sensors

MDPI

Article

Accurate Behavioral Simulator of All-Digital Time-Domain Smart Temperature Sensors by Using SIMULINK

Chun-Chi Chen *, Chao-Lieh Chen and You-Ting Lin

Department of Electronic Engineering, National Kaohsiung First University of Science and Technology, Kaohsiung 811, Taiwan; frederic@nkfust.edu.tw (C.-L.C.); u0452804@nkfust.edu.tw (Y.-T.L.)

* Correspondence: ccchen@nkfust.edu.tw; Tel.: +886-7-601-1000 (ext. 2513)

Academic Editor: Vittorio M. N. Passaro

Received: 6 June 2016; Accepted: 2 August 2016; Published: 8 August 2016

Abstract: This study proposes a new behavioral simulator that uses SIMULINK for all-digital CMOS time-domain smart temperature sensors (TDSTSs) for performing rapid and accurate simulations. Inverter-based TDSTSs offer the benefits of low cost and simple structure for temperature-to-digital conversion and have been developed. Typically, electronic design automation tools, such as HSPICE, are used to simulate TDSTSs for performance evaluations. However, such tools require extremely long simulation time and complex procedures to analyze the results and generate figures. In this paper, we organize simple but accurate equations into a temperature-dependent model (TDM) by which the TDSTSs evaluate temperature behavior. Furthermore, temperature-sensing models of a single CMOS NOT gate were devised using HSPICE simulations. Using the TDM and these temperature-sensing models, a novel simulator in SIMULINK environment was developed to substantially accelerate the simulation and simplify the evaluation procedures. Experiments demonstrated that the simulation results of the proposed simulator have favorable agreement with those obtained from HSPICE simulations, showing that the proposed simulator functions successfully. This is the first behavioral simulator addressing the rapid simulation of TDSTSs.

Keywords: CMOS; all digital; time-domain; smart temperature sensor; SIMULINK; HSPICE

1. Introduction

Computer-aided design (CAD) tools are often used to facilitate electronic circuit design before implementation. They are developed to achieve numerous goals such as the reuse of design components, ease of design modification, automatic generation of designs, and verification of designs against specifications and design rules. The ultimate purpose is to derive predictions of circuit behavior that are highly similar or even identical to the real physical behavior of the circuit after implementation. For analog or mixed-signal integrated circuit (IC) designs, electronic design automation (EDA) tools such as HSPICE are widely used to accurately simulate the ICs for performance evaluations. This transistor-level simulator can derive circuit information precisely, enabling designers to predict the functionality, power dissipation, timing, and reliability of their designs. Although this class of transistor-level simulators is powerful for evaluating circuits for various performance metrics, the simulation process is time-consuming. Generally, this would require at least several hours or possibly several days, depending on the simulation settings. To overcome this problem, at the early design stage of numerous ICs, such as phase locked loops (PLLs), frequency synthesizers, and delta-sigma modulators, behavioral models have been built and evaluated using SIMULINK; some simulation precision was sacrificed for the rapid calculation of rough results [1–8]. Researchers developed so-called "behavioral simulation techniques", which trade accuracy for speed; the design procedure can thus be

accelerated by applying these high-speed simulation techniques. Typically, circuit trimming techniques are further applied during the design of physical device models to mitigate the gap between behavioral and transistor-level simulations. However, the required time for the trimming process is unpredictable.

In numerous industrial, home, and office electronic devices, smart temperature sensors (STSs) are increasingly required to monitor and manage temperatures. Due to the demand for small, low-dissipation devices, CMOS STSs are highly competitive and have strong appeal. Several CMOS time-domain STSs (TDSTSs) have been developed over the past 10 years [9–20]. Compared with CMOS voltage-domain STSs, which have highly favorable accuracy for voltage and process variations [21–24], CMOS TDSTSs possess the advantages of lower cost and lower circuit complexity. Thus, several TDSTSs have been reported [10–20]. HSPICE simulations are precise but extremely time-consuming, particularly because STS simulations must simulate multiple temperature points to evaluate performance. Thus, researchers must develop a time-efficient and accurate simulation tool that can effectively derive valid temperature information regarding TDSTSs. However, to the best of our knowledge, no behavioral simulator for TDSTSs meets the standards of simulators for PLLs, frequency synthesizers, or delta-sigma modulators. Beyond behavioral simulation, accurate predictions are also desirable in transistor-level simulations of TDSTS designs. With such predictions, designers can modify a TDSTS at the early design stage without several rounds of time-consuming HSPICE simulations.

The widely used SIMULINK platform provides time-efficient behavioral simulations. Thus, this study proposes a high-speed SIMULINK-based behavioral simulator for TDSTSs. In this paper, we organize simple and accurate equations describing TDSTS into a temperature-dependent model (TDM) and derive temperature-sensing models of a CMOS NOT gate by using HSPICE, which has a very short simulation time. With the TDM and the derived models, the proposed simulator achieved rapid and accurate simulation. The remainder of this paper is arranged as follows: Section 2 introduces the TDM of the TDSTSs, and the building blocks of the proposed SIMULINK simulator are introduced in detail in Section 3. Subsequently, Section 4 presents the experimental results, which validate the proposed technique, and finally, Section 5 concludes this study.

2. TDM for the TDSTSs

In a TDSTS, an inverter-based delay line or oscillator is typically adopted to sense temperature because a simple CMOS NOT gate can act as an effective proportional to absolute temperature (PTAT) sensor that generates a temperature-dependent delay time $t_{NOT}(T)$ [9,12–17], which can be expressed as follows [13,17]:

$$t_{NOT}(T) = \frac{2LC_L T_0^{km}}{\mu_0 W C_{ox} V_{DD}} \times \frac{\ln(3 - 4V_{th}/V_{DD})}{1 - V_{th}/V_{DD}} \times \frac{1}{T^{km}} = \gamma \times T^{-km} \quad (1)$$

where μ_0, T, T_0, V_{th}, W/L, and C_L are the reference carrier mobility, operation temperature, reference temperature, threshold voltage, effective aspect ratio of transistors, and loading capacitance of the NOT gates, respectively. Without considering the effect of voltage variation, the summed parameter γ can be regarded as a process-dependent factor that is nearly independent of temperature. The parameter $-km$ is considered to be temperature independent; this parameter determines the thermal characteristics of the CMOS NOT gate. For example, its value ranges from -1.2 to -2.0 for a 0.35-µm CMOS process and, thus, the $t_{NOT}(T)$ changes linearly with the temperature [17].

