3-2005

# Built-In-Self-Testing Techniques for Programmable Capacitor Arrays

Amit Laknaur
*Southern Illinois University Carbondale*

Haibo Wang
*Southern Illinois University Carbondale*, haibo@engr.siu.edu


Published in Laknaur, A., & Wang, H. (2005). Built-in-self-testing techniques for programmable capacitor arrays. Proceedings of the Sixth International Symposium on Quality Electronic Design (ISQED'05), 434-439. doi: 10.1109/ISQED.2005.28 

# Built-In-Self-Testing Techniques for Programmable Capacitor Arrays

Amit Laknaur and Haibo Wang
Department of Electrical and Computer Engineering
Southern Illinois University Carbondale
Carbondale, IL 62901

## Abstract

*Programmable capacitor arrays (PCAs) are frequently used in reconfigurable analog circuits. Since PCAs can be programmed to numerous values, testing PCAs by exhaustively examining all PCA values can lead to lengthy testing processes. To address this problem, we present an efficient built-in-self-testing (BIST) method for PCAs used in reconfigurable analog circuits. The proposed BIST method takes advantage of existing programmable resources and, hence, introduces very small hardware overhead. Additionally, we present two simple and effective capacitor comparison techniques for implementing the proposed BIST method. The accuracy of the proposed circuit techniques is investigated and closed-form equations are derived for estimating comparison accuracy that can be achieved by the proposed techniques. Finally, circuit simulations are performed to validate the proposed techniques.*

## 1. Introduction

Adding reconfiguration capabilities to VLSI circuits is becoming an interesting trend in hardware design. In digital domain, a large number of reconfigurable digital systems have been demonstrated for real applications. Meanwhile, in analog domain many efforts have also been devoted to developing programmable analog circuits, which include programmable gain amplifiers [1, 2, 3], reconfigurable filters [4, 5, 6, 7], programmable neural networks [8, 9], and state-of-the-art field programmable analog arrays (FPAAs) [10, 11, 12, 13, 14, 15]. Such programmable analog circuits have been widely used in various applications. More interestingly, they provide viable approaches to implement intelligent systems, such as adaptive or self-repairing analog circuits.

While providing excellent design flexibility, programmable analog circuits also pose substantial testing challenges. Unlike conventional analog circuits, programmable circuits can have numerous configurations. Exhaustively testing each circuit configuration typically leads to lengthy testing processes. As a part of the effort to address the above testing challenges, this work tackles the problem of testing programmable capacitor arrays (PCAs), which are frequently used in reconfigurable analog circuits. Comparing to other programmable analog components (e.g. programmable resistors or transconductors), PCAs have advantages of high accuracy and large programmable ranges. As a result, many reconfigurable analog circuits use PCAs as a mechanism to program circuit parameters. Despite the popularity of PCAs in the design of reconfigurable analog circuits, efficient PCA testing methods have been rarely studied.

Previously, several works [16, 17, 18, 19, 20, 21, 22] were reported on testing programmable analog circuits. The majority of them focus on testing circuits that are implemented on programmable devices, rather than systematically testing all the resources on the given reconfigurable platforms. The study presented in [23] attempts to address the problem of systematically testing programmable resources on an FPAA device. However, its focus is limited to interconnect networks. In the past, techniques to measure capacitor ratios for detecting parametric faults in switched-capacitor (SC) circuits were studied in [24, 25]. The method proposed in [24] converts capacitor ratios into voltage outputs. In order to achieve accurate capacitor ratio measurement, circuits or instruments that can precisely read different voltage levels are needed. To avoid this stringent requirement, an analog-to-digital capacitor ratio converter (ADCRC) circuit is presented in [25]. The ADCRC relies on an integrator, a comparator, and simple digital arithmetic circuits to measure capacitor ratios.

This paper presents simple and effective circuit techniques to compare capacitor values. We also investigate the accuracy of the proposed techniques. Relations between comparison accuracy and circuit non-ideal parameters are derived. By using the proposed circuit techniques, an efficient built-in-self-testing (BIST) method for systematically testing PCAs is presented. The proposed BIST method takes advantage of programmable resources on the circuit under test. Hence, it introduces very small hardware overhead. Finally, experiments are conducted through circuit

Proceedings of the Sixth International Symposium on Quality Electronic Design (ISQED'05)
0-7695-2301-3/05 $ 20.00 IEEE

simulation. It demonstrates that the proposed BIST techniques are capable of detecting various PCA faults. The rest of the paper is organized as follows. Section 2 discusses PCA implementations and associated fault models. Section 3 develops circuit techniques for capacitor comparison. Section 4 describes the proposed PCA BIST method. Experimental results are provided in Section 5, and the paper is concluded in Section 6.

## 2. PCA implementation and fault models

A PCA normally contains a set of binary-weighted capacitors connected in parallel. In Figure 1(a), four binary-weighted capacitors constitute a PCA whose value can be programmed from 1C to 15C, where C is the unit capacitance used in the PCA. For each capacitor branch, a serial switch is used to configure the connection of the capacitor. In order to achieve good capacitor matching, all PCA binary-weighted capacitors are made of equally-sized small capacitors, which are often referred to as unit capacitors. Figure 1(b) shows the unit capacitor array used in the PCA.

