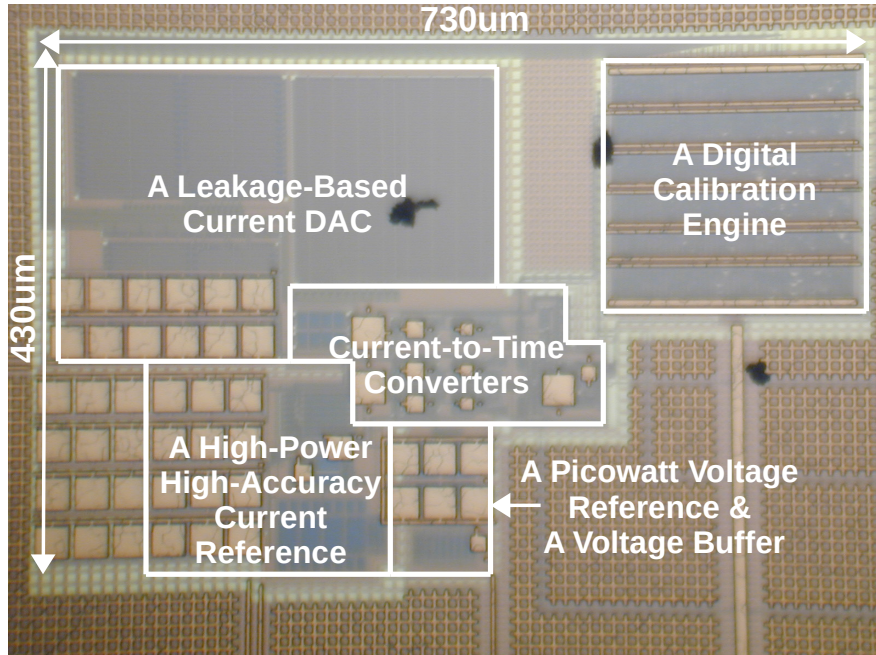
Figure 3.15: Die photograph. ©2020 IEEE.

3.7.4 Die Photograph and Comparison Chart

Fig. 3.15 shows a die photograph of the proposed current reference. The area of the circuit is 0.269 mm^2, including the area of the MIM capacitors. Table 3.4 compares the proposed current reference with other low-power current references that generate currents smaller than 1 µA. For a fair comparison, power consumption figures reported in [25, 30, 32] are recalculated. Compared to the current references in [22, 25, 30], which have proven measured results with no multiple-temperature trimming requirement, the proposed circuit consumes at least 74 % less power, while maintaining competitive high accuracy, with the calibration mode activated every 25 minutes to compensate a reference current variation caused by a 0.1 °C ambient temperature change. The current references in [31] and [33] are not directly comparable to the proposed circuit, because [31] and [33] only have simulation results, although they do not need multiple-temperature trimmings. The current reference in [32] is also hard to compare, since it utilizes a two-temperature trimming, although it achieves 1000 times smaller power consumption than this work, with comparable accuracy. All things considered, this work proves that the calibration scheme is an effective way to

Table 3.4: Comparision Chart. ©2020 IEEE.

Publication Title	This Work	SSCL [32]	ISCAS [31]*	ISCAS [33]*	ISSCC [30]	ESSCIRC [20]	ESSCIRC [25]	TCAS2 [22]
Year		2018	2016	2019	2017	2010	2017	2010
Technology	180 nm	65 nm	65 nm	180 nm	180 nm	350 nm	180 nm	350 nm
Reference Current (I_{REF})	**1 nA**	1.2 pA	5.0 pA	2.7 nA	6.6 nA	10.0 nA	35.0 nA	96.0 nA
Sleep Mode Support	**Yes**	No	No	No	No	No	No	No
Average Power Consumption @ Room Temperature	**4.5 nW (25-min cycle auto cal.), 8.7 nW (5-min cycle auto cal.), 14.0 nW (2.5-min cycle auto cal.)**	3.9 pW (includes an additional mirroring)	14.5 pW	26.0 nW	17.9 nW (includes an additional mirroring)	88.5 nW	53.5 nW (includes I_{REF})	1.0 μW
Temperature Range (°C)	-20~80	-20~60	0~100	-40~125	0~100	-20~80	-40~120	0~80
Multiple Temperature Trimming	No	Yes	Yes	No	Yes	Yes	No	No
TC Without Multiple Temp. Trimming (Static + Dynamic, ppm/°C)	**289 (183+106, 25-min cycle auto cal.), 279 (183+96, 5-min cycle auto cal.), 265 (183+82, 2.5-min cycle auto cal.)**	N/A	128	309	680	N/A	282	600
TC After Multiple Temp. Trimming (ppm/°C)	N/A	469	31	N/A	283	1190	N/A	N/A
Supply Voltage Range	1.5 V ~ 2.0 V	0.4 V ~ 1.5 V	0.5 V ~ 1.8 V	2.0 V ~ 3.6 V	1.3 V ~ 1.8 V	1.3 V ~ 3.0 V	1.5 V ~ 2.5 V	1.8 V ~ 3.0 V
Line Sensitivity	**1.40 %/V**	2.5 %/V	0.95 %/V	9.4 %/V	1.16 %/V	3.00 %/V	0.13 %/V	0.20 %/V
Area (mm²)	0.332 (reference: 0.269, timer & counter logics: 0.063)	0.008	0.0002	0.009	0.055	0.120	0.017	0.015
Number of Samples	10	13	N/A	N/A	10	15	32	4

* Simulation

relax the trade-off between the power consumption and the accuracy of a current reference, while investing on a large area, compared to other references. In addition, the proposed circuit needs a slow ambient temperature change lower than 0.1 °C for 2.5 minutes, which is valid for many applications as discussed in Section 3.7.2.

3.7.5 Range and One-Step Current Accuracy of IDAC

As discussed in Section 3.3.1, the range of IDAC should include a target current (1 nA) from -20 °C to 80 °C. Measurement results show that the IDAC generates 4.08 nA and 38.06 nA at -20 °C and 80 °C respectively when the input of the IDAC has the maximum code value (27'h7FF_FFFF). Furthermore, the IDAC outputs a current smaller than 100 fA and 60 pA at -20 °C and 80 °C respectively when the input code is the minimum (27'h000_0000). Therefore, the IDAC has enough margin in its output current range. Additionally, the IDAC should have small one-step currents (DNLs) when it generates a current close to the target current (1 nA). Since there are total 2^{27} possible input combinations, all input combinations cannot be tested in a reasonable time. Instead, we pick an input base code value that generates a current around 900 pA and check one-step currents by increasing the input code value across the boundaries between two adjacent binary-weighted leakage sources in Fig. 3.3. For example, at 80 °C, we select 27'h030_0000 as a base input code since it generates 898 pA. Afterwards, two output currents are compared when the input code increases from 27'h0030_0001 to 27'h0030_0002, from 27'h0030_0003 to 27'h0030_0004, from 27'h0030_0007 to 27'h0030_0008, and so on until the input code generates a current larger than 1 nA. In this experiment, the largest one-step currents are 2.2 pA and 1.1 pA at -20 °C and 80 °C respectively. Therefore, the quantization error of IDAC is smaller than 0.22 %.

3.8 Conclusion

This chapter shows a current reference that can generate 1 nA with high accuracy (289 ppm/°C, 1.4 %/V) and small power consumption (4.5 nW) when the calibration mode is enabled every 25 minutes to compensate a reference current variation caused by a 0.1 °C ambient temperature change in indoor low-power applications. By using the automatic calibration, the proposed circuit can relax the trade-off between the power consumption and the accuracy of a current reference with no help from a multiple-temperature trimming. On the other hand, the proposed circuit requires a large area and a slow ambient temperature change lower than 0.1 °C for 2.5 minutes.

4.   A SURVEY ON IN SITU ANALOG CIRCUIT OPTIMIZATION*

4.1   Design Centering

Process-Voltage-Temperature (PVT) variation and device aging have been one of the critical issues of analog circuits especially in modern technologies. To meet all specifications even in the worst-case scenario, the design centering technique has been utilized [47, 48]. When a customer explicitly gives specifications for circuit characteristics z_1 and z_2, we can draw the region of acceptable performance specifications (gray area) on a 2-dimensional performance space [48] as Fig. 4.1 shows. In Fig. 4.1, x-axis and y-axis represent metrics of z_1 and z_2, respectively, and larger z_1 and z_2 the better performance. After mapping the region of acceptable performance to a design parameter space, we can accomplish the design centering by finding a design point that maximizes the yield in the parameter space. Fig. 4.1 includes the design point obtained from the design centering. The z_1 and z_2 characteristics of a design placed at the design center can stay in the acceptable performance region even when severe PVT variations and device aging affect the design after chip fabrication. However, the design centering requires large margins. For example, in Fig. 4.1, z_1 and z_2 are higher than their minimum requirements at the design center, resulting in sacrificing other circuit characteristics such as power or area consumption. In some cases, the sacrifice can be prohibitively expensive.

4.2   Tuning/Calibration Methodologies and Their Limitations

To overcome the limitation of the design centering, one possible alternative is placing a design point close to an edge of an acceptable performance specification region such as design point A in Fig. 4.1. Since the design at A does not have an excessive margin for z_1 anymore, the design does not need to sacrifice other circuit characteristics such as power or area. However, severe PVT variations and device aging can move the actual operating point of the design to the outside of an
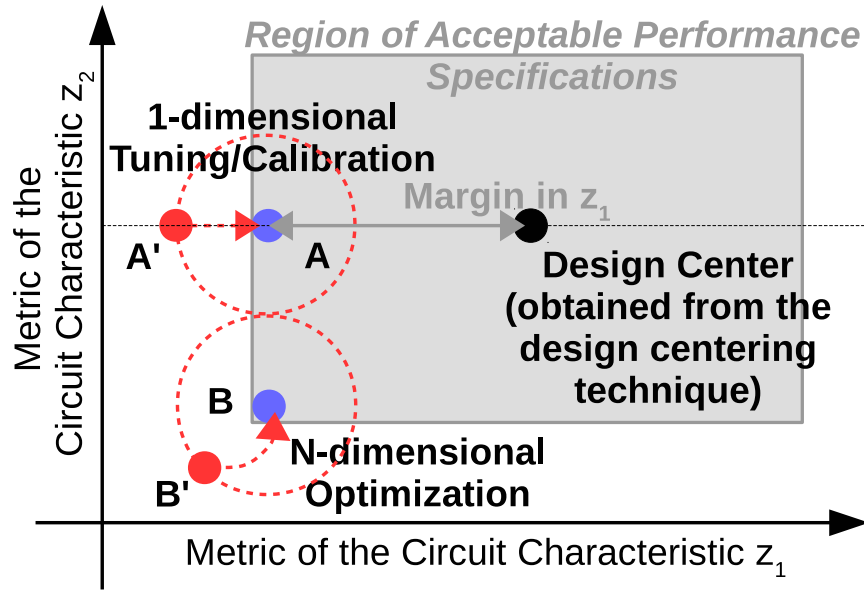
Figure 4.1: Comparison between a conventional 1-dimensional tuning/calibration and an N-dimensional optimization. ©2018 IEEE.

acceptable performance region like A'. A conventional approach to solve this issue is utilizing a tuning/calibration that relocates A' to the inside of the acceptable performance region.

One common tuning/calibration methodology is designing analog circuits with digitally controlled tuning knobs and using automatic test equipment (ATE) [49]. After chip fabrication, ATE can automatically check the validity of the analog circuits chip by chip and find an optimal control code for each chip. This code can be written in a ROM or eFUSEs and be retrieved when the chip is powered on. Unfortunately, ATE usually requires high setup cost and does not support complex/high-accuracy measurements for analog circuits.

In this context, many on-chip methodologies, including built-in self-tests (BISTs) and on-chip tuning/calibration, have been researched for various analog circuits [50]. We can categorize the methodologies into two groups depending on how to verify their target characteristics. The first category uses indirect measurements based on statistics [51–53]. Since the characteristics of an analog circuit are correlated with each other, a statistical model can predict target characteristics from other characteristics we can measure easily with a low cost. For example, [51] predicts the target characteristics of an RF power amplifier, such as gain, linearity, power consumption,

and power efficiency, from the DC bias current of the amplifier and the R/C values of its passive components. Even though this method can test and calibrate multiple analog circuit characteristics simultaneously from the minimal set of relatively simple measurements, it has several drawbacks. First, it has limited accuracy. Because of the statistical nature of this approach, there are deviations between predicted and actual characteristics. Due to these deviations, the calibrated circuit might not be in an optimal setting. Second, it is hard to fully integrate the statistical model on a chip because of the high complexity of the model. Therefore, an external computer is still needed, which makes an in-situ calibration impossible.

The other category utilizes on-chip direct measurements to evaluate target characteristics [54–60]. For example, [54] and [55] calibrate a matching network and the gain of a low-noise amplifier, respectively, by measuring signal power or amplitudes at various nodes. Each circuit proposed in [56–58] tunes Q or ω_O of an analog baseband filter by evaluating the amplitude/phase response of a replica (master) circuit or by measuring the RC time constant of a capacitor array. Although methods in this category can provide relatively accurate calibration results and in situ corrections, there are two major drawbacks. First, these methods can handle a very limited number of specifications and cannot make a balance between one and another. For example, although the circuits proposed in [56, 58] can tune Q and/or ω_O, the circuits cannot validate other circuit characteristics such as stability, power consumption, and settling time. Even if we can use dedicated tuning circuits for the characteristics, there is no systematic way to make a "balance" among multiple circuit characteristics that are in a trade-off relationship and to find the "optimal" control code for a circuit under test (CUT). This issue will be more evident when the CUT has a large programmability to support many standards and scenarios because the total number of possible combinations of control codes increases exponentially with the number of control bits. Second, some methods in this category require a replica circuit that is tuned instead of a main circuit. These methods can be problematic because recent technologies cannot guarantee a good matching between the two circuits anymore.

4.3 Motivation for In Situ Analog Circuit Optimization

In situ analog circuit optimization techniques can overcome the aforementioned limitations of the on-chip tuning/calibration methodologies. The optimization techniques evaluate multiple circuit characteristics by stimulating a circuit under optimization (CUO) and by monitoring the responses of the CUO. Afterwards, the techniques calculate the cost of a tuning knob setting applied to the CUO. After trying multiple tuning knob settings and evaluating their costs, the optimization techniques find the best tuning knob setting that minimizes the cost based on an algorithm each technique utilizes. Since the cost is a function of measured circuit characteristics, the optimization techniques can achieve a good balance among competing circuit characteristic goals depending on the definition of a cost function, resulting in a better trade-off among the circuit characteristics. Fig. 4.1 clearly shows the strength of analog circuit optimization techniques. If we employ one of analog circuit optimization techniques, we can place our design point near the bottom-left corner of the acceptable performance specification region in Fig. 4.1 like B. Since the design point does not have margins for z_1 and z_2 unlike A and the design center, we do not need to sacrifice other circuit characteristics, such as area and power consumption. When PVT variations and device aging move the actual operating point of the design to B', the N-dimensional optimization can relocate B' to the optimal operating point B, whereas the 1-dimensional tuning/calibration cannot move B' to B. Therefore, in situ analog circuit optimization techniques allow designers to position their design at B since the effects of the variations can be well compensated. In addition, since the techniques stimulate a CUO and capture the responses of the CUO directly, they normally do not require replica circuits.