Figure 1 shows a block diagram of an all-digital CMOS oscillator-based TDSTS [12]. An oscillator consists of an odd number (k) stages of NOT gates that generate a PTAT period width $t_{OSC}(T)$. To achieve a satisfactory temperature resolution (R), a time amplifier (TA) is used to amplify the $t_{OSC}(T)$ with a time gain n. With a simple XOR gate, a sufficiently wide PTAT pulse $t_P(T)$ is generated. A reference clock with a stable period width (t_{REF}) is used to convert the $t_P(T)$ into a corresponding digital code $N(T)$ by using an AND gate and output counter. The $N(T)$ of the simple sensor can be formulated as follows [13,17]:

$$N(T) = \frac{t_p(T)}{t_{REF}} = \frac{n \times t_{OSC}(T)}{t_{REF}} = \frac{n}{t_{REF}} \times 2 \times k \times \gamma \times T^{-km} \quad (2)$$

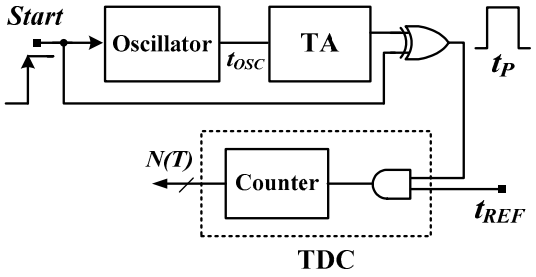

Figure 1. Building block of a simple CMOS TDSTS.

Since (in theory) n, k, and t_{REF} are ideally temperature- and process-insensitive, the thermal characteristics of $N(T)$ are ideally identical to those of $t_{NOT}(T)$. Equation (2) is a simple, but accurate, equation that describes the temperature behavior of the TDSTS, which has been verified in previous studies [12,13,17]. If the models (i.e., $\gamma \times T^{-km}$ for process and temperature variation) of a single CMOS NOT gate can be derived precisely, the sensor code $N(T)$ can be attained by computing Equation (2) such that the performance can be roughly evaluated and the design procedure can be simplified. Various crucial specifications, such as the limits of accuracy and the resolution, can be estimated quickly.

Figure 2 shows a block diagram of an all-digital TDSTS with one-point calibration support that incorporates process-variation calibration [13]. In contrast to the simple TDSTS shown in Figure 1, the adjustable-gain time amplifier (AGTA) shown in Figure 2 is modified from a standard time amplifier to realize the variable time gain n_C for calibration. The calibration circuit is composed of a magnitude comparator and SAR (successive approximation register) control logic; this eliminates the influence of process variation (denoted as i) to support one-point calibration. The calibration technique and detailed procedure were presented in [13]. For one-point calibration, the calibrated code $N_{C,i}(T_C)$ at the calibration temperature T_C can be adjusted to match the calibration value N_S. This calibration condition can be expressed as follows:

$$N_{C,i}(T_C) = \frac{n_{C,i} \times t_{OSC,i}(T_C)}{t_{REF}} = N_S \quad (3)$$

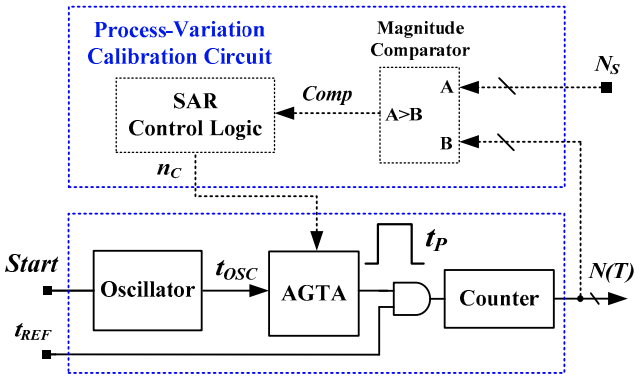

Figure 2. Block diagram of a TDSTS with a process-variation calibration circuit.

The oscillation period width of the ith sensor on T_C ($t_{OSC,i}(T_C)$) is compensated dynamically by adjusting the corresponding $n_{C,i}$. The calibration result is presented as follows:

$$n_{C,i} = \frac{t_{REF} \times N_S}{t_{OSC,i}(T_C)} = \frac{t_{REF} \times N_S}{2 \times k \times \gamma \times T_C^{-km}} \quad (4)$$

Finally, Equation (4) is substituted into Equation (2), and the $N_{C,i}(T)$ after the calibration can be expressed as:

$$N_{C,i}(T) = \frac{n_{C,i} \times 2 \times k \times T^{-km}}{t_{REF}} = \left(\frac{T}{T_C}\right)^{-km} \times N_S \quad (5)$$

In other words, with the calibration, the calibrated time gain $n_{C,i}$ replaces the fixed n in the uncalibrated sensor to generate the $N_{C,i}(T)$ of the calibrated sensor. The result of Equation (5) reveals that the process term γ can be eliminated effectively, enabling the calibrated sensor to support one-point calibration. According to the two chosen values ($N(T_1)$ and $N(T_2)$) and their temperature interval ($T_1 - T_2$), the sensor resolution R can be determined as follows:

$$R = \frac{T_1 - T_2}{N(T_1) - N(T_2)} \quad (6)$$

Furthermore, the conversion time can be estimated as the product of the digital value at the highest temperature and t_{REF} (i.e., $N(T_2) \times t_{REF}$) without considering the delay time of the devices in the sensor and, thus, the ideally lowest conversion rate (CR) is determined as:

$$CR = \frac{1}{N(T_2) \times t_{REF}} \quad (7)$$

The transistor-level of the CMOS TDSTSs can be simulated using HSPICE to derive the uncalibrated codes $N_i(T)$ accurately with process and temperature variations (i.e., Equation (2)). With the calibration, the calibrated $n_{C,i}$ (i.e., Equation (4)) can be simulated to further derive the calibrated codes $N_{C,i}(T)$ (i.e., Equation (5)). The HSPICE simulation is precise but extremely time-consuming. To reduce the simulation time substantially, Equations (1)–(5) were used in the proposed simulator as a TDM for behavioral simulation.

3. Proposed Behavioral Simulator Using SIMULINK

According to Equations (1)–(5) of the TDM, the $t_{NOT}(T)$ is the most crucial item because it is the only quantity related to the temperature and process. This feature can be used to develop a time-efficient behavioral simulator for TDSTSs in order to estimate behavior without using HSPICE, significantly reducing the simulation time. Another vital issue is that the simulator is concise because its equations are simple and linear. The equations have been verified to function effectively [12–17], which is conducive to precise simulation.

