Adaptive Circuit Design Methodology and Test Applied to Millimeter-Wave Circuits

J.-O. Plouchart, Benjamin Parker, Bodhisatwa Sadhu, Alberto Valdes-Garcia, and Daniel Friedman
IBM T. J. Watson Research Center

Fa Wang and Xin Li
Carnegie Mellon University

Mihai Sanduleanu
Masdar Institute of Science and Technology

Andreea Balteanu
University of Toronto

Editor's Notes:
This paper presents a design methodology of millimeter-wave circuits that are insensitive to process, voltage, and temperature variations. Instead of using conventional direct sensing, the authors propose an indirect sensing method with Bayesian fusion, which simplifies the sensors and allows more adaptive circuit loops to be integrated.
—Deukhyoun Heo, Washington State University

■ **Millimeter-wave (mm-wave) transceivers** integrated in advanced nanometer CMOS are difficult to design using the traditional six sigma circuit design methodology, in part because of their sensitivities to process (P), supply voltage (V), temperature (T), and also to model unknowns, thus motivating new research work focused on developing self-healing circuit capabilities [1], [2]. One of the primary goals of self-healing has been to use on-chip sensors, actuators, and algorithms to increase high-frequency TRx yield. This objective can be formalized using the following inequality:

$$M_{\text{spec},i}(P, V, T, \text{Act}) \geq T_{\text{spec},i}, \quad \text{for all } i = 1 \text{ to } n \tag{1}$$

where $M_{\text{spec},i}$ is the measured specification i, and $T_{\text{spec},i}$ is the target specification i for a set of n specifications.

The measured specifications are functions of P, V, T, and actuators (Act). The circuit specifications are typically derived from system analysis and hardware measurements. By meeting or exceeding the set of TRx circuit specifications, it is possible to achieve a bit error rate target for a given communication channel. Therefore, the target expression can be reformulated as a single bit-error rate inequality

$$\text{BER}(P, V, T, \text{Chan}, \text{Act}) \leq T_{\text{BER}} \tag{2}$$

where T_{BER} is the targeted bit error rate for the communication link. Based on circuit sensor information and driven by a resident algorithm, actuators

Digital Object Identifier 10.1109/MDAT.2014.2343192
Date of publication: 25 July 2014; date of current version: 04 December 2014.

Copublished by the IEEE CEDA, IEEE CASS, IEEE SSCS, and TTTC

will be set such that adaptive circuits will use the minimum power to transmit data at a targeted bit error rate. As shown by the close relationship between (1) and (2), self-healing design techniques focused on yield maximization may also enable data communication energy minimization. Note that the bit error rate target can be dependent on the type of data sent. For example, the bit error rate can be relaxed for voice or picture, and be more stringent for text or code. Even though the inequality appears to be simple, it in fact represents a complex nonlinear optimization problem. Another level of complexity is that for wireless communication the channel can change rapidly with time.

Several initial adaptive radio-frequency (RF) circuits have already been reported [3]–[8]. In [3], an automatic amplitude control circuit was proposed and its feedback ratio was optimized for noise minimization. In [4], a self-calibrated frequency-domain on-chip phase-noise measurement circuit is designed to accurately measure phase noise. In [5], a cost-effective alternative-test-based performance calibration method is introduced, where a number of performance models are created by considering process and knob variations separately. These models are then used to adaptively configure the knobs and improve the yield during production test. In [6], an orthogonally tunable low-noise amplifier (LNA) is designed, where the performance specifications (i.e., gain and linearity) can be independently controlled to explore tradeoffs between circuit performance and power consumption. In [7]–[9], an indirect performance sensing method is developed to adaptively tune a voltage-controlled oscillator (VCO) to improve its performance and/or yield. The key idea of indirect performance sensing is to use a set of easily measurable metrics [e.g., direct current (dc) bias current] to predict other performance metrics of interest that are difficult to measure by on-chip sensors.

In this paper, we describe a design and test methodology which adaptively configures mm-wave circuits for performance and/or yield enhancement by on-chip self-healing. In the Design and Test Methodology for Adaptive Circuits section, we will review a novel approach, indirect performance sensing based upon Bayesian model fusion (BMF) that can effectively enable sophisticated mm-wave performance prediction without demanding integrated mm-wave circuit measurement infrastructure. In the System-on-Chip Architecture for Adaptive Circuits section, we present an adaptive system-on-chip architecture that allows the efficient integration of a large number of sensors, actuators, and adaptive control loops. Also, in this section, we discuss how to practically integrate hardware, software, and test. Finally, in the Case Study: An Adaptive Millimeter-Wave LNA section, we will show a design example of an adaptive mm-wave LNA.

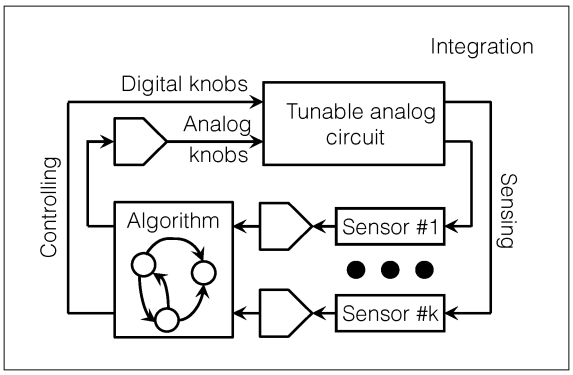

Figure 1. Design of an adaptive circuit must take into account three critical components: sensing, controlling, and integration.