4.4 Previous Works on In Situ Analog Circuit Optimization

There are only a few previous works on in situ analog circuit optimization. [61] optimizes the gain and the linearity of a power amplifier by analyzing the responses of the amplifier when a well-tailored signature input stimulates the amplifier. Since measuring the linearity of a power amplifier directly on a chip requires complex and high-performance circuits, utilizing a signature testing

can be a good option to relax the hardware requirement if it can guarantee enough optimization accuracy based on the correlation between the response of the amplifier in the signature testing and the actual characteristics of the amplifier. Even though the optimization technique proposed in [61] does not require complex circuits, [61] does not show a completely integrated prototype chip and proves the validity of the technique by only using discrete components, measurement equipment, and an external computer. [62] minimizes spurious tones of a phase-locked loop (PLL) by optimizing a correction signal generator that compensates the control voltage fluctuation of a voltage-controlled oscillator. Since the optimization technique proposed in [62] needs to measure the control voltage fluctuation directly, a high-resolution high-speed analog-to-digital converter (ADC) is required for the technique, leading to a large area overhead. [62] uses an external ADC and a computer for the proposed optimization. [63] optimizes a phase rotator to minimize gain and phase errors caused by the rotator. [63] uses a vector network analyzer and an external computer to evaluate the errors and to operate an optimization algorithm. [64] proposes an optimization technique for an 18th-order G_m-C filter. Even though [64] claims that the proposed technique can minimize the gain and the group delay errors of the filter even when the filter consumes small area and power, [64] does not discuss about a stimulation signal generation and a response capturing mechanism. The prototype chip shown in [64] also does not include those two parts. In summary, [61–64] do not show a completely-integrated in situ analog circuit optimization technique. In addition, the articles do not discuss required accuracy for each building block of the proposed techniques. To the best of the author's knowledge, the only work that presents a fully-integrated in situ analog circuit optimization system is [4]. We discuss about the platform proposed in [4] in Chapter 5.

5. A BUILT-IN SELF-TEST AND IN SITU ANALOG CIRCUIT OPTIMIZATION PLATFORM[*]

5.1 Introduction

[4] proposes a built-in self-test and in situ analog circuit optimization platform. The platform directly measures excitation and response signals for a circuit under optimization (CUO). Based on the results of the measurements, a fully-digital optimization engine extended from [65] automatically finds an optimal control code for the CUO to fulfill multiple arbitrary weighted characteristic goals simultaneously. Therefore, the CUO can have and maintain well-balanced optimal characteristics even in severe PVT variations and device aging.

Because most circuit blocks except the CUO will be powered-off after the optimization process completes, the power consumption overhead is not a critical issue for this platform. Also, reusing mixed-signal circuits and digital computation blocks in many system-on-chip products, such as a frequency synthesizer, a low-speed analog-to-digital converter (ADC), and arithmetic logic units (ALUs), can mitigate the area overhead of this platform. Even if those blocks are not available for the overhead reduction, the platform can be justified by a higher yield and lower power optimal characteristics of the CUO.

The rest of this chapter is organized as follows. In Section 5.2, the proposed platform architecture is introduced. Section 5.3 describes the role and the structure of a cost function. The reason why we need an optimization engine and the algorithm that supports it are discussed in Section 5.4. In Section 5.5, the required accuracy of each building block in the platform is analyzed. Section 5.6 presents the Monte-Carlo simulation results of the platform and the measurement results of an integrated circuit (IC) prototype. Finally, conclusions are made in Section 5.7.
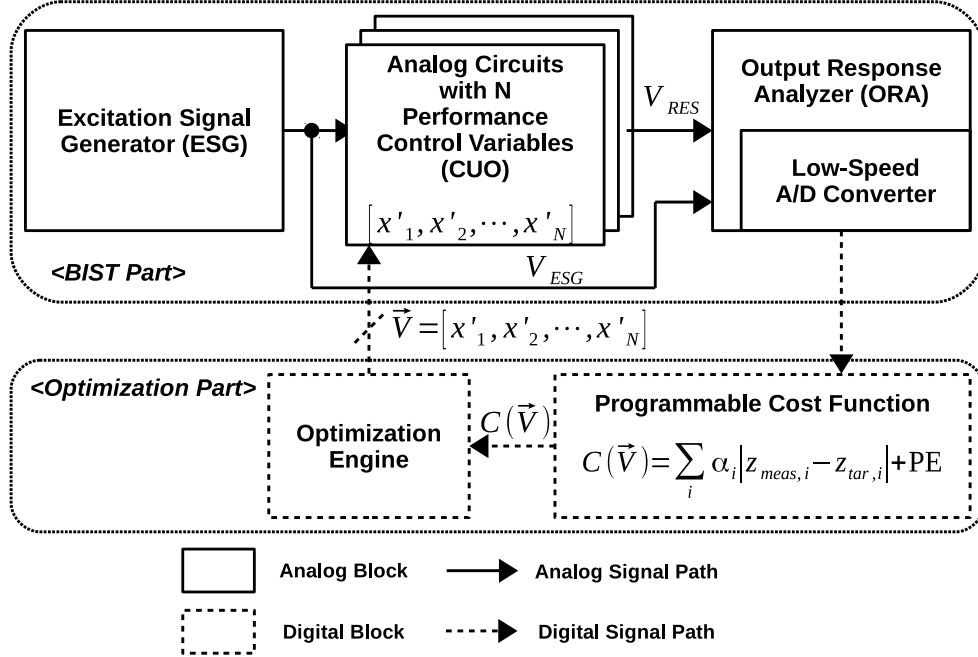
Figure 5.1: Conceptual architecture of the proposed platform. ©2018 IEEE.

5.2 The Proposed Platform Architecture

5.2.1 Optimization With BIST

The full concept of the proposed platform is illustrated in Fig. 5.1. The complete platform consists of an analog BIST part and a digital optimization part. Characteristics of the CUO in the analog BIST part can be changed by the N-dimensional control vector $\vec{V} = [x'_1, x'_2, \cdots, x'_N]$, which is a collection of tuning variables (knobs), such as widths of transistors, resistances, capacitances, and bias currents. These variables are modified by implementing arrays of transistors, resistors, and capacitors with digitally-controlled switches [65]. Once the vector is given as a digital code, characteristics of the CUO can be evaluated by stimulating it and analyzing its responses. In this platform, two types of responses can be measured and analyzed. One is a frequency-domain response and the other is a time-domain response. These two responses will be discussed in detail in Section 5.2.2 and 5.2.3. During the evaluation of i-th response, $z_{meas,i}$ is calculated. After all evaluations are complete, the overall performance of the CUO is quantified by computing a cost

59

function value of the control vector $\overrightarrow{V}$. The cost function is defined as a summation of differences between the extracted characteristic $z_{meas,i}$ and the target characteristic $z_{tar,i}$ for all i. Before the optimization, the cost function value can be high due to severe PVT variations and device aging. The optimization engine finds the optimal control vector $\overrightarrow{V_{opt}}$, which makes the cost function a minimum, by changing $\overrightarrow{V}$ via an optimization algorithm.

5.2.2 Frequency-Domain Characterization of a CUO

To characterize a CUO in the frequency domain, the excitation signal generator (ESG) in Fig. 5.1 utilizes a square wave and generates a sinusoidal signal V_{ESG} by suppressing harmonic components of the square wave [66]. The CUO is stimulated by the sinusoidal signal and generates a sinusoidal response V_{RES}. The ORA samples the input and the output signals of the CUO (V_{ESG}, V_{RES}) with the same frequency ω_1. Since the frequency of the sampled signals are identical to that of the sampling clock, the input of ORA is down-converted to DC, and the DC output is digitized using a low-speed ADC. By changing the phase of the sampling clock, the two-step in-phase and quadrature (I/Q) sampling [67] can be accomplished without an additional sampler as depicted in Fig. 5.2(a). If the input of ORA can be expressed as a sinusoidal signal,

$$y(t) = A\cos\left(\omega_1 t - \theta\right).$$ (5.1)

The sampled I/Q values can be represented as

$$y_I[m] = y(t) \cdot \delta(t - mT) = A\cos\left(\theta\right),$$

$$y_Q[m+1] = y(t) \cdot \delta(t - (m+1)T - T/4) = A\sin\left(\theta\right).$$ (5.2)

Here, $T = 2\pi/\omega_1$, and $\delta(t)$ indicates a Dirac delta function. The magnitude and the phase responses of the CUO can be extracted by comparing the magnitudes and the phases of V_{ESG} and V_{RES}, which are calculated from the sampled values.

After the evaluation of the magnitude and the phase responses of the CUO at ω_1, a new si-
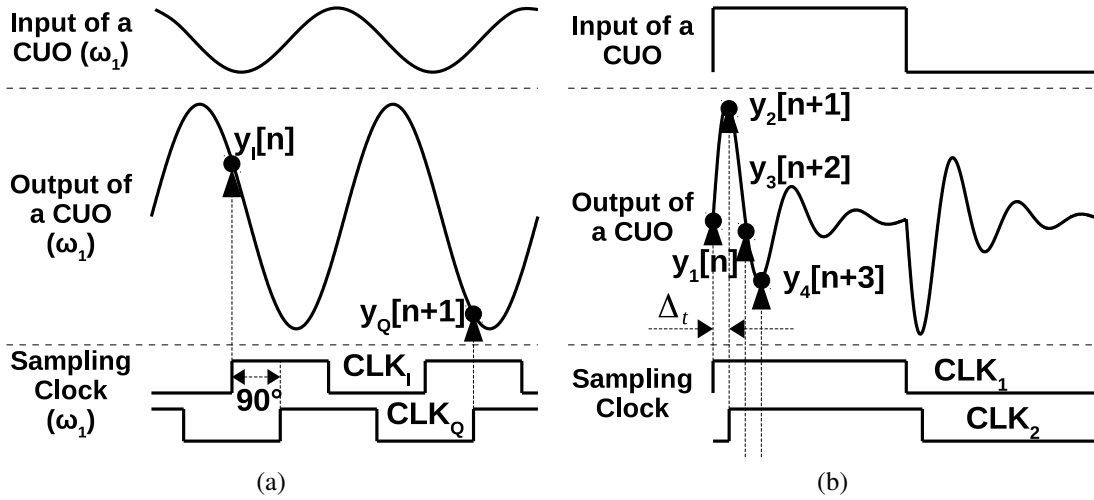
Figure 5.2: Frequency- and time-domain characterizations. (a) I/Q sampling for the frequency-domain characterization. (b) Multi-phase sub-sampling for the time-domain characterization. ©2018 IEEE.

nusoidal signal at frequency ω_2 is generated, and the entire calculation process is repeated. By iterating this procedure multiple times, the transfer function of the CUO can be estimated. A cost function quantifies the difference between the estimated transfer function and the target transfer function as a metric that can be utilized for CUO optimization.

5.2.3 Time-Domain Characterization of a CUO

For a highly optimized CUO, direct measurements of time-domain characteristics are mandatory because indirect prediction of the characteristics from the frequency-domain measurements is not accurate. It is true that time-domain characteristics can be derived from the estimated transfer function when a CUO is an ideal 1st- or 2nd-order system [68]. However, many practical circuit systems have higher orders than second order since many non-ideal factors, such as parasitic components and finite GBW of op-amps, can be prominent in a real system. Therefore, time-domain characteristics of a CUO cannot simply be deduced from its frequency characteristics; they should be optimized together with other circuit characteristics.

To evaluate time-domain characteristics of a CUO, a step input signal is applied to the CUO.
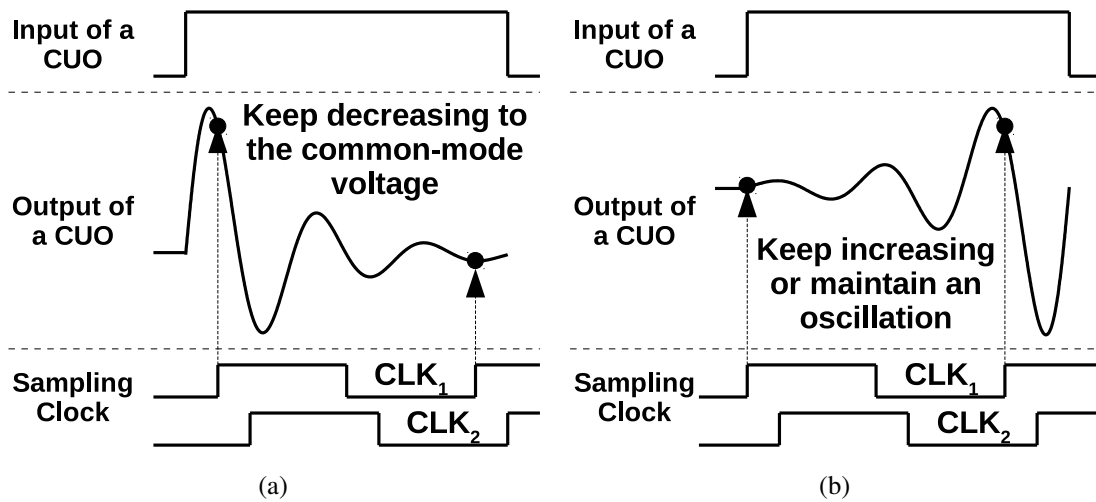
Figure 5.3: Stability test. (a) Stable case. (b) Unstable or marginally stable case. ©2018 IEEE.

The step input can be emulated by utilizing a square wave in the ESG. Because many circuit systems have complex poles, the CUO might show an under-damped response that has a frequency close to ω_O as depicted in Fig. 5.2(b). The most straightforward way to extract the peak value and the settling time of the response is sampling the response with a frequency much higher than ω_O. However, it requires a fast sampler, and that can be a burden for the entire platform. To mitigate this issue, a multi-phase sub-sampling can be deployed [59]. Instead of sampling multiple times within a single-clock cycle, an integer number of clock-cycle delays can be introduced between two consecutive sampling actions. For convenience, sub-sampled data are displayed in a single-clock cycle in Fig. 5.2(b). This sub-sampling can be effective only when the input and the output of the CUO are periodic signals. Even though the effective operating speed of the sampler can be relaxed by utilizing this technique, the fine timing resolution (Δ_t) is still required. Adequate values for Δ_t will be explored in Section 5.5.

In addition, the time-domain characterization is also useful for the stability test of a CUO. As shown in Fig. 5.3, the step response of a CUO will diminish in time if the CUO is stable. However, the response will keep growing or maintain an oscillation if it is unstable or marginally stable. Therefore, the oscillation can be detected by sampling the step response during multiple-

clock cycles. If the sampled values converge to a DC voltage, the CUO can be considered a stable system. Otherwise, the CUO is in an unstable state or a marginally-stable state. This test result can be incorporated in a cost function as a penalty term.

5.3 Cost Function

Before the optimization engine finds an optimal control vector $\overrightarrow{V_{opt}}$, on-chip quantitative evaluation of each control vector $\overrightarrow{V}$ is required. For that purpose, a cost (or error) function should be defined. It measures the difference between the target and the measured characteristics of a CUO when the control vector $\overrightarrow{V}$ is assigned to the CUO as below.

$$C(\overrightarrow{V}) = \sum_{i=1}^{M} \alpha_i \left| z_{meas,i}(\overrightarrow{V}) - z_{tar,i} \right| + \mathrm{PE}(\overrightarrow{V}). \tag{5.3}$$

α_i is a constant for each i, PE is a penalty function, and M is the total number of measured characteristics of a CUO. When the measured characteristics of a CUO at $\overrightarrow{V}$ are far from the target, the cost function value $C(\overrightarrow{V})$ is high, and the optimization engine will try a different $\overrightarrow{V}$ to minimize $C(\overrightarrow{V})$. On the other hands, when $C(\overrightarrow{V})$ is smaller than a certain cost criterion, or when the maximum number of algorithmic iteration has been reached, the optimization engine stop finding a better $\overrightarrow{V}$, and $\overrightarrow{V}$ will be fixed.

Ideally, the target design point should be located at the edge of the acceptable performance region to minimize power and area consumption while satisfying all specifications. However, in this case, the actual operating point of a CUO after optimization can be outside the acceptable performance region, since there is a possibility that the optimization engine fails to find the design point that has exact zero cost due to the limited number of iterations. Therefore, by assigning a small margin (θ_1) to the target design point (B) as shown in Fig. 5.4, we can achieve a good balance between the optimization quality and the probability of finding an actual operating point inside the acceptable performance region after optimization (reliability). The quantitative relations between the probability, the margins, and the algorithm stopping criteria will be presented in Section 5.4.