A conceptual block diagram of a concise simulation tool for a TDSTS is shown in Figure 3. The designer selects the basic parameters (n, k, t_{REF}, N_C, T_C) and loads the models ($\gamma \times T^{-km}$) from the library for mathematical computation; the system calculates the equations using the models and the input parameters. After the simulation results have been generated, the performance can be evaluated. Moreover, the figures for the results ($N_i(T)$ and $N_{C,i}(T)$) and the processed data (for example, the inaccuracy) can be directly displayed using a suitable program, which simplifies the operations and increases the functionality of the tool.

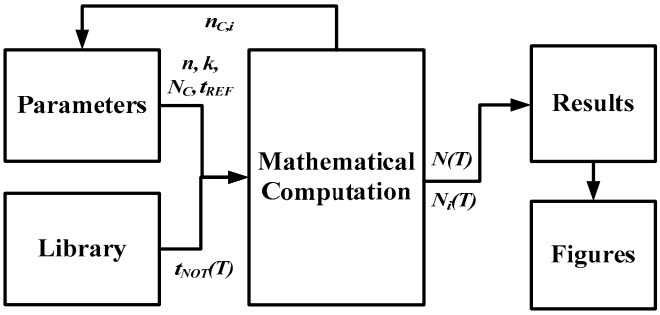

Figure 3. Conceptual block diagram of a concise simulation tool for a TDSTS.

The objective of this study was to develop a concise and low-cost simulator for performing rapid and accurate TDSTS simulations. SIMULINK is a widely used behavioral simulation platform that is highly suitable for implementing the simulator. Implementation in the SIMULINK environment brings numerous advantages such as high construction flexibility, large computation capability, a user-friendly graphical interface, and extremely short computation time. The design presented in Figure 3 can be elaborated with TDSTSs and process-variation calibration in the SIMULINK environment to yield the constructed block of the proposed simulator shown in Figure 4. The block of the TDSTSs comprises a library that stores the temperature-sensing models of a single NOT gate derived from HSPICE simulations, and two sensors (without and with calibration) for estimating system performance. The block of the calibration is used to generate the calibrated time gain $n_{C,i}$ for the sensor with calibration. Finally, the simulation results ($N_i(T)$ and $N_{C,i}(T)$) are stored in .mat files for figure generation with MATLAB. The software structure is concise and the defining equations are simple. Thus, a low-cost simulator for rapid performance evaluations can be constructed.

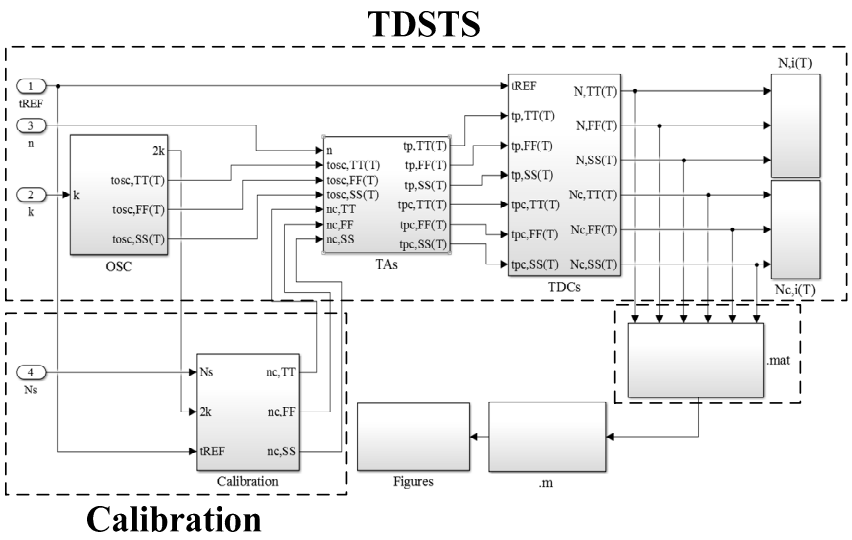

Figure 4. Constructed block of the proposed simulator in the SIMULINK environment.

First, considering the uncalibrated sensor (i.e., the simple sensor without calibration), the input values of the parameters are k for determining the oscillation period width $t_{OSC,i}(T)$ (i denotes the three process corners: typical-typical (TT), fast-fast (FF), and slow-slow (SS)), n for amplifying the width to generate $t_{P,i}(T)$, and t_{REF} for time-to-digital conversion. The results of $N_i(T)$ are obtained using Equation (2) and the models of $t_{NOT,i}(T)$. Second, the process-variation calibration is activated by giving the value of N_S on T_C. Through Equation (3), the corresponding $n_{C,i}$ of the three process corners can be generated for the AGTA in the calibrated sensor. Then, the estimation of the calibrated

sensor is performed. With the same $t_{NOT,i}(T)$, k, and t_{REF}, the computation of Equation (5) generates the results of $N_{C,i}(T)$.

After the results have been generated, the data of $N_i(T)$ and $N_{C,i}(T)$ are stored in a .mat file, which is loaded into a prewritten MATLAB file to estimate the sensor resolution R by using Equation (6), and to generate the figures that display the performance. In other words, the linearity and inaccuracy after data processing can be directly presented in the figures in a simple manner. This simplifies the figure generation procedure considerably.

3.1. Temperature-Sensing Model of a Single CMOS NOT Gate

In the SIMULINK platform, the blocks are described by the equations of the TDM, which express their outputs in terms of the input parameters and NOT gate models. Therefore, the precision of the behavioral simulation depends on how accurately those equations describe the actual condition of each building block and the precision of the captured model. The accurate equations are presented in previous subsections. This subsection introduces the temperature-sensing model of a NOT gate.

To build a library that stores the models for behavioral simulation, an oscillator composed of 21 stages of CMOS NOT gates (PMOS: W/L = 2/2 μm, NMOS: W/L = 1/2 μm) was simulated using HSPICE in a TSMC 0.35-μm CMOS process for process and temperature variations (TT, SS, and FF of the three process corners, ranging from 0 to 80 °C in 10 °C increments). The long MOS channel length was selected to facilitate the realization of a wider oscillation period width. An oscillator with relatively high stages was used to optimize the precision. Furthermore, the models of the single NOT gate were derived by averaging the period width and were stored in the library, as presented in Table 1. This averaging data would be expected to dominate the precision of the proposed simulator. Thus, the numerical precision of the data for the captured models is expected to suffice. It is related to the least significant bit (LSB) time (i.e., t_{REF}) and the value of $2 \times k \times n$. For example, Equation (2) shows that when t_{REF} = 25 ns (i.e., 40 MHz) and $2 \times n \times k$ = 100,000, the precision needs to be in the order of tens of femtoseconds (fs) (i.e., 1/1000 in picoseconds (ps)). This is because when the data are amplified by 100,000, the amplified data of the LSB is near 10 ns at most, which is less than 25 ns (the LSB time). Equivalently, the non-ideal effect for the sensor code is less than 1 LSB and is practically irrelevant to the result. When it is necessary to estimate more results, the data for greater numbers of temperature points in a range wider than the commercial temperature range can be obtained in the same manner.