Design and test methodology for adaptive circuits

Designing an effective adaptive circuit is not a trivial task. The major challenge stems from the requirement that the circuit must adaptively respond to changes in process and environmental conditions and automatically correct its behavior. In general, three critical components must be carefully considered for adaptive circuit design, as shown in Figure 1.

- Sensing: The behaviors and performance variables of an adaptive circuit must be accurately monitored by on-chip sensors. The power consumption and silicon area of these on-chip sensors must be minimized so that the overhead of an adaptive circuit is sufficiently small.
- Controlling: Based on the information collected by on-chip sensors, an efficient algorithm must be implemented to optimally control the tuning knobs (e.g., bias current, capacitor array, etc.) to maintain the desired circuit behavior.
- Integration: All on-chip sensors and tuning knobs must be seamlessly integrated with other core

circuit blocks with small silicon area and low power consumption, while quickly and automatically accommodating environmental changes.

Among these three components, on-chip performance sensing is the most challenging, because many analog performance metrics [e.g., noise factor (NF)] cannot be easily measured by on-chip sensors. To address this issue, we propose the idea of indirect performance sensing [9] where the performance of interest (PoI) is not directly measured by an on-chip sensor. Instead, it is accurately predicted from a set of other performance metrics, referred to as the performances of measurement (PoM), which are highly correlated with PoI and are easy to measure. In the following, we will discuss the algorithms and methodologies of the proposed indirect performance sensing in detail.

On-chip indirect performance sensing

Without loss of generality, we denote a PoI as f and the PoM as

$$\mathbf{x} = [x_1 \; x_2 \; \cdots \; x_M]^T \tag{3}$$

where M stands for the number of performance metrics belonging to the PoM. The objective of indirect performance sensing is to build a mathematical model $f(\mathbf{x})$ to accurately estimate the PoI f from the PoM $\mathbf{x}$.

Generating an indirect sensor model $f(\mathbf{x})$ typically consists of three major steps:

- presilicon feature extraction aims to identify a set of important performance metrics (i.e., the PoM) that are highly correlated with the PoI and can be easily measured by on-chip sensors;
- presilicon model training aims to approximate the indirect sensor model $f(\mathbf{x})$ by linear regression based on presilicon simulation data;
- postsilicon model calibration aims to calibrate the indirect sensor model $f(\mathbf{x})$ based on postsilicon measurement data so that the errors posed by presilicon simulation can be appropriately corrected.

To facilitate efficient generation of indirect sensor models, advanced statistical algorithms must be developed to keep the modeling cost affordable. In other words, since both presilicon simulation and postsilicon measurement can be expensive, the aforementioned three steps for indirect sensor modeling must be accomplished with limited simulation and measurement data. As such, the overhead of indirect performance sensing and, eventually, the overhead of adaptive analog circuit can be minimized.

Presilicon indirect sensor modeling

As previously mentioned, the objective of presilicon modeling is to identify a set of important performance metrics as the PoM $\mathbf{x}$ (i.e., feature selection) and then fit the mathematical model $f(\mathbf{x})$ (i.e., model training). To achieve this goal, we propose to adopt the sparse regression (SR) technique [10] to seamlessly integrate the feature selection step and the modeling training step into a unified framework.

In general, a mathematical function $f(\mathbf{x})$ can be approximated as the linear combination of a set of basis functions

$$f(\mathbf{x}) = \sum_{k=1}^{K} \alpha_k \cdot g_k(\mathbf{x}) \tag{4}$$

where $\{g_k(\mathbf{x}); k = 1, 2, \ldots, K\}$ contains the basis functions (e.g., linear and quadratic polynomials), $\{\alpha_k; k = 1, 2, \ldots, K\}$ contains the model coefficients, and K is the total number of basis functions. To intuitively illustrate the basic idea of SR, we consider the following simple case where the basis functions are linear:

$$f(\mathbf{x}) = \sum_{m=1}^{M} \alpha_m \cdot x_m + \alpha_0 \tag{5}$$

where $\{\alpha_m; m = 1, 2, \ldots, M\}$ are the linear model coefficients and α_0 is the constant term.

Since we do not know the PoM $\mathbf{x}$ (i.e., the important performance metrics that should be measured by on-chip sensors) in advance, our proposed strategy is to take a large number of possible candidates and apply SR to fit the model coefficients $\{\alpha_m; m = 0, 1, \ldots, M\}$ as in (5), such that only a small number of these coefficients are nonzero and all other coefficients are zero. Hence, based on the SR results, we can find the set of important performance metrics $\{x_m; \alpha_m \neq 0\}$ as the features of interest. In addition, the model coefficients associated with the selected PoM are simultaneously determined during the aforementioned SR procedure.

Unlike the conventional least squares fitting that solves an overdetermined linear equation by minimizing the mean squared error, SR formulates a convex optimization problem with a unique constraint (e.g., based on $L1$-norm regularization) that is able to promote sparse model coefficients. Such a convex optimization can be solved both efficiently (i.e., with low computational cost) and robustly (i.e., with guaranteed global convergence). More details regarding the SR algorithm can be found in [10].

The aforementioned SR method can be efficiently applied to presilicon feature selection and model training. However, the device and parasitics models used for presilicon simulation are not perfectly accurate and may differ from the postsilicon measurement results. For this reason, there is a strong need to further calibrate the proposed indirect sensor models based on postsilicon measurement data, as will be discussed in the next section.