In (5.3), $z_{meas,i}(\overrightarrow{V})$ can represent various values. In the frequency-domain characterization,
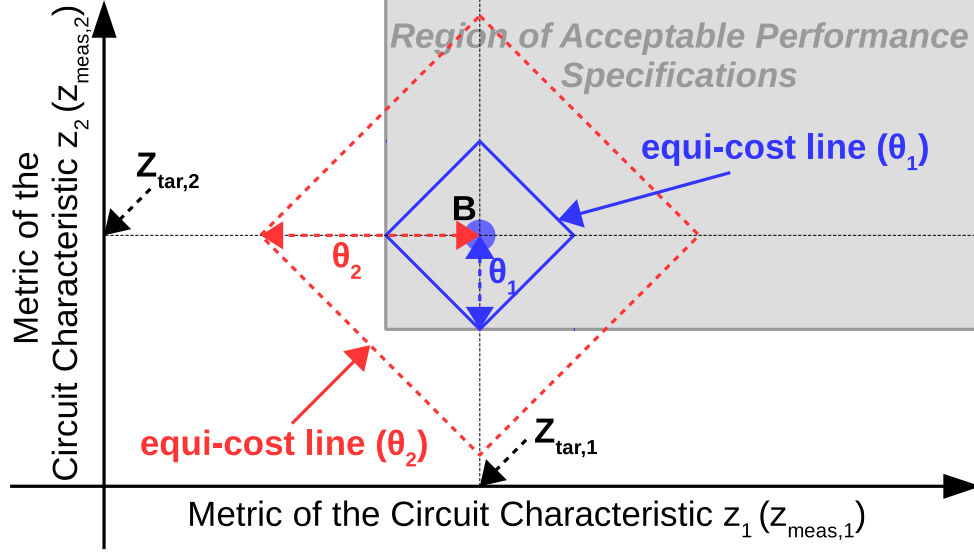
Figure 5.4: Magnified view around the design point B in Fig. 4.1, and equi-cost lines when $M = 2$, $\alpha_1 = \alpha_2 = 1$, and PE $= 0$ in (5.3). ©2018 IEEE.

$z_{meas,i}(\overrightarrow{V})$ can be a magnitude or a phase response of a CUO at a certain frequency ω. In the time-domain characterization, $z_{meas,i}(\overrightarrow{V})$ can be a time-domain characteristic such as a peak value or a settling time of a CUO. By varying α_i, each term of the cost function can have a different weight. For instance, if $\alpha_1 = 1$ and $\alpha_i = 0 \, \forall i \neq 1$, then the cost function only evaluates $z_{meas,1}(\overrightarrow{V})$ and ignores all the other $z_{meas,i}(\overrightarrow{V})$. In this specific case, the optimization engine will find the $\overrightarrow{V}$ that makes $\left| z_{meas,1}(\overrightarrow{V}) - z_{tar,1} \right|$ a minimum. In this way, relative importance of each specification can be given by the user of this platform, and a right "balance" among various circuit characteristics can be achieved.

Another term PE$(\overrightarrow{V})$ represents a penalty function. In a constrained optimization problem, its constraint can be added to the cost function as a penalty term. Then, this transformed problem can be solved in the same way as an unconstrained optimization [69]. Through design-time simulations, we can carefully choose the weighting factor for the penalty term to satisfy the corresponding constraint while the original objective is not significantly sacrificed. For example, in Fig. 5.4, z_1 and z_2 characteristics of a CUO can be optimized only when the CUO is stable. Therefore, the characteristics should be constrained within the region, where the stable operation of the

64

CUO is guaranteed. The confinement can be achieved by inserting the following penalty to the cost function.

$$\text{PE}_{stb}(\overrightarrow{V}) = \begin{cases} \infty & \text{if unstable or marginally stable,} \\ 0 & \text{otherwise.} \end{cases} \tag{5.4}$$

The optimization engine automatically rules out the control vector $\overrightarrow{V}$ that makes the CUO unstable or marginally stable because the cost is too high in that case.

In addition, the power consumption of a CUO can also be included in a cost function as a penalty. Because the minimum power consumption is always a desired characteristic, it does not need to be measured when bias currents can be controlled monotonically even in PVT variations and device aging. This monotonicity can be easily obtained by utilizing thermometer codes for the control of resistors. The penalty function for minimum bias current can be represented as

$$\text{PE}_{pwr}(\overrightarrow{V}) = x'_{bias}. \tag{5.5}$$

Here, x'_{bias} is an element of $\overrightarrow{V}$. When the actual bias current of a CUO is proportional to the control code value x'_{bias}, the optimization engine will try to minimize the control code x'_{bias} itself, and the bias current will be the smallest value while satisfying all the other constraints and requirements.

Overall, the entire penalty term can be expressed as a function of stability and power consumption.

$$\text{PE}(\overrightarrow{V}) = \text{PE}_{stb}(\overrightarrow{V}) + \beta \cdot \text{PE}_{pwr}(\overrightarrow{V}). \tag{5.6}$$

In (5.3) and (5.6), appropriate weight (α_i, β) should be chosen because each term can have a different unit and emphasis according to CUO specifications. This can be done by checking optimization results in simulations and adjusting the weight.

5.4 Optimization Engine

Based on the costs of various control vectors, the optimization engine tries to find the optimal control vector $\overrightarrow{V_{opt}}$, which makes $C_{opt}(= C(\overrightarrow{V_{opt}}))$ a minimum. The most intuitive algorithm to find

(a)

(b)
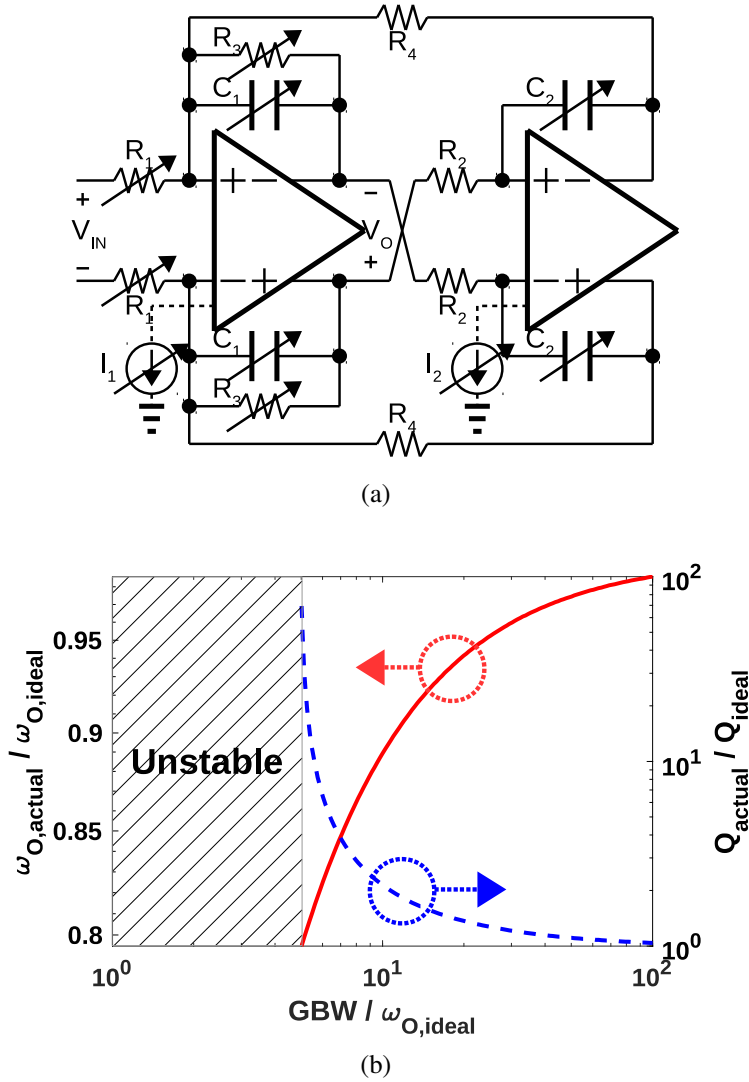
Figure 5.5: Tow-Thomas biquad. (a) Schematic. (b) ω_O & Q characteristics. ©2018 IEEE.

$\overrightarrow{V_{opt}}$ is enumerating all possible combinations of the control knob settings. However, this approach is not realistic in that the size of the space of $\overrightarrow{V}$ increases exponentially as the number of bits of the control knobs increases.

To overcome this issue, one common approach is utilizing an orthogonal tuning/calibration [70]. In the orthogonal tuning, each circuit characteristic can be tuned one by one, and each tuning action does not affect other circuit characteristics. Since each characteristic is tuned independently, a set of specifications can be divided and conquered separately. Thus, this approach can find $\overrightarrow{V_{opt}}$

much faster because the size of the total search space increases linearly with the number of control knobs.

Unfortunately, orthogonal tuning is not a possible option when many specifications should be dealt simultaneously because circuit characteristics have trade-off relationships with each other by nature. This can be well illustrated by the following example. In Fig. 5.5(a), if each op-amp is modeled as a two-pole system, the relation between Q_{actual}, $\omega_{O,actual}$, and GBW can be shown as in Fig. 5.5(b). In this figure, Q_{ideal} and $\omega_{O,ideal}$ mean Q and ω_O of the filter when the GBW is infinite, whereas Q_{actual} and $\omega_{O,actual}$ are Q and ω_O when the GBW is a given value. As we can see, Q_{actual} and $\omega_{O,actual}$ are not orthogonal to the GBW. When the GBW is too small, $\beta \cdot \mathrm{PE}_{pwr}(\overrightarrow{V})$ in (5.6) are negligible because the GBW is proportional to the power consumption of the op-amps, whereas $\alpha_i \left| z_{meas,i}(\overrightarrow{V}) - z_{tar,i} \right|$ in (5.3) is large due to the limited tuning ranges. On the contrary, when the GBW is larger than needed, $\beta \cdot \mathrm{PE}_{pwr}(\overrightarrow{V})$ is too high. After all, $\overrightarrow{V_{opt}}$ is located between those two extreme cases. Thus, all settings for the GBW should be checked to find $\overrightarrow{V_{opt}}$. When the orthogonal tunings are applied to the CUO at each GBW setting, the size of the total search space will be 3×2^{10}, provided that the control knobs for Q_{actual}, $\omega_{O,actual}$, and gain at $\omega_{O,actual}$ are orthogonal to each other, and all control knobs have 5 bits per each knob, including the knob for the GBW. If the GBW of two op-amps are tuned separately, the size will be 3×2^{15}. The important point here is that the size of the search space increases exponentially when more non-orthogonal circuit characteristics should be tuned together. Consequently, the orthogonal tuning scheme cannot be a universal solution due to its limitation.

Instead of trying to reduce the size of search space, another approach to discover $\overrightarrow{V_{opt}}$ efficiently is utilizing well-established optimization algorithms. Regardless of dependencies among control knobs, the methodologies previously introduced are based on enumeration, which are close to brute-force methods. Optimization algorithms can be a powerful tool as they reduce the required time for searching when the size of search space is given.

In general, analog circuit optimization is a non-convex problem if there are no special approximations [71]. Even though a complete theory discovering an "exact" global minimum within a

reasonable time has not been found yet for non-convex optimization, there are many meta-heuristic algorithms that can converge to a sub-optimal point [72]. Since those algorithms are on the basis of heuristics, discovering $\overrightarrow{V_{opt}}$ cannot be guaranteed. However, if the algorithms are designed properly, the final results of the algorithms can be very close to the optimal one. There are two types of meta-heuristic algorithms. The first type is a single-solution approach, which stores only one previous candidate and modifies it to get a new candidate. Pattern search (PS), sensitivity search (SS), and simulated annealing (SA) can be included in this category. The second type is a population-based approach. Algorithms in this category maintain a number of previous candidates and exploit previous search experience to guide the search process for a new candidate better. Genetic algorithm (GA) and Nelder-Mead method (NM) are examples of this type.

The SA-SS hybrid algorithm is utilized in this paper even though any kind of meta-heuristic algorithm can be used for the platform proposed in this paper. First, because population-based algorithms require larger memory than single-solution algorithms, they are ruled out to reduce hardware complexity and area overhead. Second, among single-solution algorithms, SA is good at exploring search space and approaching to the points close to a global optimum, but poor at converging to the optimum. On the other hand, SS searches well around an initial starting point, but can be trapped in a local minimum. Therefore, by merging the two algorithms, a global optimum can be found quickly, with local minima avoided [65].

A pseudo code for the hybrid algorithm is shown in Algorithm 1. It consists of two parts. The first part is an SA phase, which is described in Steps 3-15. The second part is an SS phase, which is shown in Steps 17-24. In the SA phase, a random control vector $\overrightarrow{V_{new}}$ is newly generated at each iteration (Step 4). If the cost of the new control vector is smaller than the previous optimal cost, the previous vector and the cost of it are updated to the new one (Steps 5-8). Even if the new cost is larger than the previous cost, the update is executed if the condition shown in Step 10 is true (Steps 10-12). In Step 10, T is virtual temperature that is utilized for the algorithmic annealing process. C_Δ is the difference between the new cost and the previous optimal cost. $random(0, 1)$ is a randomly selected number between 0 and 1 at each iteration. When T is very large, $e^{-C_\Delta/T}$

Input : Initial virtual temperature T_{max}, cooling rate k, maximum number of SA iterations
MAX_{SA}, maximum number of SS iterations MAX_{SS}, and cost function threshold θ
Output: The optimal solution $\overrightarrow{V_{opt}}$ and the corresponding cost function value C_{opt}

  1 Initialize $T \leftarrow T_{max}$; $C_{opt} \leftarrow \infty$; $\overrightarrow{V_{opt}} \leftarrow INIT$;

  2 Set global counter $i \leftarrow 0$;

  3 **while** $i < MAX_{SA}$ & $C_{opt} > \theta$ **do**

  4 $\overrightarrow{V_{new}} \leftarrow RANDOM$;

  5 $C_{new} \leftarrow C(\overrightarrow{V_{new}})$;

  6 $C_{\Delta} \leftarrow C_{new} - C_{opt}$;

  7   **if** $C_{\Delta} < 0$ **then**

  8 $(\overrightarrow{V_{opt}}, C_{opt}) \leftarrow (\overrightarrow{V_{new}}, C(\overrightarrow{V_{new}}))$;

  9   **else**

 10   **if** $exp(-C_{\Delta}/T) > random(0,1)$ **then**

 11 $(\overrightarrow{V_{opt}}, C_{opt}) \leftarrow (\overrightarrow{V_{new}}, C(\overrightarrow{V_{new}}))$;

 12   **end**

 13   **end**

 14 $T \leftarrow kT$; $i++$;

 15 **end**

 16 $C_{tmp} \leftarrow \infty$;

 17 **for** $i = 0$; $i < MAX_{SS}$ & $C_{opt} < C_{tmp}$; $i++$ **do**

 18 $C_{tmp} = C_{opt}$;

 19   **for** *each neighbor $\overrightarrow{V_j}$ of $\overrightarrow{V_{opt}}$* **do**

 20   **if** $C(\overrightarrow{V_j}) < C_{opt}$ **then**

 21 $(\overrightarrow{V_{opt}}, C_{opt}) \leftarrow (\overrightarrow{V_j}, C(\overrightarrow{V_j}))$;

 22   **end**

 23   **end**

 24 **end**

 25 return $(\overrightarrow{V_{opt}}, C_{opt})$;

Algorithm 1: SA-SS hybrid algorithm. ©2018 IEEE.