Table 1. Simulation model (data) with numerical fineness of tens in digit in fs.

Temperature \ Process Corner	TT (ps)	FF (ps)	SS (ps)
0 °C	682.71	615.89	761.48
10 °C	707.32	638.92	790.24
20 °C	732.76	661.34	818.87
30 °C	756.34	682.60	845.57
40 °C	781.72	704.23	872.86
50 °C	805.21	726.93	898.20
60 °C	828.53	747.62	926.09
70 °C	851.50	769.51	951.71
80 °C	875.17	790.70	978.69

3.2. TDSTSs

Figure 5 presents the detailed construction of TDSTSs in the SIMULINK environment for (a) the block of the oscillator including the library; (b) the block of the TAs; and (c) the block of the TDCs. The sensor codes ($N_i(T)$ and $N_{C,i}(T)$) for the two sensors are generated by the two TAs and the corresponding TDCs. Their operation is totally identical except for n and $n_{C,i}$. Although two sensors were designed in this simulator, the total simulation time was not increased considerably because the proposed simulator is concise and performed by using SIMULINK, which was a notable advantage of the SIMULINK platform.

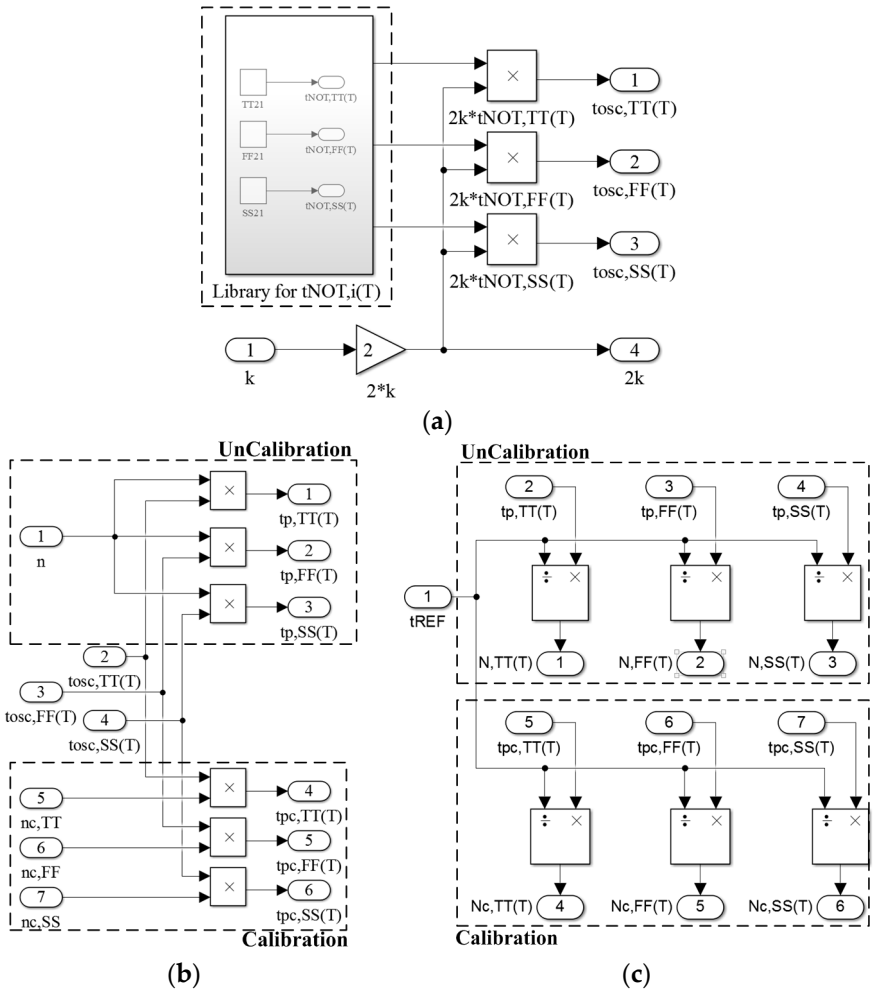

Figure 5. (**a**) Constructed scheme of the oscillator in the SIMULINK environment; (**b**) constructed scheme of the TAs; and (**c**) TDCs in SIMULINK environment.

As shown in Figure 5a, by the product of 2 and k and the captured models in the library, the $t_{OSC,i}(T)$ are generated and used in the two sensors. Simultaneously, the value of $2 \times k$ is sent to the calibration block (next subsection). Next, when the $t_{OSC,i}(T)$ is amplified with n or $n_{C,i}$ generated from the calibration block, the $t_{P,i}(T)$ in the uncalibrated sensor and the $t_{PC,i}(T)$ in the calibrated sensor are produced, as presented in Figure 5b. Figure 5c presents the process generating $N_i(T)$ or $N_{C,i}(T)$ by counting $t_{P,i}(T)$ or $t_{PC,i}(T)$, respectively, by using the t_{REF}. The three steps refer to Equations (2) and (5).

3.3. Process-Variation Calibration for the Three Process Corners

To evaluate the calibrated sensor, the calibrated time gain $n_{C,i}$ ($n_{C,TT}$, $n_{C,FF}$, and $n_{C,SS}$) must be derived for time amplification. The calibration compensates for the process variation of the oscillator on T_C. Thus, only the model on T_C (i.e., $t_{NOT,i}(T_C)$) is used in this scheme. The constructed scheme is illustrated in Figure 6. The value N_S on T_C is selected to perform one-point calibration. The $t_{OSC,i}(T_C)$ are generated using the value of $2 \times k$ (from the block of the oscillator) and the model of $t_{NOT,i}(T_C)$. With the product of N_S and t_{REF}, the corresponding $n_{C,i}$ can be determined using Equation (4) and is used in the calibrated sensor. The higher the value of N_S is, the higher the value of n_C and the sensor resolution are.