Postsilicon indirect sensor calibration

The objective of postsilicon sensor calibration is to further correct the modeling error arising from presilicon simulation inaccuracies and also to accommodate the process shift associated with manufacturing lines. One straightforward approach for sensor calibration is to collect a large amount of postsilicon measurement data and then completely refit the indirect sensor model. Such a simple approach, however, can be practically unaffordable, since postsilicon testing is time consuming and, hence, it is overly expensive to collect a large set of postsilicon measurement data.

To address this cost issue, we propose a novel statistical framework, referred to as BMF [11], for efficient postsilicon sensor calibration. BMF relies on an important observation, namely, that even though the simulation and/or measurement data collected at multiple stages (e.g., presilicon versus postsilicon) are not exactly identical, they are expected to be strongly correlated. Hence, it is possible to borrow the data from an early stage (e.g., presilicon) for sensor calibration at a late stage (e.g., postsilicon). In this context, even a small number of postsilicon measurements can be leveraged effectively to enhance model accuracy, allowing the cost of sensor calibration to be substantially reduced.

More specifically, our indirect sensor models are initially fitted by using the early-stage (e.g., presilicon) data. Next, a model template is statistically extracted and encoded as our prior knowledge based on the early-stage model. Finally, the model template is further calibrated by applying Bayesian inference to very few late-stage (e.g., postsilicon) measurements to accurately update these sensor models. Here, by "fusing" the early-stage and late-stage sensor models through Bayesian inference, the amount of required measurement data (hence, the measurement cost) can be substantially reduced.

To fully understand the proposed BMF method, we need to first explain the common characteristics between the early-stage and late-stage sensor models. To this end, we represent both the early-stage model $f_E(\mathbf{x})$ and the late-stage model $f_L(\mathbf{x})$ as the linear combinations of K basis functions, similar to (4)

$$f_E(\mathbf{x}) \approx \sum_{k=1}^{K} \alpha_{E,k} \cdot g_k(\mathbf{x}) \tag{6}$$

$$f_L(\mathbf{x}) \approx \sum_{k=1}^{K} \alpha_{L,k} \cdot g_k(\mathbf{x}) \tag{7}$$

where $\{\alpha_{E,k}; k = 1, 2, \ldots, K\}$ and $\{\alpha_{L,k}; k = 1, 2, \ldots, K\}$ denote the early-stage and late-stage model coefficients, respectively. Since both $f_E(\mathbf{x})$ and $f_L(\mathbf{x})$ model the same PoI of the same circuit, we expect that the model coefficients $\alpha_{E,k}$ and $\alpha_{L,k}$ are similar, but not exactly identical. Such prior knowledge can be statistically encoded as a prior distribution. Next, we apply Bayesian inference to statistically combine the prior distribution and very few late-stage measurements to accurately estimate the late-stage model coefficients $\{\alpha_{L,k}; k = 1, 2, \ldots, K\}$ based on the posterior distribution. Such an approach is referred to as maximum *a posteriori* (MAP) estimation in the statistics community. Additional mathematical details regarding BMF can be found in [11].

In summary, SR and BMF are two core techniques that facilitate our proposed on-chip indirect performance sensing for adaptive analog circuits. In the Case Study: An Adaptive Millimeter-Wave LNA section, we will describe a mm-wave circuit example where indirect performance sensing has been successfully applied to enable adaptive circuit operations.

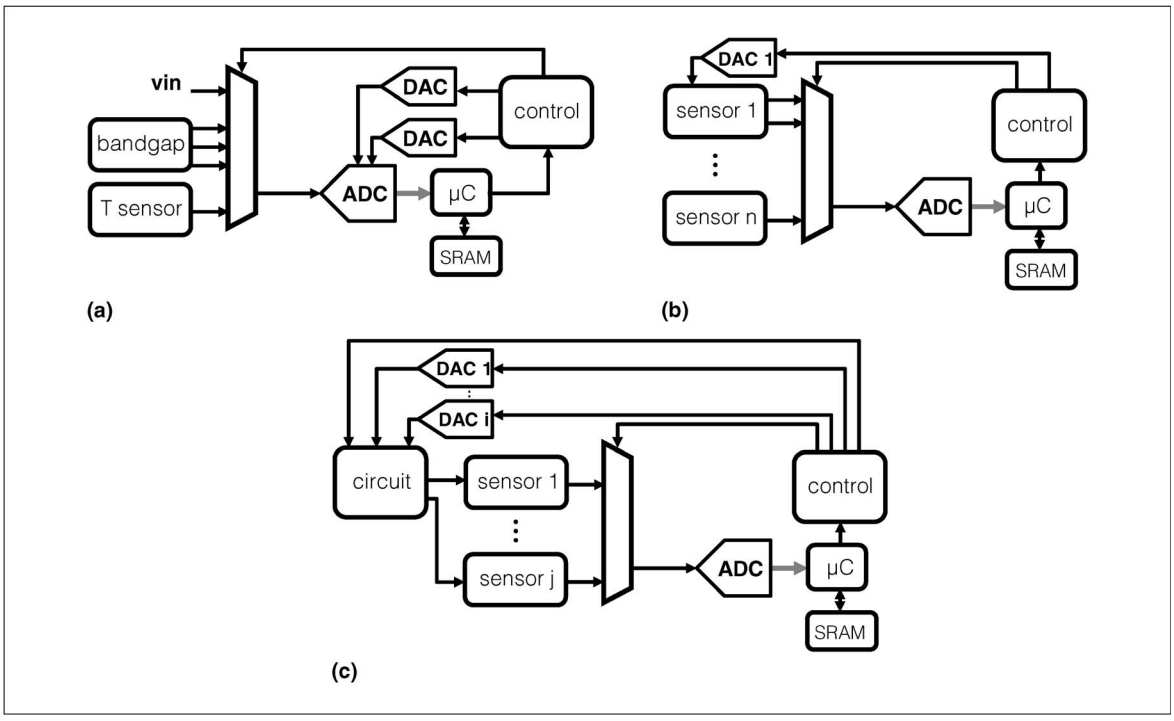

Figure 2. Healing and adaptive control loops. (a) ADC control loop. (b) Sensor control loop. (c) Adaptive circuit control loop.