will be close to 1, and a majority of new control vectors are going to replace previous vectors even
if the replacement can increase the cost temporarily. By repeating this process, "hill climbing"
can occur, and local minima can be avoided. However, the hill-climbing activity becomes unusual
as T decreases ($0 < k < 1$). Therefore, the final solution can converge to a global minimum.
After MAX_{SA} iterations, or after the point that has a cost smaller than θ is found, the SA phase
is closed, and the optimization engine starts the SS algorithm (Step 16). By adding/subtracting 1-
LSB to each control knob, all neighbors ($\overrightarrow{V_j}$) of the current optimal vector can be evaluated (Steps
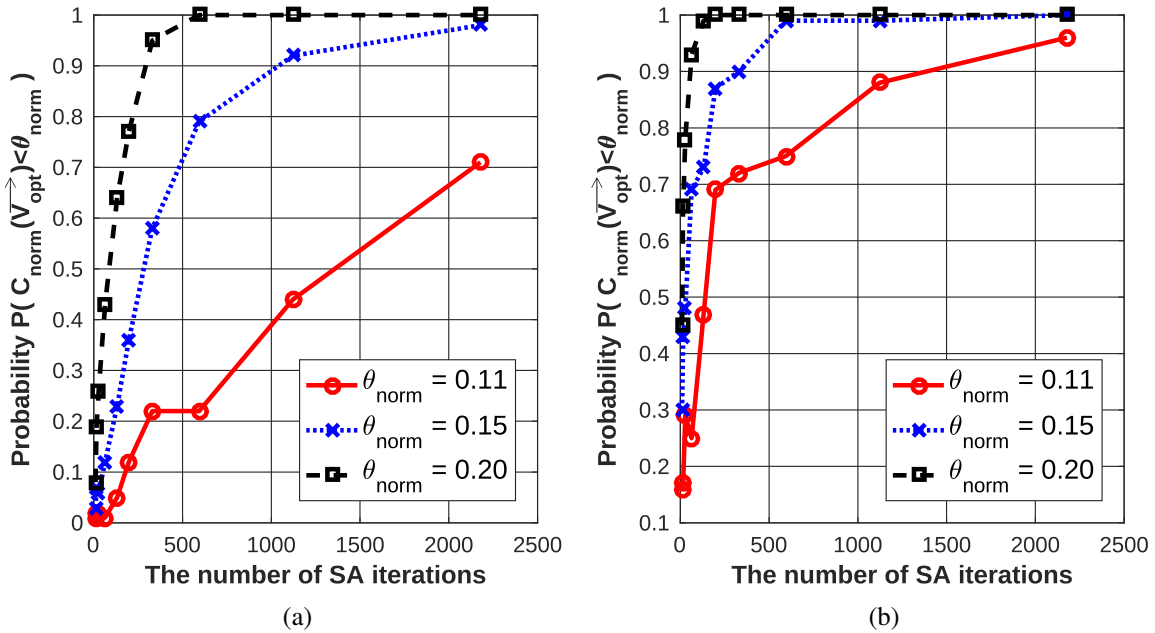
Figure 5.6: Relation between the number of SA/SS iterations, the normalized cost criterion, and the probability of having a cost smaller than the criterion after the number of iterations. (a) $MAX_{SS} = 0$. (b) $MAX_{SS} = 3$. ©2018 IEEE.

19-20). The engine will take the $\overrightarrow{V_j}$ that generates the lowest cost (Step 21). This process will be repeated until there is no room for improvement, or until the maximum iteration limit (MAX_{SS}) has been reached.

To make the SA-SS algorithm more reliable and effective, the stopping criteria and the parameter k should be chosen and verified by testing sample chips or by running repetitive simulations. Fig. 5.6 shows the results of such simulations. It reveals the relation between the number of SA/SS iterations and the probability of having a cost smaller than a certain criterion after the number of iterations. In the simulations, all costs (C_{norm}) and cost criteria (θ_{norm}) are normalized by the cost of a fixed starting point. As the figure indicates, if we have large θ_{norm}, a relatively small number of SA/SS iterations are required to obtain $C_{norm}(\overrightarrow{V_{opt}})$ smaller than θ_{norm} with high probability (0.8~0.9). However, large θ_{norm} means huge back-off (margin) from the edge of an acceptable performance region as shown in Fig. 5.4. Therefore, the criteria (MAX_{SA}, MAX_{SS}, θ) should be

chosen according to the user's need; if the user wants high-quality optimization, θ should be small but MAX_{SA} and MAX_{SS} will be large.

One possible concern regarding this platform is that the optimization algorithm can converge to a bad solution that is far from an acceptable performance region because of the heuristic nature of the algorithm. Even though it is true that we can lower the probability of having the bad solution by allowing enough iterations, we cannot avoid the situation 100% especially when the CUO should be optimized periodically to track PVT variations and device aging.

A feasible solution is supporting multiple modes of operation. For example, a CUO can support a dual-mode operation: a conservative mode and an optimization mode. The dual modes have two different design points. For example, the design center in Fig. 4.1 for the conservative mode and B for the optimization mode. When the CUO is powered on, it starts from the conservative mode; thus, the operating point of it is still inside an acceptable performance region even if there are large PVT variations and device aging. In some applications, the CUO does not need to be always on [73]. When the CUO is in an idle state, the mode can be changed to the optimization mode. If the optimization process successfully finds a good operating point inside an acceptable performance region before the end of the idle state, the CUO will have the updated operating point. Otherwise, the CUO will maintain the previous conservative operating point and wait until the next idle state comes. If the CUO should always be in an on state for other applications, we can exploit two same CUOs: one is in the optimization mode while the other is in use. By following this approach, we can guarantee that the actual operating point of a CUO is always inside an acceptable performance region after the optimization process is complete.

5.5 Analysis of Required Accuracies for Platform Building Blocks

5.5.1 Definitions

5.5.1.1 Control vector

A control vector can be defined as a collection of tuning variables (knobs).

$$\overrightarrow{V} = [x'_1, x'_2, \cdots, x'_N], \; x'_k = (x_k - x_{k,min})/x_{k,LSB}. \tag{5.7}$$

x'_k is a normalized non-negative integer value.

5.5.1.2 Euclidean distance

A Euclidean distance between two control vectors, $\overrightarrow{V}$ and $\overrightarrow{V_{cen}}$, can be defined as below.

$$d(\overrightarrow{V}, \overrightarrow{V_{cen}}) = \sqrt{\sum_{i=1}^{N} \left(x'_i - x'_{cen,i} \right)^2}. \tag{5.8}$$

N indicates the total number of tuning variables.

5.5.1.3 Percent root-mean-square error (%RMSE)

In order to quantify how close the actual characteristics of a CUO are to the specifications given by the user of this platform, a %RMSE can be defined as

$$\%\text{RMSE}(\overrightarrow{V}) = 100 \times \sqrt{\frac{1}{L} \sum_{i=1}^{L} \left(\frac{z_{act,i}(\overrightarrow{V}) - z_{tar,i}}{z_{tar,i}} \right)^2}. \tag{5.9}$$

$z_{act,i}$ means i-th actual characteristic of a CUO at $\overrightarrow{V}$, and $z_{tar,i}$ indicates i-th target characteristic of a CUO set by the user. It should be mentioned here that $z_{act,i}$ is different from $z_{meas,i}$ in (5.3). Due to the non-idealities of an ESG, an ORA, and digital computation blocks, the measured characteristics of a CUO on a chip will have some errors compared to the actual characteristics of the CUO. If we consider that all circuits except the CUO will be powered-off after the optimization process is completed, the measured characteristics themselves are not important in the perspective

of the user. Instead, the important thing is whether the actual characteristics of the CUO at $\overrightarrow{V}$ are close enough to specifications or not. This can be revealed by the %RMSE, and it can be utilized as an indicator of optimization accuracy.

In the frequency-domain characterization, $z_{act,i}$ and $z_{tar,i}$ will be an actual gain and a required gain, respectively, at f_i. According to the frequency range of interest, the total number of frequency points (L) should be big enough to cover all the range. Also, the frequency step (f_{step}) between two adjacent frequency points should be small enough to accurately measure the difference between a target transfer function and an actual transfer function. In this paper, it is assumed that $L = 100$ and the frequency points are spread evenly in a logarithmic scale from $\omega_O/10$ to $10 \cdot \omega_O$ when the CUO has a bandpass frequency response. Regardless of the types and orders of a CUO, L and f_{step} can be set in a similar manner.

In the time-domain characterization, $z_{act,i}$ and $z_{tar,i}$ can be a settling time, peak value, or peak time. Because we are not going to optimize the entire shape of a step response, the %RMSE does not need to be defined over a finite set of time samples $\{t_i\}$ different from the %RMSE for the frequency-domain characterization.

5.5.2 Design of the Cost Function

There are two major differences between the cost function and the %RMSE. First, unlike the %RMSE, the cost function is based on measured characteristics. If there are significant errors in the measurements, the %RMSE and the cost will show a significant deviation. Second, some realistic factors of on-chip in-situ optimization should be considered for the cost function. For instance, in the frequency-domain characterization, the total number of frequency points (M) and f_{step} should be reasonable values. If M is too big, the time that is required to complete the optimization process will be unrealistically long. Therefore, the cost function should be designed properly.

The cost function for the frequency-domain characterization can be:

$$C(\overrightarrow{V}) = \sum_{i=1}^{M} \alpha_i \left| G_{meas}(\overrightarrow{V}, f_i) - G_{tar}(f_i) \right| + \text{PE}(\overrightarrow{V}). \tag{5.10}$$

73
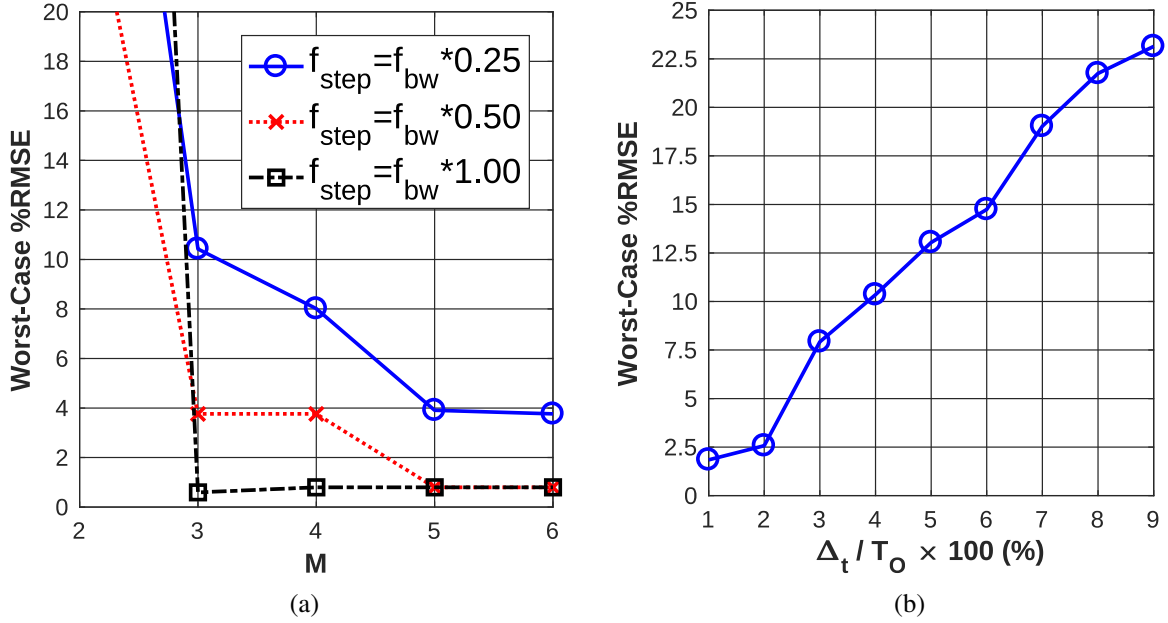
Figure 5.7: Relation between the %RMSE and the design parameters of the cost functions. (a) %RMSE and $\{M, f_{step}\}$. (b) %RMSE and Δ_t. ©2018 IEEE.

G_{meas} means the measured gain at $\overrightarrow{V}$ and f_i; G_{tar} indicates the target gain at f_i, which is given by the user. To choose right values for M and f_{step}, the relationship between those parameters and the %RMSE should be evaluated. Ideally, the optimization engine should always find the optimal control vector $\overrightarrow{V_{opt}}$ among $\overrightarrow{V}$. However, in reality, it is not always possible because of several non-idealities, which will be clarified later. If we define a sub-optimal control vector $\overrightarrow{V_{sopt}}$ as a vector that makes the normalized cost $(C(\overrightarrow{V_{sopt}})/C(\overrightarrow{V_{opt}}))$ smaller than a certain criterion (C_{crit}), there will be a number of $\overrightarrow{V_{sopt}}$ that satisfy the condition, and it can be assumed that the engine can discover one of the several $\overrightarrow{V_{sopt}}$ regardless of the non-idealities. In this case, the worst %RMSE($\overrightarrow{V_{sopt}}$) can be extracted among the various $\overrightarrow{V_{sopt}}$ for each combination of M and f_{step}. To find all $\overrightarrow{V_{sopt}}$, $C(\overrightarrow{V})$ should be enumerated for all $\overrightarrow{V}$ at each M and f_{step} setting.

For the time-domain characterization, the cost function can be:

$$C(\overrightarrow{V}) = \sum_{i=1}^{3} \alpha_i \left| TC_{meas,i} - TC_{tar,i} \right| + PE.(\overrightarrow{V}) \qquad (5.11)$$

74

$\text{TC}_{meas,i}$ and $\text{TC}_{tar,i}$ indicate the i-th element of TC_{meas} and TC_{tar}, respectively. The two TC vectors can be defined as below.

$$\text{TC}_{meas} =$$
$$\left[\text{ST}_{meas}(\vec{V}, \Delta_t), \text{PV}_{meas}(\vec{V}, \Delta_t), \text{PT}_{meas}(\vec{V}, \Delta_t)\right],$$
$$\text{TC}_{tar} = \left[\text{ST}_{tar}, \text{PV}_{tar}, \text{PT}_{tar}\right]. \tag{5.12}$$

In (5.12), ST, PV, and PT mean a settling time, peak value, and peak time individually. An appropriate Δ_t can also be chosen by following the same simulation procedure discussed before.

Fig. 5.7 shows the simulation results for the cost function design. In these simulations, it is assumed that the CUO has a 2nd-order bandpass frequency response and $C_{crit} = 1.5$. As Fig. 5.7(a) shows, if M and f_{step} are too small, the %RMSE can be large. To achieve better than 1% accuracy in the frequency-domain characterization, $\{M, f_{step}\}$ should be larger than or equal to $\{3, f_{bw}\}$, where $f_{bw} = \omega_O/(2\pi Q)$, and the center of the frequency points is located at ω_O. The relation between the %RMSE and Δ_t/T_O for the CUO is revealed in Fig. 5.7(b), where $T_O = 2\pi/\omega_O$. To get accuracy close to 1%, Δ_t should be around 1% of T_O. If the time-domain characteristics are not the primary concern, this requirement can be relaxed.

Even though we assumed that the CUO had a 2nd-order bandpass frequency response, the analyses proposed here and the following subsections are not limited to the specific CUO. In other words, the same analyses can be applied to any orders/types of CUOs to get valuable design information.