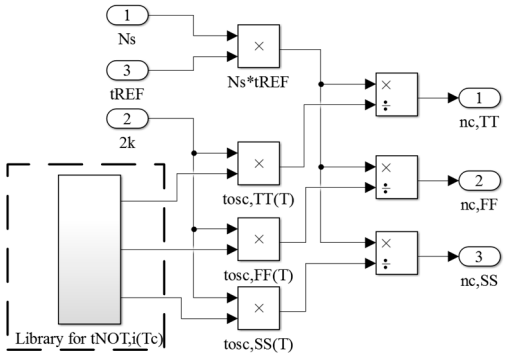

Figure 6. Constructed scheme of the process-variation calibration.

3.4. Figure Generation for Performance Evaluation

Generally, after a simulation has generated results, a suitable program processes the output into a user-friendly format with figures that present the required functions (e.g., the linearity and the inaccuracy). To simplify the procedure, the simulated data of the proposed simulator are stored in .mat format during simulation and are automatically loaded into a prewritten .m file. MATLAB processes the data in the .m file and generates the required figures. For example, the linearity of the simulation results and the corresponding inaccuracy, including two-point calibration for the uncalibrated sensor and one-point calibration for the calibrated sensor, are often presented in figures to illustrate system performance. Since this simulator features an .m file, the figures of the required functions for the two sensors can be simultaneously exhibited in a remarkably simple manner. Designers can easily see whether the presented performance levels satisfy the specifications.

4. Experimental Results

To validate the behavioral simulator, behavioral and transistor-level simulations were performed with the proposed simulator and HSPICE, respectively. A sufficiently wide t_{REF} can effectively reduce the influence of the delay mismatch and delay variation along various signal paths, thereby improving the precision of the simulation. In addition, a low operational frequency can mitigate high power consumption. Thus, the design should specify relatively wide values of t_{REF} and $t_{OSC}(T)$. For simplicity, t_{REF} = 25 ns is given for all situations, and the T_C is set to 40 °C (i.e., the median of 0–80 °C).

In the uncalibrated sensor, the value k = 21 was set to achieve a relatively wide oscillation period width $t_{OSC}(T)$ (for example, 2 × 21 × $t_{NOT,TT}$ (40 °C) = 32.832 ns) for low power consumption. In additional, the fixed time gain n = 5000 was set for a satisfactory R. With the transistor-level simulations using HSPICE, the simulation time for the three process corners in 10 °C steps from 0 °C to 80 °C was around 27 hours to derive the results. By the contrast, under the same simulation setting, the total simulation time of the proposed system was only a few seconds, which saved a notable amount of time. After the proposed simulation generated the outcome of $N_i(T)$, it was stored in $N_i(T)$ .mat; the graphical output is presented in Figure 7. MATLAB loaded the .m file, which contained the data of $N_i(T)$, and produced figures depicting the linearity and the inaccuracies after two-point calibration, as shown in Figure 8. Furthermore, the figure generation procedure was simplified considerably. Simultaneously, the proposed simulator calculated the R values for the three process corners. For example, in TT mode, N(0 °C) = 5735 and N(80 °C) = 7352 were chosen with a temperature interval of 80 °C; the R was calculated as approximately 0.05 °C. Moreover, the estimated conversion time at 80 °C can be calculated as 183.8 µs (7352 × 25 ns) and the lowest CR is determined as 5.44 k samples/s in this TT mode. These operations demonstrated the functionality of the proposed simulator.

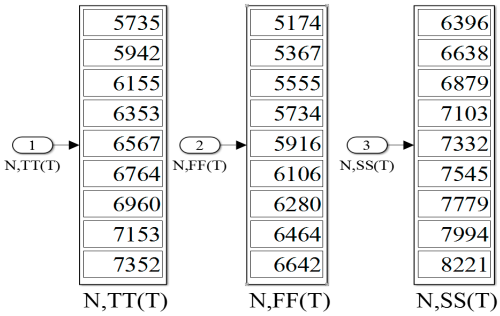

Figure 7. Outcome of $N_i(T)$ in the proposed simulator.

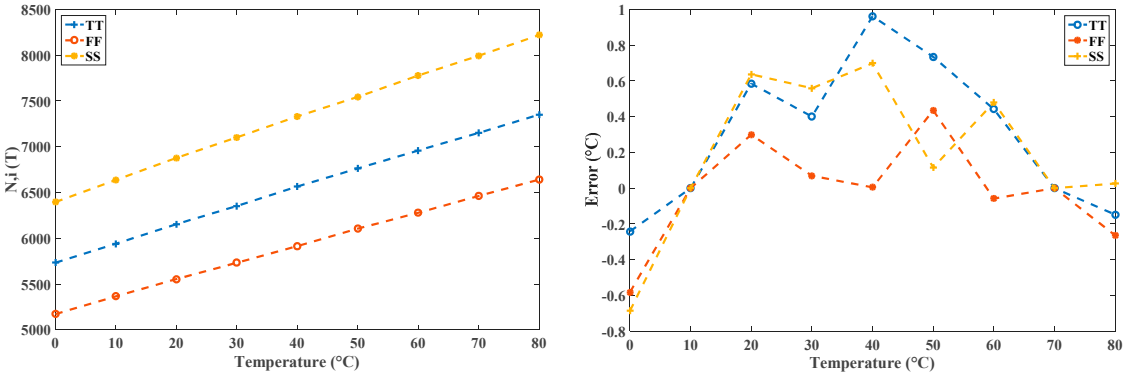

Figure 8. (**a**) Linearity of $N_i(T)$ using the proposed simulator; and (**b**) corresponding inaccuracies after two-point calibration of the uncalibrated sensor.

To further demonstrate the precision of the proposed simulator, the two simulation results $N_i(T)$, and their differences in degrees Celsius, are presented in Figure 9 for comparison. The highly similar conditions between the behavioral and transistor-level simulators are shown. The maximal difference is within −0.8 °C~0.35 °C only, which verifies the functionality of the proposed simulator.