System-on-chip architecture for adaptive circuits

Adaptive SoC architecture

The implementation of on-chip adaptive algorithms requires complex SoC design, especially for mm-wave TRx. Figure 2 shows the proposed SoC architecture intended to enable circuits to be adaptive. In this architecture, the serial interface and microcontroller data flows are multiplexed such that the algorithm can be run either outside the chip on a host or inside the chip on the microcontroller. A microcontroller was chosen to run most of the algorithms, because it offers much more flexibility (as compared to direct algorithm implementation in hardware) to enhance the algorithm once the chip is tested or in the field. Each circuit is connected through a data bus that allows writing and reading registers. The data bus includes a control scheme that allows writing to a single circuit or multiple circuits (broadcast mode). For added flexibility, a control circuit can be used to control multiple circuits. The enablement of series and parallel register writing and series register reading allows the proper synchronization and control of multiple tasks running in parallel. A synthesizer is required to generate the multiple clocks used in the different circuits and the clock used for the microcontroller. Because multiple clock domains coexist inside the SoC, careful task orchestration and synchronization is required.

ADC control loop

A large number of control loops may be required to enable a complete adaptive TRx. Since most sensor outputs are analog and must be digitized to enable algorithmic control, the first loops that must be established at startup are those for the analog-to-digital converters (ADCs). As shown in Figure 2a, the ADC can be, for example, calibrated by applying one or several bandgap voltages to its inputs. Different analog-to-digital conversion modes can be supported by selecting different digital-to-analog converter (DAC) settings, thus enabling ADC range programmability. In operation, it is desirable for the self-healing control system to periodically check the temperature sensor output, using this result to evaluate temperature drift and then decide whether a new calibration is required.

Sensor control loop

Once the self-healing control system has completed the ADC calibration, it can start to calibrate

the sensors that are sensitive to process, supply voltage, and temperature variations. An implementation example is shown in Figure 2b. Since the ADC circuit area is typically much larger than the sensor area, it is advantageous to group local sensors and select the sensor that needs to be read with an analog multiplexer (Figure 2b). Of course, sensors that are far away from a given ADC might require the instantiation of an additional ADC, since the sensitive analog voltage might be contaminated, for example, by digital or clock circuit noise. The sensor calibration algorithm can take into account not only the sensor output, but also that of other internal sensor nodes (Figure 2b), as well as temperature. The algorithm integrates all the different inputs and adjusts the DAC setting to adjust and calibrate the sensor. An algorithm methodology (using direct and indirect measurements) similar to the one described in the Design and Test Methodology for Adaptive Circuits section can be used to calibrate the sensor across P, V, and T.

Adaptive circuit control loop

Once the self-healing control system has completed the sensor calibrations, it can start to run the adaptive circuit algorithms. As shown in Figure 2c, the analog information coming from j sensors is digitized one by one. The digital information is used by the adaptive algorithms to set the circuit control bits and settings for the current control DACs. The DACs can be implemented using many different architectures, including capacitor, resistor, or field-effect transistor (FET) bank switching. The algorithms can take several iterations to converge to the optimum circuit setting. Each circuit can be set to target certain specifications across P, V, and T using the methodology discussed in the Design and Test Methodology for Adaptive Circuits section. We are also envisioning running adaptive algorithms for several circuits at once, which could allow further reduction in power consumption. Since the software algorithm latency is in the order of a microsecond to a millisecond, the fast adaptive algorithm must be implemented in hardware. Therefore, hardware and software adaptive algorithms will coexist on the same SoC.

Software and hardware codesign methodology

Another important aspect of the self-healing approach is its software and hardware codesign. As shown in Figure 3a, the software is typically developed after the hardware is fabricated and is under test. This methodology has two drawbacks: first, there is no development during the fabrication; and second, it is not possible to coverify the software with the hardware, which can lead to some implementation issues that can only be discovered when the SoC is tested.