5.5.3 Analysis of the Effect of Distortions

Distortions in the analog blocks can distort a cost function. As shown in Fig. 5.8, the measured gain (G_{meas}) can be represented as the summation of the actual gain (G_{act}) and the error (E) that stems from the distortions of the analog blocks. In the worst case scenario, E can be so large that G_{meas} matches the target gain (G_{tar}) even though G_{act} is quite different. In this extreme case, the cost will be close to zero according to (5.10) if there is no penalty term. However, the %RMSE
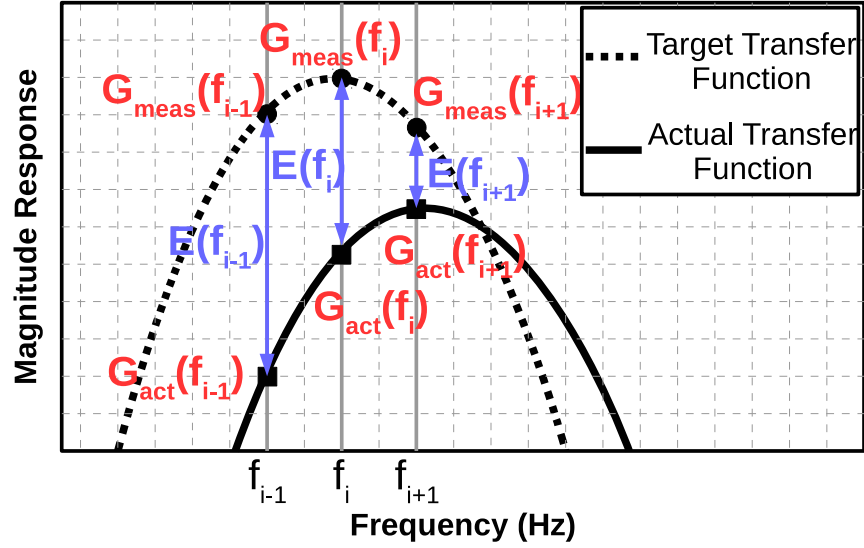
Figure 5.8: Effect of distortions in the frequency-domain characterization. ©2018 IEEE.

is not zero because the actual transfer function is different from the target. Unfortunately, the optimization engine operates based on the cost function, and the engine will find the $\overrightarrow{V}$ that makes $G_{meas} = G_{tar}$. Therefore, the final control vector after optimization ($\overrightarrow{V_{dis}}$) can have large %RMSE if errors originated from the distortions are big enough.

The error E can be obtained as follows. In Fig. 5.9, if the node B is connected to the ORA through the mux, the signal at node D can be represented as below when we assume that the phases of the three tones are the same.

$$y_D(t) \simeq B_1 \cos(\omega_i t) + B_2 \cos(2\omega_i t) + B_3 \cos(3\omega_i t). \tag{5.13}$$

Then the magnitude of $y(t)$ will be expressed:

$$\text{Mag}(y_D(t)) = \sqrt{y_I^2[m] + y_Q^2[m+1]} = B_1 + B_2 + B_3. \tag{5.14}$$

If $x_D(t)$ also has three tones that have A_1, A_2, and A_3 amplitudes, and if all phases of the three
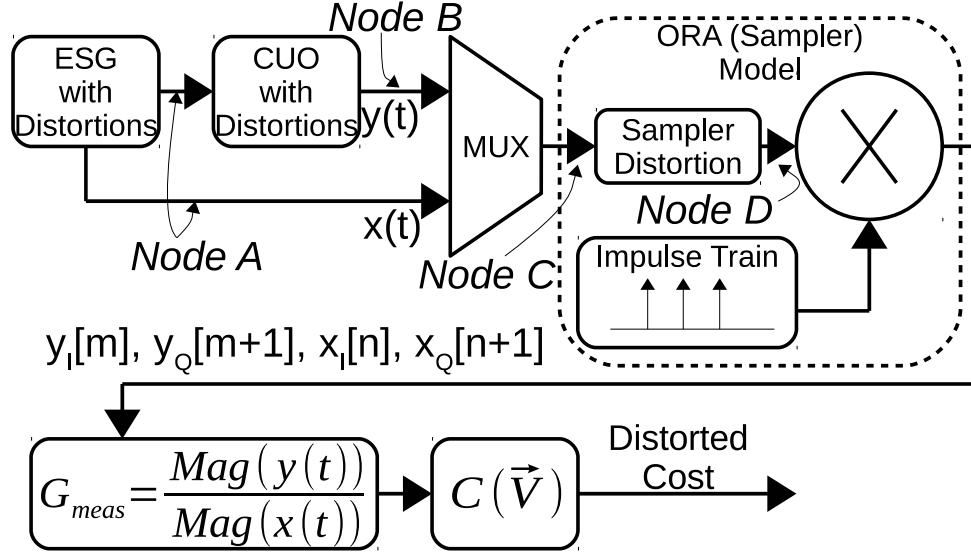
Figure 5.9: Block diagram for the distortion analysis. ©2018 IEEE.

tones are identical, G_{meas} can be derived:

$$G_{meas} = \frac{B_1 + B_2 + B_3}{A_1 + A_2 + A_3} = \frac{B_1}{A_1} \times \frac{1 + B_{disto}/B_1}{1 + A_{disto}/A_1}. \tag{5.15}$$

$A_{disto} = A_2 + A_3$ and $B_{disto} = B_2 + B_3$. To calculate A_{disto} and B_{disto}, the power of 2nd- and 3rd-order harmonic distortions (HD$_2$, HD$_3$) should be obtained at node A, B, and D. At node A, the power of harmonic distortions can be expressed as below [74].

$$\text{HDk}_{ESG} = \text{OIPk}_{ESG} - k(\text{OIPk}_{ESG} - P_{ESG}). \tag{5.16}$$

Here, $k = 2$ or 3, and P means the power of an output main tone. Also, OIP2 and OIP3 indicate an output intercept point for 2nd- or 3rd-order harmonic distortion individually. All terms in (5.16) are in the dBm scale. The three tones at node A are transferred to node B.

$$P_{CUO} = G_{act}(\vec{V}, f_i)\big|_{dB} + P_{ESG}.$$
$$\text{HDk}_{CUO,trans} = G_{act}(\vec{V}, kf_i)\big|_{dB} + \text{HDk}_{ESG}. \tag{5.17}$$
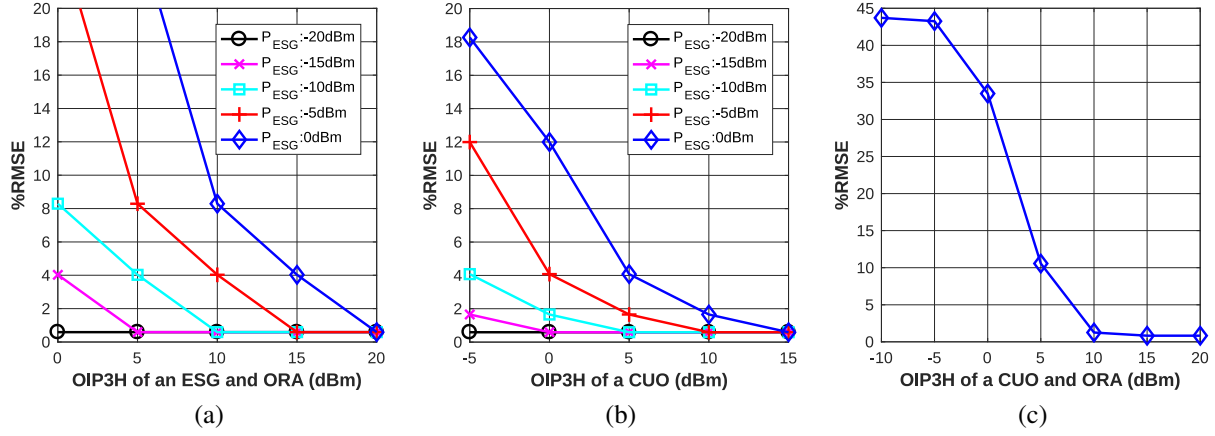
Figure 5.10: Simulation results showing the relation between the %RMSE and OIP3H of each block (OIP2H = 60 dBm for all blocks). (a) Simulation of the frequency-domain characterization. OIP3H$_{CUO}$ = 30 dBm. (b) Simulation of the frequency-domain characterization. OIP3H$_{ESG}$ = OIP3H$_{ORA}$ = P_{ESG}+20 dB. (c) Simulation of the time-domain characterization. P_{ESG} = 8.6 dBm (−3 dBFS for a 1.2 V supply voltage). ©2018 IEEE.

Because of the output main tone of the CUO, 2nd- and 3rd-order harmonics are newly generated.

$$\text{HDk}_{CUO,self} = \text{OIPk}_{CUO} - k(\text{OIPk}_{CUO} - P_{CUO}). \tag{5.18}$$

Approximately, HDk$_{CUO}$ can be expressed as the power summation of the two signals.

$$\text{HDk}_{CUO} \simeq 10 \log_{10}(10^{\text{HDk}_{CUO,trans}/10} + 10^{\text{HDk}_{CUO,self}/10}). \tag{5.19}$$

By taking the similar approach, HD2$_{ORA}$ and HD3$_{ORA}$ can be calculated for each mux setting. After that, the power of the two harmonics can be converted to A_{disto} or B_{disto}.

By enumerating $\overrightarrow{V}$ and calculating the distorted cost at each $\overrightarrow{V}$, $\overrightarrow{V_{dis}}$ can be found if OIP2H and OIP3H are given for each block. In this way, %RMSE($\overrightarrow{V_{dis}}$) and linearity specifications can be related. The relation is shown in Fig. 5.10. As indicated in Fig. 5.10(a), to maintain the %RMSE smaller than 1% for the frequency-domain characterization, OIP3H of the ESG and ORA should be 20 dB larger than P_{ESG}. This is equivalent to −40 dB total harmonic distortion (THD). When

the ESG and ORA have -40 dB THD, the CUO should have -30 dB THD from Fig. 5.10(b). In these simulations, OIP2H of all blocks are assumed very high because 2nd-order distortions are negligible if we use fully-differential circuits.

A similar analysis can be applied to the time-domain characterization. The only difference between the two analyses is that the input and the output of the CUO have many frequency components in the time-domain characterization. Because most power of the CUO output is concentrated on around a certain frequency depending on the frequency-domain characteristic of the CUO, we can add the power of the tones near the frequency and consider it the power of a main tone. Then 2nd- and 3rd-order harmonic tones can be obtained from the main-tone power when OIP2H and OIP3H specifications are given. The tones can be added to the original step response that does not include any non-idealities, and the time-domain characteristics can be extracted from the realistic waveform. Fig. 5.10(c) shows the required OIP3H for the CUO and ORA to get a certain level of optimization accuracy in the time-domain characteristic optimization. To obtain around 1% accuracy, OIP3H of both blocks should be more than 10 dBm.

5.5.4 Analysis of the Effect of Noise

There are many noise sources in this platform. First, the thermal and flicker noises of the ESG, CUO, and ORA contribute to the total noise of this platform. Second, the ADC in this platform makes noise because of the effects of its quantization and its integral and differential nonlinearities (INL, DNL). Third, the digital computation block, which makes computation errors because of its finite bit-width, can be considered a noise source. Since the output of the CUO is a random number while the optimization algorithm is progressing, the errors that stem from the ADC and the digital computation block are randomized as well and can be classified as noise.

As discussed in Section 5.4, the SA-SS hybrid algorithm is utilized in this platform. Therefore, the effect of noise should be evaluated in the two phases (SA & SS) separately. For convenience, the relation between the SS algorithm and the circuit noise is discussed first. Fig. 5.11 shows the effect of noise in the SS phase. Even if the SS phase is started from the global optimum $(\overrightarrow{V_{opt}})$, $\overrightarrow{V_{opt}}$ can have a bigger cost than that of its neighbors temporarily because of the noise at $t = t_0$.
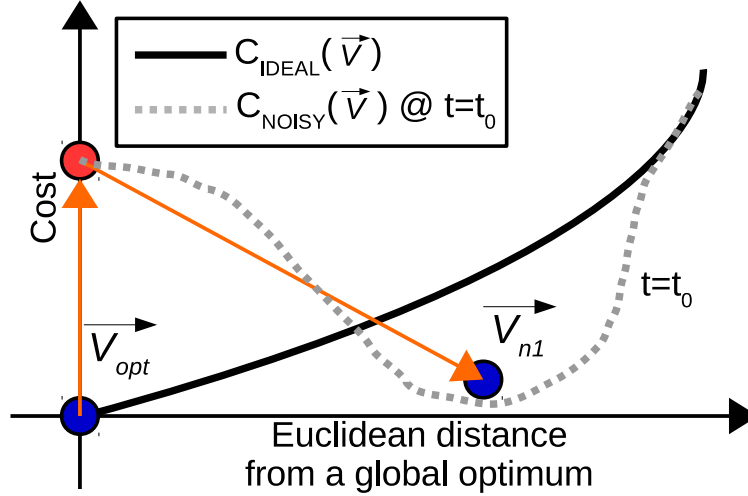
Figure 5.11: Effect of noise in sensitivity-search optimization. ©2018 IEEE.

In this case, $\overrightarrow{V_{opt}}$ will be substituted by one of its neighbors ($\overrightarrow{V_{n1}}$). By following the same process, $\overrightarrow{V_{n1}}$ can be replaced by another control vector at $t = t_1$. We can consider this phenomenon a hill climbing because the ideal cost of the newly selected control vector can be bigger than that of the previous vector. As this illustration shows, a current control vector will keep changing to its neighbor around $\overrightarrow{V_{opt}}$ due to the fluctuation of the cost. By recording the history of the selected control vectors and extracting the biggest %RMSE in the history, the worst-case %RMSE can be obtained in a simulation.

In the SA phase, hill climbing can occur and local minima can be escaped even if there is no circuit noise. Therefore, this natural hill-climbing action can be considered an "intentional noise" injected by the algorithm itself. If the total circuit noise of this platform is not very huge compared to the "intentional noise," the circuit noise is not going to degrade the quality of the SA optimization. However, near the end of the SA algorithm, the "intentional noise" is diminished a lot, and it might be comparable to the total circuit noise. For simplicity, we assume that the SA algorithm can find one of the sub-optimal control vectors ($\overrightarrow{V_{sopt}}$) regardless of the level of circuit noise. Then we can focus on the effect of circuit noise in the SS phase, which was discussed in the previous paragraph, and ignore the noise effect in the SA phase.

To simulate the effect of noise, an additive white noise model is exploited and all analog circuit
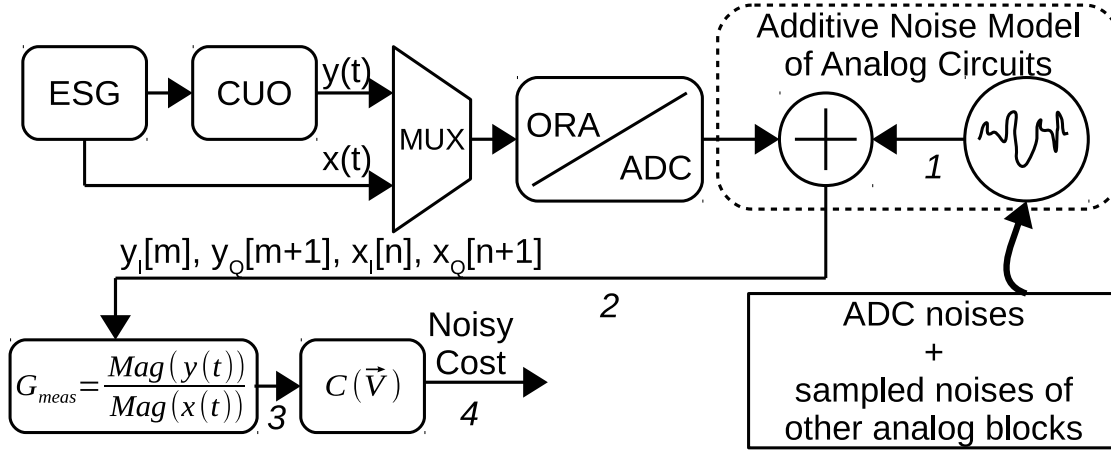
Figure 5.12: Block diagram for the noise analysis. ©2018 IEEE.

blocks are assumed noiseless as shown in Fig. 5.12. The model includes noise from the ADC and other analog blocks. Because the sampling speed of the ADC is much slower than the sampling speed of the ORA, the noise appeared at the input of the ADC will be heavily aliased by the sampling activities of the ADC. Therefore, it can be assumed that the sampled noise of the analog blocks has a white spectrum at the output of the ADC. Other errors that stem from the ADC itself can also be considered white noise because the input DC signal of the ADC can be any values while the optimization is progressing as mentioned earlier. Another thing that has to be mentioned here is that the phase noise of the sampling clock in the ORA will be ignored in this analysis. This can be justified because the input of the ORA also has the similar phase noise and those two noises are correlated. The correlation can be understood if we consider that the ESG and ORA receive clock signals from the same frequency synthesizer, and the ESG does not add the significant amount of phase noise because it is based on delay cells [66].