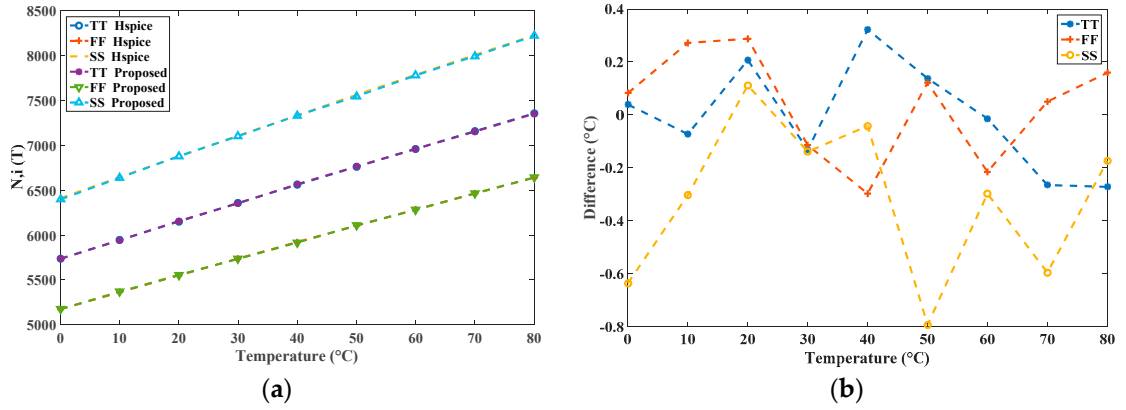

Figure 9. (**a**) Two simulation results $N_i(T)$ from the proposed simulator and HSPICE; and (**b**) differences in degrees Celsius.

To perform the process-variation calibration, the value of N_S = 6500 was set to T_C = 40 °C to realize a calibrated R of approximately 0.05 °C. The same values of k = 21 and t_{REF} = 25 ns were selected and the model $t_{NOT,i}$(40 °C) was used to determine the $n_{C,i}$ for the three process corners. With the same operation as that of the uncalibrated sensor, the calibrated sensor attained the calibrated $N_{C,i}(T)$ in

an extremely short time. The generated figures for the simulated $N_{C,i}(T)$ after the calibration and for the corresponding inaccuracy after one-point calibration are shown in Figure 10. The $N_{C,i}(T)$ values for the three corners nearly coincided and a stable R was achieved, thereby validating the proposed one-point calibration.

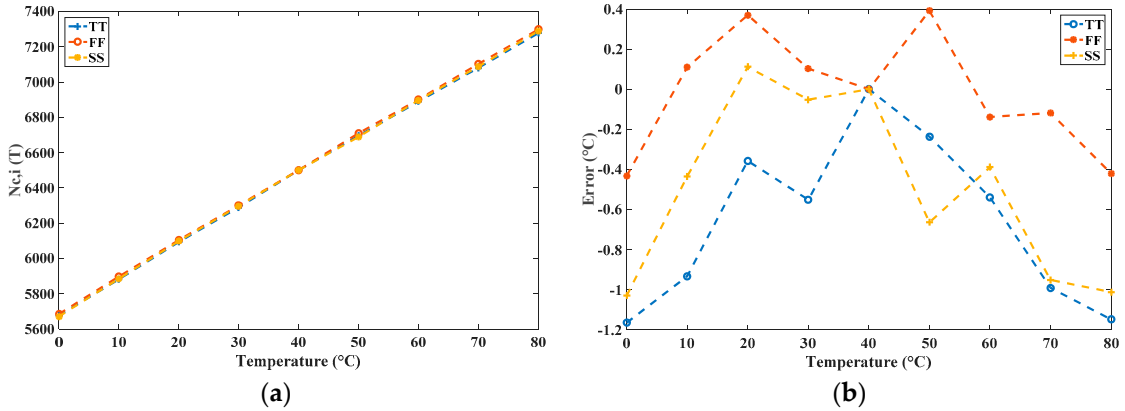

Figure 10. (**a**) $N_{C,i}(T)$ after process-variation calibration by using the proposed simulator; and (**b**) corresponding inaccuracies after one-point calibration for the calibrated sensor.

To verify the precision, a calibration was executed using HSPICE for comparison; this required a substantial amount of simulation time. The HSPICE-simulated $n_{C,i}$ were used as the corresponding time gain values to simulate a transistor-level calibrated sensor, and (predictably) the simulation time was very long. The simulation results $N_{C,i}(T)$ of the two simulators are shown in Figure 11a and the maximal difference is in the range of -0.7 °C to 0.6 °C, as shown in Figure 11b. The highly similar results of these tests validate the calibration of the proposed simulator. To further verify the effect of the voltage variation, we capture the models of the CMOS inverter from 3.1 V to 3.5 V and store them in the library. With the same operation, Figure 12 shows the corresponding differences and presents the similar results, which validate the proposed simulator for the different-voltage operation.

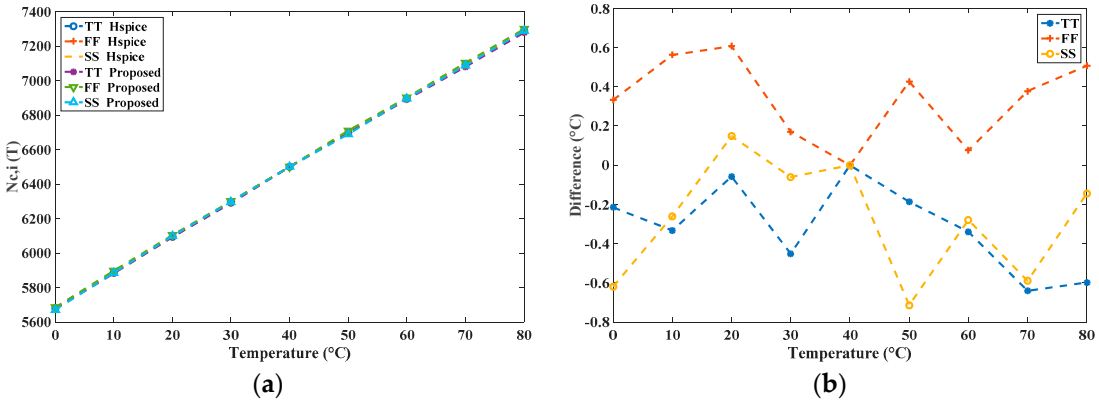

Figure 11. (**a**) Simulation results of the two simulators after calibration; and (**b**) differences in degrees Celsius.

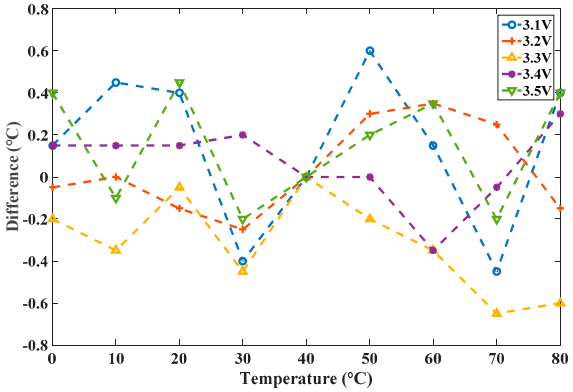

Figure 12. Differences in degrees Celsius for voltage variation.