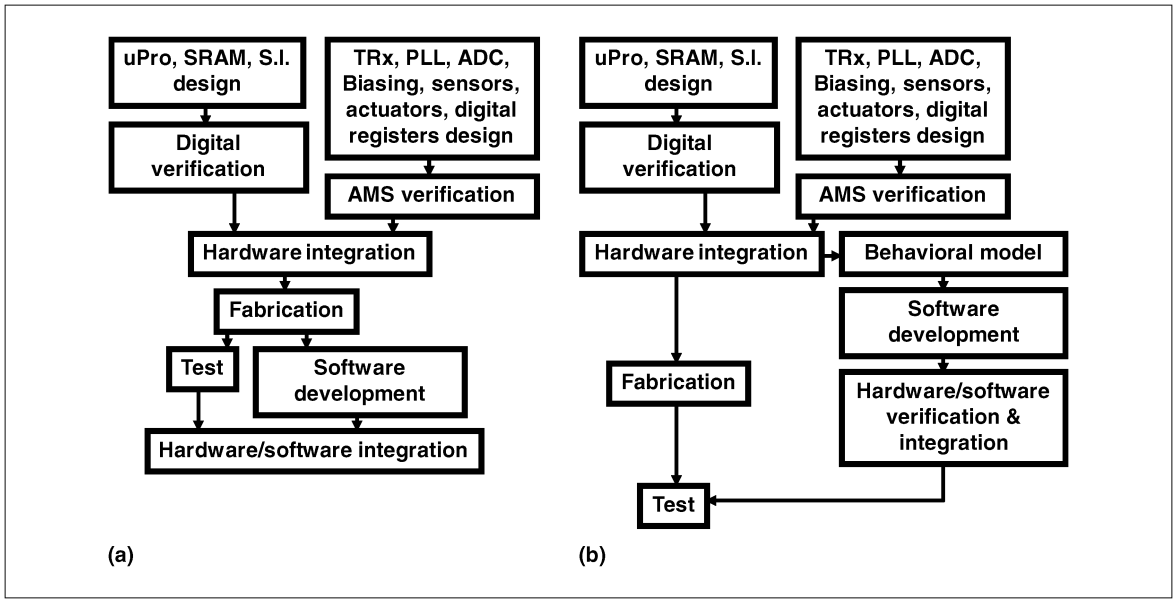

Figure 3. (a) Typical SoC design, verification, and integration flow. (b) Proposed verification and software and hardware codesign flow using high-level behavioral models.

In order to efficiently cosimulate the hardware and the software, we have developed high-level very high speed integrated circuit (VHSIC) hardware description language (VHDL) mathematical models for the mm-wave, RF, analog, and mixed-signal macros used in our designs. The advantage of using VHDL is that continuous analog signals can be applied to the circuits and can be processed by the behavioral models. Thus, analog and digital signals can be computed efficiently by the VHDL simulator. The other means of acceleration exploited in our work is that the software is initially run externally, with sense data and control signals applied to the mm-wave circuit through its serial interface. The software is written in tool command language (TCL), which is also the language used to drive the VHDL simulations. By using this methodology (refer to Figure 3b), we can avoid time-consuming, resource-intensive emulation of the microcontroller, instead natively running the TCL code on the simulation computer, resulting in a speedup of several orders of magnitude. Another important advantage of using TCL is that the testing code used to test the hardware can be the same as that used to verify the chip using the simulator, allowing unification of testing and verification flows.

Case study: An adaptive millimeter-wave LNA

Performance variability of LNA

The effects of process variations have become a challenging issue for RF LNA design [5], [6], and are even more significant at mm-wave frequencies [1], [2]. In particular, the gain, noise factor (F), and matching of the LNA are susceptible to process variations. Figure 4a and b shows the design schematic and layout of a 60-GHz LNA designed in a 32-nm