Simulation results are shown in Fig. 5.13. In these simulations, the same CUO was utilized as the previous analyses, and the worst-case %RMSE was extracted after 1000 sensitivity-search iterations. In addition, the errors from finite bit-width computations were ignored, and they will be analyzed in Section 5.5.5. As Fig. 5.13(a) shows, if the output power of the ESG (P_{ESG}) is 0 dBm and the target gain at ω_O ($G_{tar}(\omega_O)$) for the CUO is 0 dB, more than 47.5 dB P_{ESG}/P_{Noise}
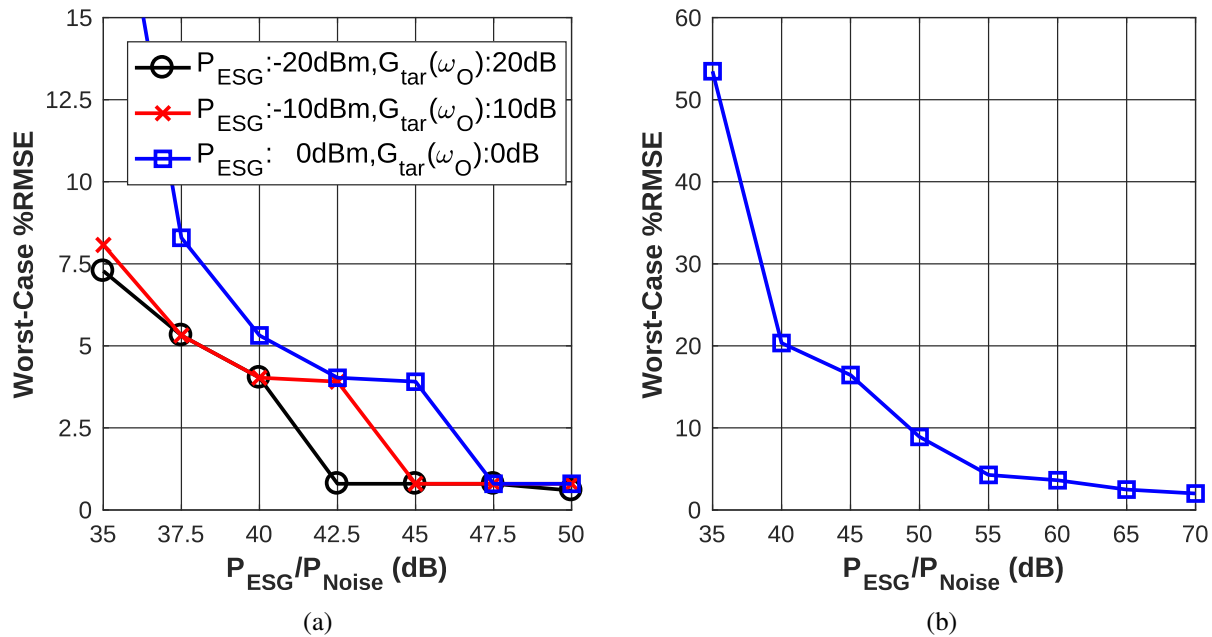
Figure 5.13: Simulation results that represent the relation between the %RMSE and P_{ESG}/P_{Noise}. (a) Simulation of the frequency-domain characterization. (b) Simulation of the time-domain characterization. $P_{ESG} = 8.6$ dBm, and $G_{tar}(\omega_O) = 0$ dB. ©2018 IEEE.

is required to get the %RMSE better than 1%, where P_{Noise} means the total noise power at node 1 in Fig. 5.12. This result means P_{Noise} should be smaller than -47.5 dBm. If P_{ESG} is reduced to -20 dBm and $G_{tar}(\omega_O)$ is increased to 20 dB, P_{Noise} should be reduced to the value smaller than -62.5 dBm to achieve better than 1% accuracy. This tough requirement can be relaxed by allowing a large signal that has power larger than 0 dBm at the output of the CUO. On top of that, averaging in the digital domain can be exploited. This will be discussed in Section 5.5.6.

Fig. 5.13(b) reveals the required P_{ESG}/P_{Noise} to achieve a certain level of accuracy for the time-domain characterization. In these simulations, a relatively large square wave (8.6 dBm) was exploited because most power of the wave is concentrated on a low-frequency range, and the low-frequency part is heavily attenuated by the CUO. If we consider that the time-domain characteristics are extracted from the output of the CUO only, the input power and the target gain of the CUO do not need to be changed while maintaining the output power of the CUO. As the simulation

results show, if the noise level is -47.5 dBm, the worst-case %RMSE will be around 9%.

5.5.5 Analysis of Bit Widths for Digital Computation Blocks

To compute a cost for the frequency-domain characterization, five arithmetic operations should be supported in the digital domain: addition, subtraction, multiplication, division, and square root. On top of these operations, the optimization algorithm shown in Algorithm 1 requires an exponential operation, random number generation, and other simple operations. Because the exponential operation can be approximated to a linear equation, it is not a mandatory operation. Also, a random number can be easily generated from a pseudo random number generator, such as a linear-feedback shift register. If we consider that all operations needed for the cost calculation and the optimization can be implemented in an area-efficient way except the multiplication, division, and square root operation, it can be assumed that all operations except those three can support full resolution. For instance, if we define one word as two bytes (16-bit), all operations except those three should support a 16-bit input, output, or both. Based on this assumption, we can focus only on the accuracy of the cost calculation. This is because the optimization algorithm shown in Algorithm 1 utilizes the three operations only at Step 10 and 14, and calculating a precise probability of a hill climbing at those steps is less important than getting an accurate cost.

Since the magnitude of $x(t)$ and $y(t)$ in Fig. 5.12 can be any real numbers, each computation block should support one of the two real number representations: fixed-point or floating-point. In this analysis, it is assumed that each computation block has a fixed-point representation because of its simplicity.

To quantify the error caused by the finite-bit width of each computation block, a signal-to-noise ratio (SNR) should be defined first. When the ideal and finite-bit-width output of each computation are Y and $\hat{Y}$, signal and noise power can be defined as a mean-square of Y and $\hat{Y} - Y$ individually as below.

$$P_{signal} = \frac{1}{n} \sum_{i=1}^{n} Y_i^2, \quad P_{error} = \frac{1}{n} \sum_{i=1}^{n} \left(\hat{Y}_i - Y_i \right)^2. \tag{5.20}$$
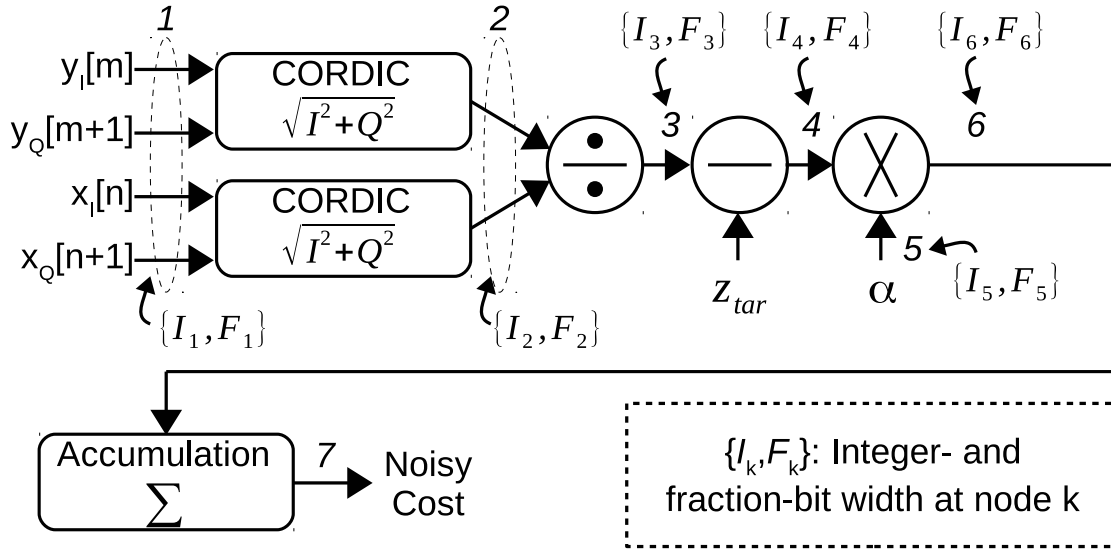
Figure 5.14: Digital computation flow and bit width at each node. ©2018 IEEE.

Then the SNR can be expressed as a ratio of the two.

Fig. 5.14 shows the entire digital computation flow and the bit width of each block in the frequency-domain characterization. Each node has an I integer-bit width and F fraction-bit width. Since the addition and subtraction support full resolution (16-bit) as mentioned earlier, the bits at node 3 and 4 should be expended and truncated, respectively.

While changing the bit widths, the SNR at node 3 and 7 in Fig. 5.14 can be calculated. Because there are too many variables, some assumptions should be made as summarized in Table 5.1. In addition, the table shows simulation results, which reveal the relation between the SNR and the bit widths (F_2, F_3). In these simulations, x_I and x_Q were sampled from a 0 dBm sinusoidal signal. Also, y_I and y_Q were extracted from a sinusoidal signal which has random power from -20 dBm to 6 dBm. The phases of the sampling clocks for $x(t)$ and $y(t)$ were given randomly as well. As the table indicates, F_2 and F_3 should be larger than or equal to 2 and 9, respectively, to achieve around 55 dB SNR at nodes 3 and 7.

To compare noise that is generated from the analog circuits and the digital computations, the SNR transfer from node 1 to nodes 3 and 7 in Fig. 5.14 should be evaluated. We are considering

Table 5.1: Remarks and Simulation Results of the Bit-Width Analysis for Digital Computations. ©2018 IEEE.

Item	Value	Remark
I_1	10	Discussed in Section 5.5.6
F_1	0	Assume that the output of an ADC is an integer
I_2	11	Maximum output value of CORDIC = $\sqrt{2}\times$ maximum input value of CORDIC
F_2		Varied in simulations
I_3	2	Maximum gain of a CUO = 12 dB
F_3		Varied in simulations
I_4	I_3	An assumption for simplicity
F_4	F_3	An assumption for simplicity ($F_3=F_4=F_6$)
I_5	2	Maximum value for $\alpha = 2$
F_5	8	Dynamic range for $\alpha = 9$-bit
I_6	3	$I_4(=I_3) = 2$, and maximum value for $\alpha = 2$
F_6	F_4	An assumption for simplicity ($F_3=F_4=F_6$)
Item	**Value**	**Simulation Result**
F_2, F_3	(0,6)	(SNR@3, SNR@7) = (39.9 dB, 39.0 dB)
	(0,9)	(SNR@3, SNR@7) = (44.3 dB, 43.4 dB)
	(1,6)	(SNR@3, SNR@7) = (41.3 dB, 38.9 dB)
	(1,9)	(SNR@3, SNR@7) = (51.0 dB, 50.6 dB)
	(2,6)	(SNR@3, SNR@7) = (41.7 dB, 38.7 dB)
	(2,9)	**(SNR@3, SNR@7) = (55.0 dB, 54.3 dB)**
	(3,6)	(SNR@3, SNR@7) = (41.8 dB, 38.3 dB)
	(3,9)	(SNR@3, SNR@7) = (58.0 dB, 55.7 dB)

here a SNR transfer instead of a noise transfer because the power of the signal is also converted while it is processed by the digital circuits. From a computer simulation, a 0 dB SNR transfer was observed at nodes 3 and 7 when there were no errors that come from the digital blocks. Therefore, if we have a 50 dB SNR at node 1, the SNR will be still 50 dB at node 7. If we consider that the total power of errors produced by the digital circuits will be 54.3 dB smaller than the signal power at node 7 when $(F_2, F_3) = (2, 9)$, the total SNR at node 7 including all noise from the analog circuits and the digital computation errors will be 48.6 dB, which is equivalent to the %RMSE smaller than 1% from Fig. 5.13.

In the time-domain characterization, computations of a square root and division are not needed because the magnitudes and the gain do not need to be calculated anymore. Therefore, the output of the ADC and node 3 in Fig. 5.14 should be connected directly. If $F_4(=F_6)$ is bigger than or equal to 9, the SNR at node 7 will be better than 54.3 dB at least because the results in Table 5.1 include

Table 5.2: Summary of Noise and Linearity Requirements. ©2018 IEEE.

| P_{ESG} | Averaging | P_{Noise} Before Averaging | | P_{Noise} After Averaging | Required Max ENOB for an ADC | Min |THD| for an ESG and ORA | Min |THD| for a CUO |
		$P_{Noise,Ana}$	$P_{Noise,ADC}$				
0 dBm	X	−53 dBm	−53 dBm	N/A	10.47-bit	40 dB	30 dB
6 dBm	X	−47 dBm	−47 dBm	N/A	9.47-bit	40 dB	30 dB
0 dBm	4-point	−47 dBm	−47 dBm	−50 dBm	9.47-bit	40 dB	30 dB

all errors that originate from the CORDIC and the divider.

5.5.6 Overall Linearity & Noise Requirements and Averaging

Noise and linearity requirements are summarized in Table 5.2. When P_{ESG} is 0 dBm, P_{Noise} should be smaller than −50 dBm to obtain 1% accuracy in the frequency-domain characterization as discussed in Section 5.5.4. If we divide this specification evenly between the ADC ($P_{Noise,ADC}$) and the other analog circuit blocks ($P_{Noise,Ana}$), each part should have smaller than −53 dBm noise power. If we assume that the peak SNDR of the ADC can be obtained at 0 dBFS, the required maximum ENOB for the ADC will be 10.47-bit for the 1.2 V supply voltage because 0 dBFS = 11.6 dBm.

There are two approaches that relax the noise requirement. The first approach is increasing P_{ESG}. For example, if P_{ESG} is increased up to 6 dBm, the required ENOB of the ADC can be 9.47-bit as shown in Table 5.2. However, maintaining −40 dB THD for the ESG and the ORA, and −30 dB THD for the CUO will be more demanding because the power of harmonic tones grow more quickly than the power of a main tone. The second approach is averaging the outputs of the ADC. If the window size for the averaging is 2^n, the reduction of noise power is $3n$ dB. Therefore, the required SNR at the output of the ADC can be relaxed to 44 dB when 4-point averaging is utilized. In this scenario, the required ENOB for the ADC will be 9.47-bit when $P_{ESG} = 0$ dBm.

In the time-domain characteristic optimization, a full-scale square wave can be utilized to stimulate the CUO because the linearity requirements are more relaxed compared to the requirements of the frequency-domain characterization as shown in Fig. 5.10. In this case, P_{ESG}/P_{Noise} can be more than 60 dB if P_{Noise} equals −50 dBm and the 1.2 V supply voltage is utilized. Overall, all requirements summarized in Table 5.2 can also guarantee around 1% accuracy in the time-domain

characterization, if Δ_t is small enough.

5.6   System Verification

To show the feasibility of this platform, the Tow-Thomas bandpass biquad shown in Fig. 5.5(a) is utilized in this case study. Unlike the conventional tuning/calibration approaches introduced in [56–58], the biquad includes a control knob that changes the GBW of op-amps. Because this platform does not depend on any characteristics of linear time-invariant CUOs, we choose the biquad as a simple example in this section to clearly prove the concept of this platform. Once it proves that the simple example can be optimized, this platform can be applicable to more complex CUOs without an additional area overhead. For instance, if a biquad can be optimized in this platform, high-order filters that consist of any number of cascaded biquads can also be optimized by the same process.