For precision considerations, the selected value of k also determines the precision of the simulator because of the precision of the captured models. Using the models listed in Table 1, simulations of the calibrated sensor were performed using the proposed system and HSPICE with values of k that varied from 17 to 25 with increments of 2. The corresponding differences in degrees Celsius are presented in Figure 13. The smallest difference was at k = 21; both k = 17 and 25 had larger differences than k = 21 did. The greater the distance between k and 21 was, the greater the difference was. This shows that the captured models for identical stages should be used to achieve the highest possible precision.

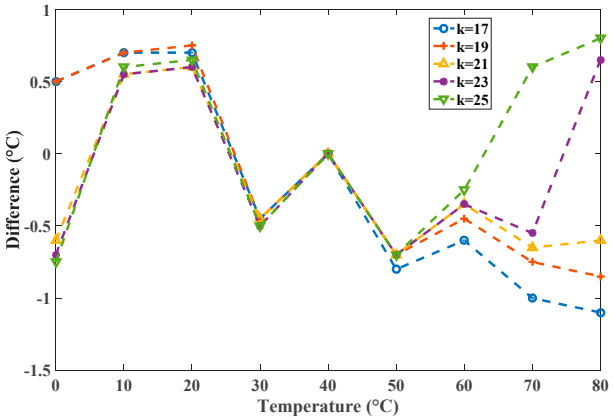

Figure 13. The differences for k = 17–25 with the same model.

To improve the precision, the library was extended with models of the appropriate stages (k = 17–25). For simplicity, Figure 14 shows the extended library with three TT corner models. In addition to the extended models, the control logics for selecting the k values and the corresponding switches were added to output the appropriate model for the oscillator. Additionally, the designer could input $k > 25$ or $k < 17$ and the system would automatically substitute the model for k = 25 or k = 17, respectively. The improvements from this modification are presented in Figure 15. The precision values were obviously enhanced by this modified library. When necessary, the detailed models for every value of k can be constructed to achieve high precision by increasing the library cost. The experiments show that the proposed simulator for all-digital CMOS TDSTSs can perform rapid simulations with accurate results.

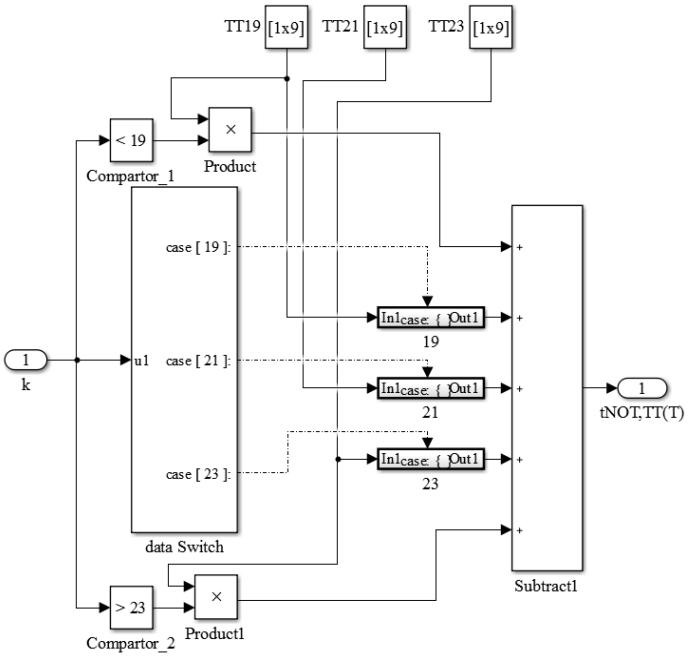

Figure 14. Construction scheme of the modified library with three models.

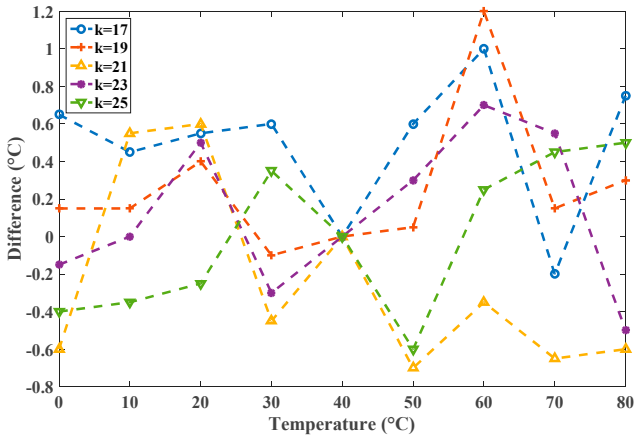

Figure 15. The differences for k = 17–25 with the corresponding model.

5. Conclusions

An accurate behavioral simulator of an all-digital CMOS TDSTS based on SIMULINK is presented in this paper. A behavioral simulation tool was developed as a rapid performance evaluator that provides the benefits of a user-friendly graphical interface, high flexibility for circuit extension, and substantial data processing capabilities. With the TDM and the captured models of the unit device, the proposed simulator derives accurate results rapidly. The high precision of the simulator was verified through a transistor-level simulation by using HSPICE. To our knowledge, this is the first behavioral simulator for TDSTSs.

Acknowledgments: This study was financially supported by Ministry of Science and Technology (MOST) of Taiwan for Grant MOST 104-2221-E-327-031. The authors would like to express their deep appreciation to National Chip Implementation Center (CIC) of Taiwan for supporting EDA tools.