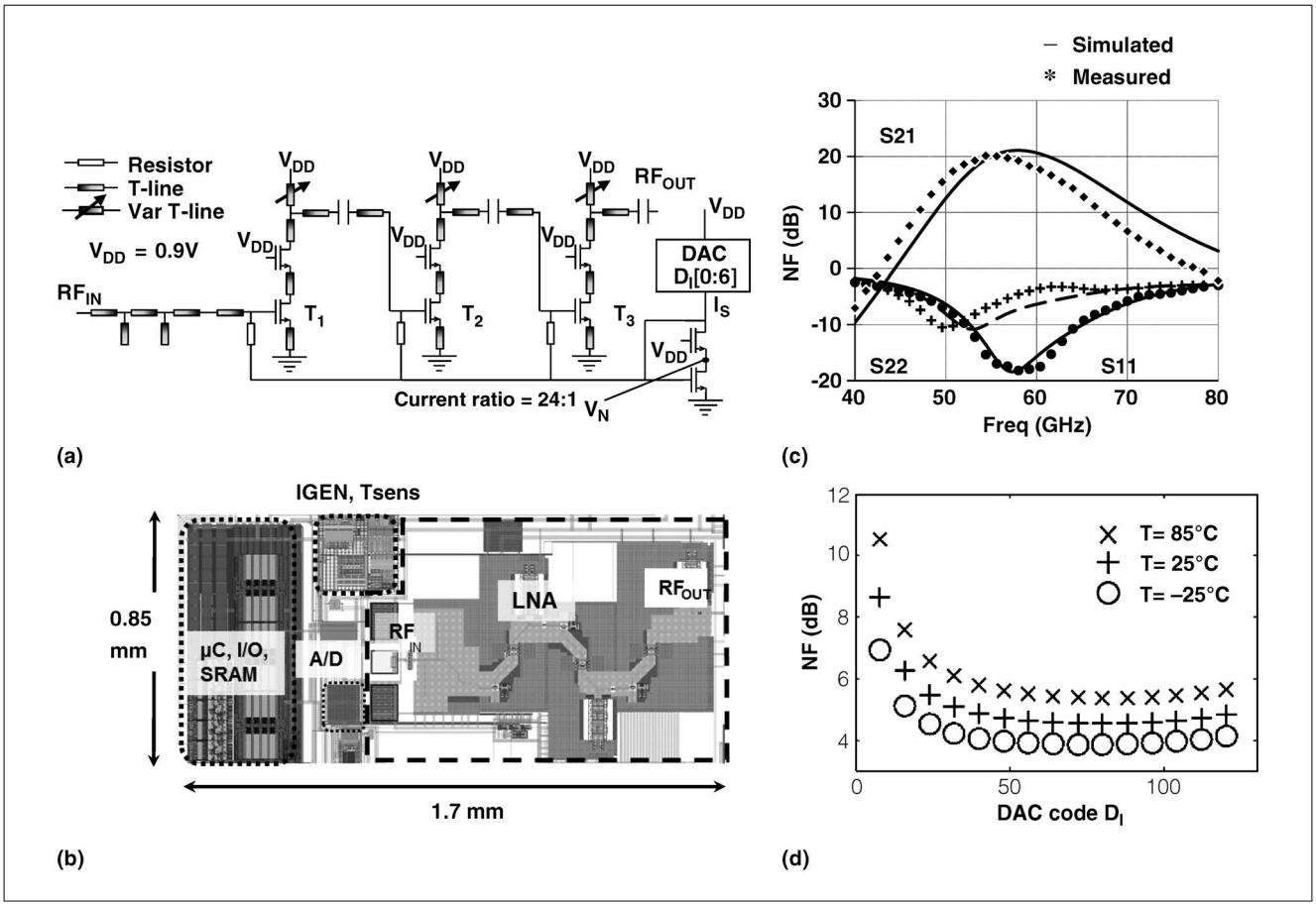

Figure 4. The 32-nm SOI three-stage 60-GHz LNA. (a) Schematic. (b) Physical design. (c) Simulated and measured *S*-parameter comparison. (d) Simulated 60-GHz NF versus current DAC code D_I at $T = 85\,^{\circ}C$, $25\,^{\circ}C$, and $-25\,^{\circ}C$.

silicon on insulator (SOI) process. The FETs, along with all the wire parasitics, were extracted from the layout to enable more accurate simulation. The circuit was simulated using high-frequency models for the transmission lines, capacitors, and resistors used in the design. In Figure 4c, we show the LNA initial *S*-parameter measurement results. The simulated *S*-parameters shown in Figure 4c are in agreement with the measured one, thus validating our models. Figure 4d shows the simulated NF sweeping versus current biasing DAC code D_I at different temperatures. D_I directly controls I_S in Figure 4a. Monte Carlo simulation results show that both gain (with mean value of 17.04 dB and standard deviation of 2.19 dB) and NF (with mean value of 5.15 dB and standard deviation of 0.49 dB) present large variability for this design.

The variation of NF and gain of the LNA will significantly affect the performance of the whole receiver system. In our case, the first stage of the receiver is the LNA, followed by an RF mixer as the second stage and an in phase and phase quadrature (IQ) mixer as the third stage. According to Friis' formula, the total NF of the receiver system F_{RX} is represented as

$$F_{\mathrm{RX}} = F_{\mathrm{LNA}} + \frac{F_{\mathrm{RF\_MIXER}} - 1}{G_{\mathrm{LNA}}} + \frac{F_{\mathrm{IQ\_MIXER}} - 1}{G_{\mathrm{LNA}} \times G_{\mathrm{RF\_MIXER}}} + \cdots \quad (8)$$

where F represents the NF, G represents the gain, and subscripts represent the name of the stage. We can see that the NF of all stages after the LNA will be attenuated by the gain of LNA, so the NF of the receiver F_{RX} is mostly determined by LNA NF and gain. Therefore, it is essential to overcome the variations in the LNA to achieve low noise for the overall receiver. In this paper, we assume that the gain of LNA will be measured by an on-chip peak detector, and we focus on NF self-healing only.

Indirect NF sensing

NF is generally difficult and expensive to measure directly on chip. Hence, here we apply the indirect sensing technique, correlating NF with easy-to-measure PoMs. We collect a set of transistor-level Monte Carlo simulation data over the joint space of process, temperature T, and bias current I_S. After the simulation data are collected, we are able to apply the presilicon indirect sensor modeling procedure described in the Presilicon Indirect Sensor Modeling section. In this LNA example, there are multiple possible PoMs (e.g., peak detector output voltage, dc voltages, temperature) that can be correlated with NF. We need to mention here that the LNA chip in Figure 4 does not include a peak detector at 60 GHz. This is because the LNA will be integrated with a downconversion mixer that will include a peak detector at the intermediate frequency (IF) which is easier to design. In this paper, we assume that the output signal amplitude is sensed by a "virtual" peak detector. In simulations, we record the 60-GHz output signal amplitude and use it as one of PoM.

By applying the SR algorithm, we are able to find the most important PoMs and select the important high-order terms associated with them. The final PoM set includes V_{PD} (the output voltage of the peak detector), V_N (the drain dc voltage in the biasing stage in Figure 4a), D_I (the digital code controlling the bias current I_S in tunable current mirror), and T (temperature). Correlation coefficients between NF and V_{PD}, V_N, D_I, and T are -0.82, -0.81, -0.67, and 0.30, respectively. Here, V_{PD} can be measured by a peak detector, V_N can be measured by an on-chip ADC, and T can be measured by an on-chip temperature sensor. D_I can be directly known from the digital code. The indirect sensor model solved from SR is

$$\begin{aligned}\mathrm{NF}(V_{PD}, D_I, T, V_N) &= a_{1000}V_{PD} + a_{0100}D_I + a_{0010}T \\ &\quad + a_{2000}V_{PD}^2 + a_{0200}D_I^2 + a_{0020}T^2 + a_{0030}T^3 \\ &\quad + a_{0003}V_N^3 + a_{1010}V_{PD}T + a_{3000}V_{PD}^3 + a_{0000}. \quad (9)\end{aligned}$$

The mean error of this indirect sensor model is 0.170 dB.