Also, this platform is not very complex for analog circuit designers to exploit. Once the designers have an accurate model of this platform, which is discussed in Section 5.5, the underlying algorithm (SA-SS) of this system can be designed and be verified relatively easily in simulations because it has only a few parameters as discussed in Section 5.4. The only things the designers need to focus on are developing a suitable cost function based on specifications and determining appropriate control knobs.

5.6.1   Verification Through System-Level Simulations

A realistic model of the biquad is used in this system-level simulations. If we model each op-amp in Fig. 5.5(a) as a two-pole system and assume that it has a 50 dB DC gain and second pole at GBW, the filter transfer function will have six poles and three zeros. Based on the system equation, a mathematical model for the filter can be developed and utilized. This model has four control knobs $[x'_{GBW}, x'_G, x'_Q, x'_{\omega_O}]$, and each one has 5 bits. The 1-LSB and the center value for x'_{GBW} are 35 MHz and 600 MHz, respectively. x'_G and x'_Q change R_1 and R_3 individually and have 1 KΩ 1-LSBs and 22 KΩ center values. x'_{ω_O} modifies C_1 and C_2 simultaneously and has 50 fF 1-LSB and 1.95 pF center value. The relatively small 1-LSBs for the R & C components are

used on purpose to show the fine optimization capability of this platform at certain levels of noise and distortions.

In addition, to mimic PVT variations and device aging, the simulations include several non-ideal factors. Normal distributions that have 10% standard deviations (σ) of original values are applied to the 1-LSBs and to the center values of all R & C components and the GBW. Also, the simulations meet the noise and linearity requirements shown in the second row of Table 5.2.

Fig. 5.16(a)-(f) represent 100 Monte-Carlo simulation results of the case study. In the figure, white bars depict characteristics of 100 samples before optimization, whereas black bars stand for characteristics after optimization. Based on the data distribution, approximated probability density functions (PDFs) are plotted in the form of solid and dotted curves on top of the histogram. Also, outliers are excluded to obtain a well-matched solid curve around a mean value. As the figure indicates, on average, the GBW of op-amps and the σ of the five characteristics are reduced by 80% and 82%, respectively. If we consider that the ratio between the GBW and $\omega_O/2\pi$ is 36.3 in [58] and 71.1 in [9], the optimized CUO has relatively small GBW ($GBW/(\omega_O/2\pi) = 22.9$) without employing any circuit techniques. Since the power consumption (GBW) of each sample is minimized while the same five frequency- and time-domain characteristics are maintained, the actual operating point of each sample is located at the edge of its operation limit, which verifies the strength of this platform. The relatively large deviation in the GBW after optimization can be understood if we recall that the value is optimized indirectly as a penalty in the cost function.

In the system-level simulations, only one trial of optimization takes place for each sample. Even though the results show small characteristic deviations after optimization, there are still a small number of outliers in the histogram. These outliers cannot be avoided 100% because Algorithm 1 operates on the basis of randomness. Therefore, multiple trials are needed for the exceptional outliers as discussed in Section 5.4.

Because the evaluations of $\overrightarrow{V}$ take most time in the entire optimization process, the total number of the evaluations determines the algorithm efficiencies in comparison between the proposed and the other approaches introduced in Section 5.4. From the simulation results, one trial of op-
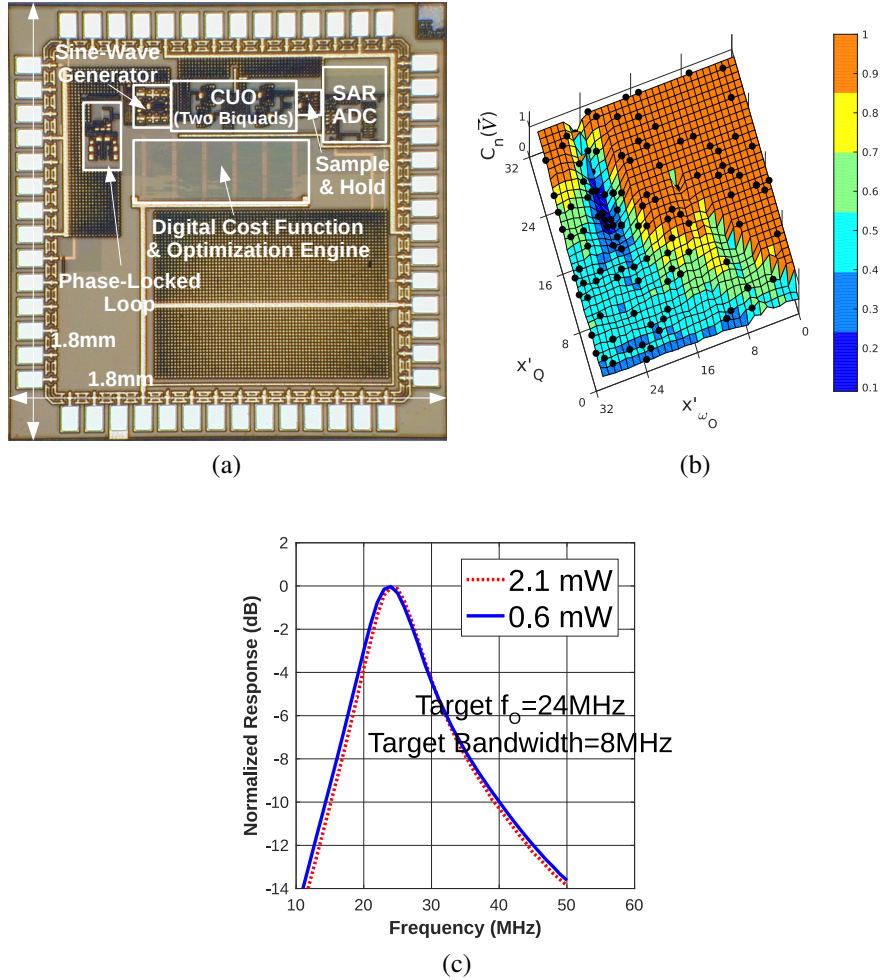
Figure 5.15: Integrated circuit prototype and measurement results. (a) Die photogarph. (b) Normalized cost function values over the 2-dimensional search space, and visited points selected by the optimization engine (dots). (c) Optimization result when the biquad has high power consumption (dotted line) & minimum power consumption (solid line). ©2018 IEEE.

timization requires around 1500 evaluations. If we compare this number to the values from the brute-force method and from the semi-orthogonal tuning in Section 5.4, more than 99% and 50% reductions are observed, respectively.

5.6.2 Integrated Circuit Prototype & Measurement Results

Fig. 5.15(a) shows an IC prototype of the self-contained system. The prototype was fabricated in 0.18 μm standard CMOS technology. Fig. 5.15(b) presents the measured values of the cost
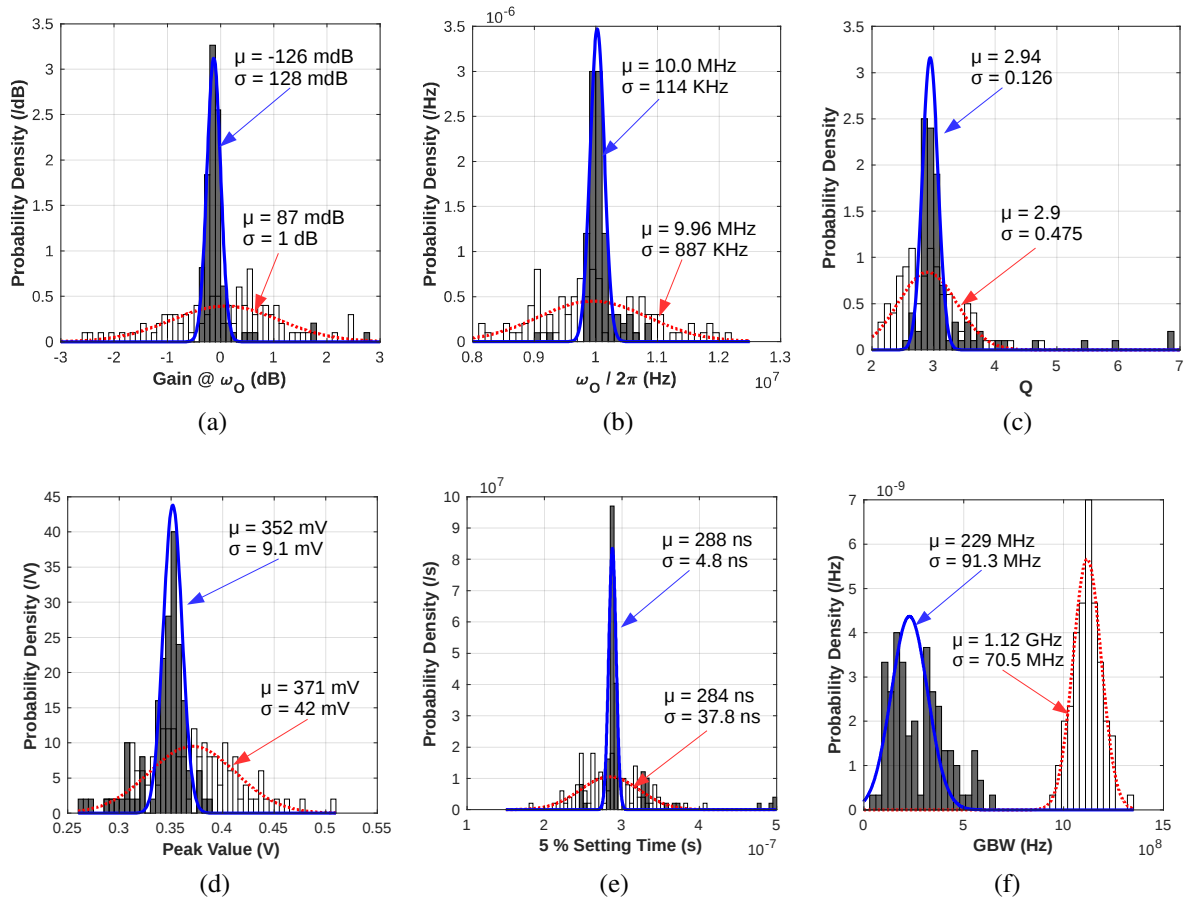
Figure 5.16: Reduction of power consumption and standard deviations of multiple characteristics of a biquad. White bars: 100 samples before optimization; Black bars: 100 samples after optimization. (a) Gain @ ω_O. (b) $\omega_O/2\pi$. (c) Q. (d) Peak value when a step input is applied. (e) 5% settling time. (f) GBW. ©2018 IEEE.

function over 2-dimensional search space. For convenience, x'_{GBW} and x'_G are fixed. In this figure, black dots indicate visited points selected by the optimization engine. Because of the finite bit width discussed in Section 5.5.5, cost function values that are larger than the maximum limit are trimmed to the maximum value. As we can see, an intensive search is conducted around the optimal point $[x'_Q, x'_{\omega_O}]=[20, 23]$. After optimization, around 71% power reduction can be achieved while other biquad specifications are maintained as shown in Fig. 5.15(c).

Table 5.3 compares this platform with other tuning platforms that use optimization algorithms. The main contribution of this paper is that all building blocks, including an ESG, an ORA, and an

Table 5.3: Comparison of Tuning/Calibration Platforms Which Utilize Optimization Algorithms. ©2018 IEEE.

Ref.	Algorithm	Algorithm complexity (the number of If-Else-Then)	Chance to find a global optimum	Fully integrated on a chip	Accuracy analyses of building blocks
[61]	MSGD*	Low	Low if control knobs are non-orthogonal [62]	No	No
[62]	NM-NS** Hybrid	High	Low in general, but can be high with an initial grid	No	No
[63]	NM-HJ*** Hybrid	High	Low in general, but can be high with an initial grid	No	No
[64]	GA	High	High (global optimizer)	No	No
This work	**SA-SS Hybrid**	**Low**	**High (hybird of global and local optimizers)**	**Yes**	**Yes**

*Multi-Start Gradient Descent ** Nelder-Mead Neighborhood-Search *** Nelder-Mead Hooke-Jeeves

optimization engine, are integrated in a single chip, proving the on-chip, in-situ operation of this platform. The optimization algorithm is selected in terms of the efficiency of hardware implementation.

5.6.3 Strengths of This Platform

Operating the Tow-Thomas (TT) biquad efficiently is a very active research topic. Unlike [9, 58], our work does not rely on a master-slave approach; our optimization can be beyond the conventional Q and ω_O tuning and can use GBW as a design parameter. Thus, we can drastically reduce the GBW while monitoring that the filter is stable and meets for instance the Q and ω_O specifications. Other specifications such as linearity requirements can be accomplished by increasing the minimum GBW without having excessive margins. That is equivalent to finding the minimum power consumption that satisfies all requirements. In addition, thanks to the versatility and the efficiency of the optimization engine, various characteristics of a CUO can be programmed based on users' need within ranges of control knobs. To the best of authors' knowledge, this approach was not available before.

5.7 Conclusion

A built-in self-test and in-situ analog circuit optimization platform has been proposed and characterized. Different from the conventional on-chip direct tuning/calibration methods dedicated to a specific characteristic, this platform seamlessly and efficiently optimizes programmable circuit characteristics as a whole. As a result, the CUO can have and maintain well-balanced optimal

characteristics even in severe PVT variations and device aging. Because this platform does not depend on special characteristics of the CUO, any linear time-invariant circuits can be the CUO.

6. CONCLUSION

This dissertation investigates two approaches that overcome the limitations of previous works on in situ analog circuit calibration/tuning. The first limitation is that there are still several analog circuits that have not been calibrated/tuned by an in situ automatic calibration/tuning circuit. [3] gets over the limitation by proposing an in situ calibration circuit for a current reference to relax the trade-off between the accuracy and the power consumption of the current reference. To the best of the author's knowledge, [3] is the first work that shows a successful in situ calibration of a current reference while keeping the power consumption of the current reference down to 4.5 nW at the same time. The second limitation is that calibration/tuning techniques that have been researched so far deal with only one circuit characteristic, leading to a suboptimal circuit. [4] surmounts the limitation by proposing an in situ analog circuit optimization platform that optimizes multiple competing circuit characteristics simultaneously. To the best of the author's knowledge, the platform is the first fully-integrated in situ optimization system for analog circuits. The measurement and simulation results of the circuits proposed in [3, 4] prove that the calibration and optimization techniques can relax sharp trade-offs among circuit characteristics and help analog circuits achieve better performance and robustness. As these research show, in situ analog circuit calibration and optimization techniques have been less researched and still have huge research potential. For example, an in situ analog circuit optimization platform that can optimize the linearity, noise, and power consumption of multiple analog/RF circuits can be a good extension of [4] because the platform can be utilized to optimize performance of an RF transceiver over PVT variations.

REFERENCES

[1] S. Borkar, "Designing reliable systems from unreliable components: the challenges of transistor variability and degradation," *IEEE Micro*, vol. 25, no. 6, pp. 10–16, Nov 2005.

[2] K. J. Kuhn, M. D. Giles, D. Becher, P. Kolar, A. Kornfeld, R. Kotlyar, S. T. Ma, A. Maheshwari, and S. Mudanai, "Process technology variation," *IEEE Transactions on Electron Devices*, vol. 58, no. 8, pp. 2197–2208, Aug 2011.

[3] S. Lee, S. Heinrich-Barna, K. Noh, K. Kunz, and E. Sánchez-Sinencio, "A 1-nA 4.5-nW 289-ppm/°C current reference using automatic calibration," *IEEE Journal of Solid-State Circuits*, 2020.

[4] S. Lee, C. Shi, J. Wang, A. Sanabria, H. Osman, J. Hu, and E. Sánchez-Sinencio, "A built-in self-test and in situ analog circuit optimization platform," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 65, no. 10, pp. 3445–3458, 2018.