Author Contributions: C.-C.C. proposed the idea, designed the simulator, and wrote the manuscript. C.-L.C. helped with the simulator design and the manuscript writing. Y.-T.L. implemented the simulator and performed the simulations.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Dias, V.F.; Liborali, V.; Maloborti, F. Design tools for oversampling data converters: needs and solutions. *Microelectron. J.* **1992**, *23*, 641–650. [CrossRef]
2. Dessaint, L.-A.; Kamal, A.-H.; Hoang, L.-H.; Sybille, G.; Brunelle, P. A Power System Simulation Tool Based on SIMULINK. *IEEE Trans. Ind. Electron.* **1999**, *46*, 1252–1254. [CrossRef]
3. Francken, K.; Gielen, G.E. A high-level simulation and synthesis environment for delta-sigma modulators. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **2003**, *22*, 1049–1061. [CrossRef]
4. Malcovati, P.; Brigati, S.; Francesconi, F.; Maloberti, F.; Cusinato, P.; Baschirotto, A. Behavioral modeling of switched-capacitor sigma-delta modulators. *IEEE Trans. Circuits Syst. I* **2003**, *50*, 352–364. [CrossRef]
5. Reina, J.M.; Rosa, J.D.; Medeiro, F.; Romay, R.; Rio, R.; Verdu, B.P.; Vazquez, A.R. A SIMULINK-based approach for fast and precise simulation of switched-capacitor, switched current and continuous-time ΣΔ modulator. In Proceedings of the International Symposium on Circuits and Systems, Bangkok, Thailand, 25–28 May 2003; Volume 4, pp. 620–623.
6. Hashem, Z.-H.; Kale, I.; Shoaei, O. Modeling of Switched-Capacitor Delta-Sigma Modulators in SIMULINK. *IEEE Trans. Instrum. Meas.* **2005**, *54*, 1646–1654.
7. Ruiz-Amaya, J.; Rosa, J.; Fernández, F.V.; Medeiro, V.F.; Río, R.; Pérez-Verdú, B.; Rodríguez-Vázquez, A. High-Level Synthesis of Switched-Capacitor, Switched-Current and Continuous-Time Modulators Using SIMULINK-Based Time-Domain Behavioral Models. *IEEE Trans. Circuits Syst. I* **2005**, *52*, 1795–1810. [CrossRef]
8. Alecsa, B.; Cirstea, M.N.; Onea, A. SIMULINK link Modeling and Design of an Efficient Hardware-Constrained FPGA-Based PMSM Speed Controller. *IEEE Trans. Ind. Inform.* **2012**, *8*, 554–562. [CrossRef]
9. Chen, P.; Chen, C.-C.; Tsai, C.-C.; Lu, W.-F. A Time-to-Digital-Converter-Based CMOS Smart Temperature Sensor. *IEEE J. Solid-State Circuits* **2005**, *40*, 1642–1648. [CrossRef]
10. Lin, Y.-S.; Sylvester, D.; Blaauw, D. An Ultra Low Power 1 V, 220 nW Temperature Sensor for Passive Wireless Applications. In Proceedings of the IEEE Custom Integrated Circuits Conference, San Jose, CA, USA, 21–24 September 2008; pp. 507–510.
11. Law, M.K.; Bermak, A. A 405-nW CMOS temperature sensor based on linear MOS operation. *IEEE Trans. Circuits Syst. II* **2009**, *56*, 891–895. [CrossRef]
12. Chen, P.; Shie, M.-C.; Zheng, Z.-Y.; Zheng, Z.-F.; Chu, C.-Y. A fully Digital Time-Domain Smart Temperature Sensor Realized with 140 FPGA Logic Elements. *IEEE Trans. Circuits Syst. I* **2007**, *54*, 2661–2668. [CrossRef]
13. Chen, P.; Chen, S.-C.; Shen, Y.-S.; Peng, Y.-J. All-Digital Time-Domain Smart Temperature Sensor with an Inter-Batch Inaccuracy of 0.7~0.6 °C after One-Point Calibration. *IEEE Trans. Circuits Syst. I* **2011**, *58*, 913–920. [CrossRef]
14. Chung, C.-C.; Yang, C.-R. An Autocalibrated all-digital temperature sensor for on-chip thermal monitoring. *IEEE Trans. Circuits Syst. II* **2011**, *58*, 105–109. [CrossRef]
15. Kim, K.; Lee, H.; Kim, C. 366-Ks/s 1.09-nJ 0.0013-mm2 Frequency-to-Digital Converter Based CMOS Temperature Sensor Utilizing multiphase clock. *IEEE Trans. VLSI* **2013**, *21*, 1950–1954. [CrossRef]
16. Hwang, S.; Koo, J.; Kim, K.; Lee, H.; Kim, C. A 0.008 mm2 500 μW 469 kS/s frequency-to-digital converter based CMOS temperature sensor with process variation compensation. *IEEE Trans. Circuits Syst. I* **2013**, *60*, 2241–2248. [CrossRef]
17. Chen, C.-C.; Chen, H.-W. A Linearization Time-Domain CMOS Smart Temperature Sensor Using a Curvature Compensation Oscillator. *Sensors* **2013**, *13*, 11439–11452. [CrossRef] [PubMed]
18. An, Y.-J.; Ryu, K.; Jung, D.-H.; Jung, S.-O. An Energy Efficient Time-Domain Temperature Sensor for Low-Power on-Chip Thermal Management. *IEEE Sens. J.* **2014**, *14*, 104–110. [CrossRef]
19. Tang, X.; Ng, W.T.; Pun, K.-P. A Resistor-Based Sub-1-V CMOS Smart Temperature Sensor for VLSI Thermal Management. *IEEE Trans. VLSI* **2015**, *23*, 1651–1660. [CrossRef]
20. Hung, C.-C.; Chu, H.-C. A Current-Mode Dual-Slope CMOS Temperature Sensor. *IEEE Sens. J.* **2016**, *16*, 1898–1907. [CrossRef]

21. Duarte, D.-E.; Geannopoulos, G.; Mughal, U.; Taylor, G. Temperature Sensor Design in a High Volume Manufacturing 65 nm CMOS Digital Process. In Proceedings of the 2007 IEEE Custom Integrated Circuits Conference, San Jose, CA, USA, 16–19 September 2007; pp. 221–224.
22. Lakdawala, H.; Li, Y.W.; Raychowdhury, A.; Taylor, G.; Soumyanath, K. A 1.05 V 1.6 mW 0.45 °C 3σ resolution ΣΔ based temperature sensor with parasitic-resistance compensation in 32 nm digital CMOS process. *IEEE J. Solid-State Circuits* **2009**, *44*, 3621–3630. [CrossRef]
23. Shor, J.-S.; Luria, K. Miniaturized BJT-based thermal sensor for microprocessors in 32- and 22-nm Technologies. *IEEE J. Solid-State Circuits* **2013**, *48*, 2860–2867. [CrossRef]
24. Deng, C.; Sheng, Y.; Wang, S.; Hu, W.; Diao, S.; Qian, D. A CMOS Smart Temperature Sensor with Single-Point Calibration Method for Clinical Use. *IEEE Trans. Circuits Syst. II* **2016**, *63*, 136–140. [CrossRef]