Given that V_{PD}, V_N, and T require on-chip measurements, the impact of quantization error is also considered for the proposed indirect sensor. According to the variation range of V_{PD}, V_N, and T in simulation data, the sensors' dynamic ranges are 0–0.1 V for V_{PD}, 0.2–0.4 V for V_N, and -20 °C–80 °C for T. A sweeping analysis for the bits is performed to evaluate the error under different quantization conditions. Final decisions for quantization bits for V_{PD}, V_N, and T are 5, 4, and 4 bits, respectively. The quantization bits are selected as the minimum bits where not much accuracy is lost. With the

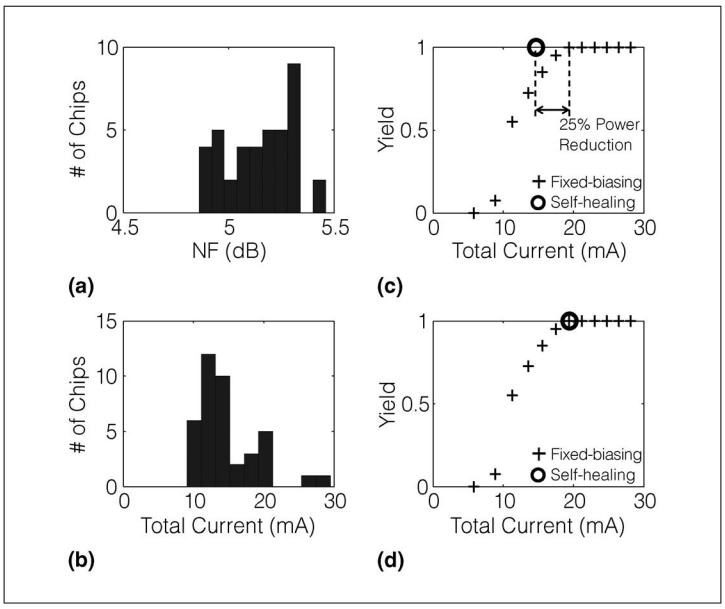

Figure 5. Monte Carlo simulation results. (a) and (b) NF and total current histograms after self-healing. (c) Comparison between self-healing with peak detector and fixed-biasing cases. (d) Comparison between self-healing without peak-detector and fixed-biasing cases.

quantization effect considered, the NF indirect sensor error is 0.175 dB.

Self-healing algorithm and results

Using the indirect sensor model, we can predict NF according to on-chip sensor measurements. In this context, we propose the following self-healing algorithm.

1) Set bias current of all chips to minimum value.
2) V_{PD}, V_N, and T are measured using on-chip sensors.
3) Calculate predicted NF of each chip using the indirect sensor model and compare the result with the NF specification. If the NF meets the specification, the algorithm stops. Otherwise, the bias current is increased by a small value, after which steps 2) and 3) are repeated until the NF specification is met or the maximum DAC control word is reached.

The algorithm generally tries to find the minimum bias current that meets the NF specification.

Indirect sensor model error must be carefully considered during self-healing. Due to the NF prediction error, the estimated NF will be different from the actual NF. Therefore, to handle the uncertainty in NF prediction, a guard band is required. The guard band is the extra margin we leave for NF in self-healing, so that high yield can be achieved. The size of the guard band is determined by statistically modeling indirect sensor error. First, we collect the error data from indirect sensor model fitting. The error data are then fitted against a distribution by using kernel density estimation. The guard band can then be optimally determined once the error distribution is known. The calculated guard band is added to the predicted NFs for all chips to guarantee high yield.

To validate the self-healing algorithm with guard band at the simulation and design level, 40 chips are randomly generated from transistor-level simulations. After applying the self-healing procedure and adding guard band, the 40 chips achieve 100% yield with the NF specification of 5.5 dB. The average total LNA current for all chips is 14.7 mA. The histograms of NF and total current are shown in Figure 5a and b. We also consider the fixed-biasing cases for comparison purpose. In the fixed-biasing cases, all the chips select the same tuning knob configuration. The power and yield of a set of fixed biasing cases are compared with the proposed self-healing method (with peak detector) in Figure 5c. The self-healing method is able to achieve 25% power reduction compared to the best fixed biasing case (with 19.4-mA total current), while not losing any yield. The key reason for the proposed self-healing method achieving better performance is that it adaptively selects an optimum bias current for each chip. For the chips with good NF, the algorithm will try to bias at low current so that power consumption is low. For the chips with bad NF, the algorithm tends to bias at a high current value so that the chip can meet the NF specification. We also compare the fixed biasing cases with the self-healing method (without peak detector) in Figure 5d. The indirect sensor model, quantization, and guard band are formulated similarly as the case with peak detector. In this case, the self-healing method is able to achieve the same performance as the best fixed-biasing case. In future work, we plan to implement the algorithm on the microcontroller and perform all the mm-wave measurement across P, V, and T.

IN THIS PAPER, we propose a theoretical framework that allows the prediction of RF and mm-wave circuit

performance with on-chip sensors through the use of indirect sensing with BMF instead of through costly and difficult direct integrated measurement. Because the simpler sensors required by indirect sensing, along with the circuit actuators, can be efficiently integrated, a large number of adaptive circuit loops can be envisioned, which will allow transceivers to adapt to process variability and external changes such that the energy spent by bit transmitted is minimized. The adaptation can be performed by algorithms running on an integrated microcontroller, or, alternately, using off-chip compute engines accessing internal sensors and actuators. We proposed a design and test methodology that allows the verification and integration of the software with the digital, mixed-signal, RF, and mm-wave circuits. This methodology also enables significant convergence between the code used for circuit verification and that use for test. Finally, we described the design example of a 60-GHz LNA that can be self-healed using indirect NF sensing and adaptive biasing. Monte Carlo simulations show that the LNA average power consumption can be improved by 25% with adaptive biasing while achieving the same 100% yield as that achieved with the optimum fixed bias. ■

■ References

[1] C. Maxey, G. Creech, S. Raman, and J. Rockway, "Mixed-signal SoCs with *in situ* self-healing circuitry," *IEEE Design Test Comput.*, vol. 29, no. 6, pp. 27–39, Nov./Dec. 2012.