[5] B. Razavi, *Design of Analog CMOS Integrated Circuits*, 1st ed. McGraw-Hill Education, 2000.

[6] P. R. Gray, P. J. Hurst, S. H. Lewis, and R. G. Meyer, *Analysis and Design of Analog Integrated Circuits*, 4th ed. Wiley, 2001.

[7] D. A. Badillo, "1.5V CMOS current reference with extended temperature operating range," in *2002 IEEE International Symposium on Circuits and Systems (ISCAS)*, vol. 3, 2002, pp. III–III.

[8] Jiwei Chen and Bingxue Shi, "1 V CMOS current reference with 50 ppm/°C temperature coefficient," *Electronics Letters*, vol. 39, no. 2, pp. 209–210, 2003.

[9] C. Wu, W. L. Goh, C. L. Kok, W. Yang, and L. Siek, "A low TC, supply independent and process compensated current reference," in *2015 IEEE Custom Integrated Circuits Conference (CICC)*, 2015, pp. 1–4.

[10] B. Yang, Y. Shin, J. Lee, Y. Lee, and K. Ryu, "An accurate current reference using temperature and process compensation current mirror," in *2009 IEEE Asian Solid-State Circuits Conference*, 2009, pp. 241–244.

[11] W. M. Sansen, F. Op't Eynde, and M. Steyaert, "A CMOS temperature-compensated current reference," *IEEE Journal of Solid-State Circuits*, vol. 23, no. 3, pp. 821–824, 1988.

[12] J. Georgiou and C. Toumazou, "A resistorless low current reference circuit for implantable devices," in *2002 IEEE International Symposium on Circuits and Systems (ISCAS)*, vol. 3, 2002, pp. III–III.

[13] T. Shima, "Temperature insensitive current reference circuit using standard CMOS devices," in *2007 Midwest Symposium on Circuits and Systems (MWSCAS)*, 2007, pp. 181–184.

[14] R. Dehghani and S. M. Atarodi, "A new low voltage precision CMOS current reference with no external components," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 50, no. 12, pp. 928–932, 2003.

[15] F. Fiori and P. S. Crovetti, "A new compact temperature-compensated CMOS current reference," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 52, no. 11, pp. 724–728, 2005.

[16] E. Vittoz and J. Fellrath, "CMOS analog integrated circuits based on weak inversion operations," *IEEE Journal of Solid-State Circuits*, vol. 12, no. 3, pp. 224–231, 1977.

[17] H. J. Oguey and D. Aebischer, "CMOS current reference without resistance," *IEEE Journal of Solid-State Circuits*, vol. 32, no. 7, pp. 1132–1135, 1997.

[18] E. M. Camacho-Galeano, C. Galup-Montoro, and M. C. Schneider, "A 2-nW 1.1-V self-biased current reference in CMOS technology," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 52, no. 2, pp. 61–65, 2005.

[19] G. De Vita and G. Iannaccone, "A 109 nW, 44 ppm/°C CMOS current reference with low sensitivity to process variations," in *2007 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2007, pp. 3804–3807.

[20] T. Hirose, Y. Osaki, N. Kuroki, and M. Numa, "A nano-ampere current reference circuit and its temperature dependence control by using temperature characteristics of carrier mobilities," in *2010 IEEE European Solid State Circuits Conference (ESSCIRC)*, Sep. 2010, pp. 114–117.

[21] Z. Huang, Q. Luo, and Y. Inoue, "A CMOS sub-l-V nanopower current and voltage reference with leakage compensation," in *2010 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2010, pp. 4069–4072.

[22] K. Ueno, T. Hirose, T. Asai, and Y. Amemiya, "A 1-µW 600-ppm/°C current reference circuit consisting of subthreshold CMOS circuits," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 57, no. 9, pp. 681–685, Sep. 2010.

[23] I. M. Filanovsky and A. Allam, "Mutual compensation of mobility and threshold voltage temperature effects with applications in CMOS circuits," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 48, no. 7, pp. 876–884, 2001.

[24] A. Bendali and Y. Audet, "A 1-V CMOS current reference with temperature and process compensation," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 54, no. 7, pp. 1424–1429, 2007.

[25] Q. Dong, I. Lee, K. Yang, D. Blaauw, and D. Sylvester, "A 1.02nW PMOS-only, trim-free current reference with 282ppm/°C from -40°C to 120°C and 1.6% within-wafer inaccuracy," in *2017 IEEE European Solid State Circuits Conference (ESSCIRC)*, Sep. 2017, pp. 19–22.

[26] K. Ueno, Tetsuya Hirose, Tetsuya Asai, and Yoshihito Amemiya, "A 46-ppm/°C temperature and process compensated current reference with on-chip threshold voltage monitoring circuit," in *2008 IEEE Asian Solid-State Circuits Conference*, 2008, pp. 161–164.

[27] M. Choi, I. Lee, T. Jang, D. Blaauw, and D. Sylvester, "A 23pW, 780ppm/°C resistor-less current reference using subthreshold mosfets," in *2014 IEEE European Solid State Circuits Conference (ESSCIRC)*, 2014, pp. 119–122.

[28] J. Lee and S. Cho, "A 1.4-µW 24.9-ppm/°C current reference with process-insensitive temperature compensation in 0.18-µm CMOS," *IEEE Journal of Solid-State Circuits*, vol. 47, no. 10, pp. 2527–2533, Oct 2012.

[29] G. Serrano and P. Hasler, "A precision low-TC wide-range CMOS current reference," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 2, pp. 558–565, 2008.

[30] Y. Ji, C. Jeon, H. Son, B. Kim, H. Park, and J. Sim, "9.3nW all-in-one bandgap voltage and current reference circuit," in *2017 IEEE International Solid-State Circuits Conference (ISSCC)*, Feb 2017, pp. 100–101.

[31] H. Wang and P. P. Mercier, "A 14.5 pW, 31 ppm/°C resistor-less 5 pA current reference employing a self-regulated push-pull voltage reference generator," in *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2016, pp. 1290–1293.

[32] ——, "A 3.4-pW 0.4-V 469.3 ppm/°C five-transistor current reference generator," *IEEE Solid-State Circuits Letters*, vol. 1, no. 5, pp. 122–125, May 2018.

[33] J. Santamaria, N. Cuevas, G. L. E. Rueda, J. Ardila, and E. Roa, "A family of compact trim-free CMOS nano-ampere current references," in *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2019, pp. 1–4.

[34] J. Rabaey, *Low Power Design Essentials*, 1st ed.   Springer, 2009.

[35] C. C. Enz and G. C. Temes, "Circuit techniques for reducing the effects of op-amp imperfections: autozeroing, correlated double sampling, and chopper stabilization," *Proceedings of the IEEE*, vol. 84, no. 11, pp. 1584–1614, Nov 1996.

[36] J. M. Rabaey, A. Chandrakasan, and B. Nikolic, *Digital Integrated Circuits: A Design Perspective*, 2nd ed.   Pearson, 2003.

[37] I. Lee, D. Sylvester, and D. Blaauw, "A subthreshold voltage reference with scalable output voltage for low-power IoT systems," *IEEE Journal of Solid-State Circuits*, vol. 52, no. 5, pp. 1443–1449, May 2017.

[38] M. Seok, G. Kim, D. Blaauw, and D. Sylvester, "A portable 2-transistor picowatt temperature-compensated voltage reference operating at 0.5 V," *IEEE Journal of Solid-State Circuits*, vol. 47, no. 10, pp. 2534–2545, Oct 2012.

[39] T. Jang, G. Kim, B. Kempke, M. B. Henry, N. Chiotellis, C. Pfeiffer, D. Kim, Y. Kim, Z. Foo, H. Kim, A. Grbic, D. Sylvester, H. Kim, D. D. Wentzloff, and D. Blaauw, "Circuit and system designs of ultra-low power sensor nodes with illustration in a miniaturized GNSS logger for position tracking: Part I-analog circuit techniques," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 64, no. 9, pp. 2237–2249, 2017.

[40] M. Qu, J. Wan, and X. Hao, "Analysis of diurnal air temperature range change in the continental united states," *Weather and Climate Extremes*, vol. 4, pp. 86–95, Aug 2014.

[41] "North america land data assimilation system (NLDAS) daily air temperatures and heat index," 2013, CDC WONDER online database https://wonder.cdc.gov/NASA-NLDAS.html.

[42] J. G. Houghton, *Nevada's Weather and Climate*. Nevada Bureau of Mines and Geology, 1975, p. 28.

[43] D. Roberts and K. Lay, "Variability in measured space temperatures in 60 homes," National Renewable Energy Laboratory, Tech. Rep. NREL/TP-5500-58059, March 2013.

[44] M. Seok, S. Hanson, Y.-S. Lin, Z. Foo, D. Kim, Y. Lee, N. Liu, D. Sylvester, and D. Blaauw, "The phoenix processor: A 30pW platform for sensor applications," in *2008 IEEE Symposium on VLSI Circuits*, June 2008, pp. 188–189.

[45] K. Sun, T.-S. Wei, B. Y. Ahn, J. Y. Seo, S. J. Dillon, and J. A. Lewis, "3D printing of interdigitated Li-Ion microbattery architectures," *Advanced Materials*, vol. 25, no. 33, pp. 4539–4543, 2013.

[46] J. Lim, T. Jang, M. Saligane, M. Yasuda, S. Miyoshi, M. Kawaminami, D. Blaauw, and D. Sylvester, "A 224 pW 260 ppm/°C gate-leakage-based timer for ultra-low power sensor nodes with second-order temperature dependency cancellation," in *2018 IEEE Symposium on VLSI Circuits*, June 2018, pp. 117–118.

[47] H. E. Graeb, *Analog Design Centering and Sizing*.   Springer, 2010.

[48] K. Antreich and R. Koblitz, "Design centering by yield prediction," *IEEE Transactions on Circuits and Systems*, vol. 29, no. 2, pp. 88–96, Feb 1982.

[49] G. Roberts, F. Taenzler, and M. Burns, *An Introduction to Mixed-Signal IC Test and Measurement*, 2nd ed.   Oxford University Press, 2011.

[50] L. S. Milor, "A tutorial introduction to research on analog and mixed-signal circuit testing," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 45, no. 10, pp. 1389–1407, Oct 1998.

[51] M. Andraud, H. G. Stratigopoulos, and E. Simeu, "One-shot non-intrusive calibration against process variations for analog/RF circuits," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 63, no. 11, pp. 2022–2035, Nov 2016.

[52] S. Sun, F. Wang, S. Yaldiz, X. Li, L. Pileggi, A. Natarajan, M. Ferriss, J. O. Plouchart, B. Sadhu, B. Parker, A. Valdes-Garcia, M. A. T. Sanduleanu, J. Tierno, and D. Friedman, "Indirect performance sensing for on-chip self-healing of analog and RF circuits," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, no. 8, pp. 2243–2252, Aug 2014.

[53] D. Han, B. S. Kim, and A. Chatterjee, "DSP-driven self-tuning of RF circuits for process-induced performance variability," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 18, no. 2, pp. 305–314, Feb 2010.

[54] T. Das, A. Gopalan, C. Washburn, and P. R. Mukund, "Self-calibration of input-match in RF front-end circuitry," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 52, no. 12, pp. 821–825, Dec 2005.

[55] X. Fan, M. Onabajo, F. O. Fernandez-Rodriguez, J. Silva-Martinez, and E. Sanchez-Sinencio, "A current injection built-in test technique for RF low-noise amplifiers," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 55, no. 7, pp. 1794–1804, Aug 2008.

[56] P. Kallam, E. Sanchez-Sinencio, and A. I. Karsilayan, "An enhanced adaptive Q-tuning scheme for a 100-MHz fully symmetric OTA-based bandpass filter," *IEEE Journal of Solid-State Circuits*, vol. 38, no. 4, pp. 585–593, Apr 2003.

[57] B. Xia, S. Yan, and E. Sanchez-Sinencio, "An RC time constant auto-tuning structure for high linearity continuous-time $\Delta\Sigma$ modulators and active filters," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 51, no. 11, pp. 2179–2188, Nov 2004.

[58] S. Kousai, M. Hamada, R. Ito, and T. Itakura, "A 19.7 MHz, fifth-order active-RC chebyshev LPF for draft IEEE802.11n with automatic quality-factor tuning scheme," *IEEE Journal of Solid-State Circuits*, vol. 42, no. 11, pp. 2326–2337, Nov 2007.

[59] M. M. Hafed, N. Abaskharoun, and G. W. Roberts, "A 4-GHz effective sample rate integrated test core for analog and mixed-signal circuits," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 4, pp. 499–514, Apr 2002.

[60] A. Valdes-Garcia, F. A. L. Hussien, J. Silva-Martinez, and E. Sanchez-Sinencio, "An integrated frequency response characterization system with a digital interface for analog testing," *IEEE Journal of Solid-State Circuits*, vol. 41, no. 10, pp. 2301–2313, Oct 2006.

[61] S. Devarakond, S. Sen, A. Banerjee, and A. Chatterjee, "Digitally assisted built-in tuning using hamming distance proportional signatures in RF circuits," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 9, pp. 2918–2931, Sept 2016.

[62] E. J. Wyers, M. B. Steer, C. T. Kelley, and P. D. Franzon, "A bounded and discretized nelder-mead algorithm suitable for RFIC calibration," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 60, no. 7, pp. 1787–1799, July 2013.

[63] E. J. Wyers, M. A. Morton, T. C. L. G. Sollner, C. T. Kelley, and P. D. Franzon, "A generally applicable calibration algorithm for digitally reconfigurable self-healing RFICs," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 24, no. 3, pp. 1151–1164, Mar 2016.

[64] M. Murakawa, T. Adachi, Y. Niino, Y. Kasai, E. Takahashi, K. Takasuka, and T. Higuchi, "An AI-calibrated IF filter: a yield enhancement method with area and power dissipation reductions," *IEEE Journal of Solid-State Circuits*, vol. 38, no. 3, pp. 495–502, Mar 2003.

[65] J. Wang, C. Shi, E. Sanchez-Sinencio, and J. Hu, "Built-in self optimization for variation resilience of analog filters," in *2015 IEEE Computer Society Annual Symposium on VLSI*, July 2015, pp. 656–661.

[66] C. Shi and E. Sanchez-Sinencio, "150-850 MHz high-linearity sine-wave synthesizer architecture based on FIR filter approach and SFDR optimization," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 62, no. 9, pp. 2227–2237, Sept 2015.

[67] S. S. Haykin and M. Moher, *Communication Systems, 4E.* Hoboken, N.J. : John Wiley & Sons, 2001, pp. 95–98.

[68] N. S. Nise, *Control Systems Engineering.* John Wiley & Sons, 2007.

[69] T. Back, D. B. Fogel, and Z. Michalewicz, *Handbook of Evolutionary Computation.* Oxford University Press, 1997.

[70] S. Sen, D. Banerjee, M. Verhelst, and A. Chatterjee, "A power-scalable channel-adaptive wireless receiver based on built-in orthogonally tunable LNA," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 59, no. 5, pp. 946–957, May 2012.

[71] Y. Wang, M. Orshansky, and C. Caramanis, "Enabling efficient analog synthesis by coupling sparse regression and polynomial optimization," in *Proceedings of 51st ACM/EDAC/IEEE Design Automation Conference (DAC)*, June 2014, pp. 1–6.

[72] E. Aarts and J. K. Lenstra, *Local Search in Combinatorial Optimization.* Princeton University Press, 2003.

[73] D. Banerjee, B. Muldrey, X. Wang, S. Sen, and A. Chatterjee, "Self-learning RF receiver systems: Process aware real-time adaptation to channel conditions for low power operation," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 64, no. 1, pp. 195–207, Jan 2017.

[74] P. Wambacq and W. M. Sansen, *Distortion Analysis of Analog Integrated Circuits*. Springer, 1998.