[2] C. Chien, A. Tang, F. Hsiao, and M. F. Chang, "Dual-control self-healing architecture for high-performance radio SoCs," *IEEE Design Test Comput.*, vol. 29, no. 6, pp. 40–51, Nov./Dec. 2012.

[3] M. Margarit, J. L. Tham, R. Meyer, and M. Deen, "A low-noise, low-power VCO with automatic amplitude control for wireless applications," *IEEE J. Solid-State Circuits*, vol. 34, no. 6, pp. 761–771, Jun. 1999.

[4] W. Khalil, B. Bakkaloglu, and S. Kiaei, "A self-calibrated on-chip phase-noise measurement circuit with −75 dBc single-tone sensitivity at 100 kHz offset," *IEEE J. Solid-State Circuits*, vol. 42, no. 12, pp. 2758–2765, Dec. 2007.

[5] N. Kupp, H. Huang, Y. Makris, and P. Drineas, "Improving analog and RF device yield through performance calibration," *IEEE Design Test Comput.*, vol. 28, no. 3, pp. 64–75, May/Jun. 2011.

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

[7] J.-O. Plouchart et al., "A 23.5 GHz PLL with an adaptively biased VCO in 32 nm SOI-CMOS," *IEEE Trans. Circuits Syst.*, vol. 60, no. 8, pp. 2009–2017, Aug. 2013.

[8] B. Sadhu et al., "A linearized, low-phase-noise VCO-based 25-GHz PLL with autonomic biasing," *IEEE J. Solid-State Circuits*, vol. 48, no. 5, pp. 1138–1150, May 2013.

[9] S. Yaldiz et al., "Indirect phase noise sensing for self-healing voltage controlled oscillators," in *Proc. IEEE Custom Integr. Circuits Conf.,* Sep. 2011, DOI: 10.1109/CICC.2011.6055416.

[10] F. Wang, W. Zhang, S. Sun, X. Li, and C. Gu, "Bayesian model fusion: Large-scale performance modeling of analog and mixed-signal circuits by reusing early-stage data," in *Proc. Design Autom. Conf.*, Jun. 2013, pp. 1–6.

[11] X. Li, "Finding deterministic solution from underdetermined equation: Large-scale performance modeling of analog/RF circuits," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 29, no. 11, pp. 1661–1668, Nov. 2010.

J.-O. Plouchart is a Research Staff Member at IBM Research, Yorktown Heights, NY, USA, working on communication circuits and technologies. Plouchart has a PhD in electronics from the University of Paris, Paris, France. He is a senior member of the IEEE.

Benjamin Parker is retired. Parker has an MS in physics from Brown University, Providence, RI, USA.

Bodhisatwa Sadhu is currently a Research Staff Member at T.J. Watson Research Center, IBM Research, Yorktown Heights, NY, USA. His research interests include low-noise and wide-tuning range PLLs, circuits for cognitive radios, and millimeter-wave transceiver circuits. Sadhu has a PhD in electrical engineering from the University of Minnesota, Minneapolis, MN, USA (2012). He is a member of the IEEE.

Alberto Valdes-Garcia is currently a Research Staff Member and Manager of the RF Circuits and Systems Group, IBM Research, Yorktown Heights, NY, USA. His present work is on silicon-integrated millimeter-wave systems and carbon electronics. Valdes-Garcia has a PhD in electrical engineering from Texas A&M University, College Station, TX, USA (2006). He is a senior member of the IEEE.

Daniel Friedman joined IBM Research, Yorktown Heights, NY, USA, in 1994. He now manages the Communication Circuits and Systems group there, investigating I/O, millimter-wave communication, and computation acceleration. Friedman has a PhD in engineering science from Harvard University, Cambridge, MA, USA (1992). He chairs the IEEE International Solid-State Circuits Conference (ISSCC) wireline subcommittee. He is a member of the IEEE.

Fa Wang is currently working toward a PhD at the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, USA. Wang has a BS in control theory and information technology from Tsinghua University, Beijing, China (2009). He is a student member of the IEEE.

Xin Li is currently an Associate Professor in the Electrical and Computer Engineering Department and the Assistant Director of the Center for Silicon System Implementation, Carnegie Mellon University, Pittsburgh, PA, USA. His research interests include integrated circuit and signal processing. Li has a PhD in electrical and computer engineering from Carnegie Mellon University (2005).

Mihai Sanduleanu is currently an Associate Professor at Masdar Institute of Science and Technology, Abu Dhabi, United Arab Emirates. He was previously with T.J. Watson Research Center, IBM Research, Yorktown Heights, NY, USA. His area of research includes wireless transceiver design for RF/millimter-wave/terahertz communication, and THz electronics. Sanduleanu has a PhD from the University of Twente, Enschede, The Netherlands.

Andreea Balteanu is currently working toward a PhD at the University of Toronto, Toronto, ON, Canada. She received the Best Student Paper Award at the 2012 IEEE International Microwave Symposium. Balteanu has a BASc from the University of Waterloo, Waterloo, ON, Canada (2007) and an MASc from the University of Toronto (2010).

■ Direct questions and comments about this article to J.-O. Plouchart, IBM T. J. Watson Research Center, Yorktown Heights, NY 10598 USA; plouchar@us.ibm.com.