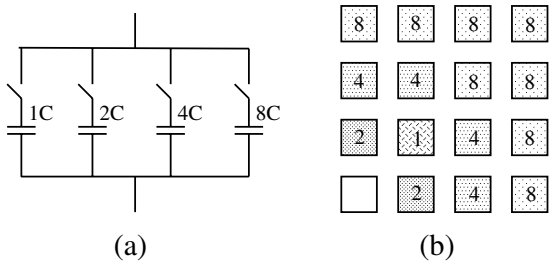


**Figure 1. Programmable capacitor array.**

Unit capacitors are typically implemented using two layers of polysilicon. The value of unit capacitors is selected to be significantly larger than parasitic capacitance. This requires the area of unit capacitors cannot be very small. Due to defects in the isolation layer (typically, silicon dioxide) that separates the two polysilicon layers, leakage paths may exist between the two terminals of a unit capacitor. We refer to this type of failure as leakage fault. In Figure 2(a), a parallel resistor $R_{leak}$ is used to model the leakage fault. When there are too many leakage paths and the isolation layer virtually becomes a conductor, the two terminals of the unit capacitor are shorted as shown in Figure 2(b). This type fault is called short fault. Furthermore, unit capacitors are connected through metal layers and contacts. Material defects on metal layers or contacts may disconnect a unit capacitor from its corresponding group. This leads to an open fault as shown in Figure 2(c). Another fault associated with interconnects is the bridge fault caused by either excess metal or a dust that connects two interconnects. The fault model for a bridge fault is shown in Figure 2(d). When the bridge fault is due to excess metal, bridge resistor $R_b$ is very small. However, if the fault is caused by a dust that is deposited during fabrication, $R_b$ can be considerably large.

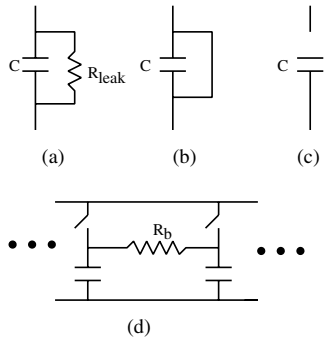


**Figure 2. PCA faults associated with unit capacitors.**

Switches used in PCAs are normally implemented using transmission gates, whose schematic is shown in Figure 3(a). Due to device failures or interconnect problems, a switch can remain always close or always open regardless of its control voltage. We refer to these failures as stuck-on and stuck-off faults. Their corresponding fault models are shown in Figure 3(b) and (c), respectively. More frequently, the switch characteristic of a transmission gate is deteriorated by too large on-resistance or too small off-resistance [26]. These two types of faults belong to the category of parametric faults, and they can be modeled by circuits shown in Figure 3(d) and (e). Many factors can be blamed for these two parametric faults. For example, abnormal threshold voltage as well as imperfect interconnect may lead to large on-resistance. Meanwhile, smaller-than-normal channel length may be a culprit for small off-resistance faults.

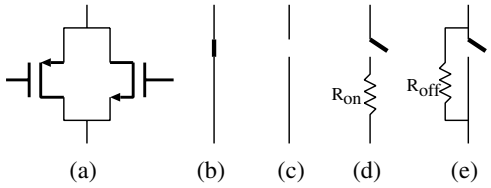


**Figure 3. PCA faults associated with switches.**

## 3. Circuit techniques for capacitor comparison

In this section, we propose two circuit techniques to compare capacitor values. The first technique aims to determining if the ratio of two capacitors is 2. The second technique targets detecting if two capacitors have the same value.

The first circuit technique is depicted in Figure 4. It is basically an SC integrator circuit. Initially, switch $s$ is

close to discharge integration capacitor $C_I$. After switch $s$ is open, $C_I$ starts to accumulate charge transferred from capacitors $C_A$ and $C_B$. During a clock cycle, the amounts of charge transferred from $C_A$ and $C_B$ are $-C_A \cdot \frac{V_1}{2}$ and $C_B \cdot V_1$, respectively. If $\frac{C_A}{C_B} = 2$, the net charge accumulated at $C_I$ during each clock cycle should be zero. However, if $\frac{C_A}{C_B} \neq 2$, certain amount of charge will be accumulated at $C_I$ and, consequently, the integrator output $V_O$ will be driven away from its normal level. We stop the integration operation after $\mathcal{N}_{CLK}$ clock cycles. Thereafter, the integrator output $V_O$ will indicate if $C_A$ is twice of $C_B$. The integrator circuit functions as an "amplifier" that amplifies small capacitor mismatches into large voltage signals. The gain of the "amplifier" can be adjusted by changing the value of $C_I$ or $\mathcal{N}_{CLK}$. The gain should be selected such that the integration circuit will detect large capacitor mismatches which are considered as faults, but ignore small mismatches caused by parasitic capacitance.

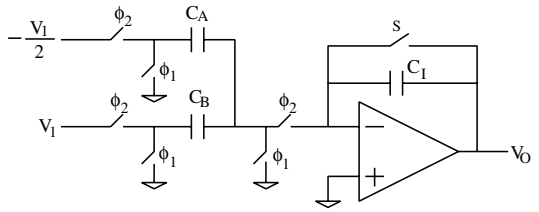


**Figure 4. Circuit technique to detect if $\frac{C_A}{C_B} = 2$.**

In Figure 4 we label the two input voltages as $V_1$ and $-\frac{V_1}{2}$ for a clear explanation. Actually, negative voltage is not needed in real circuit implementations. For example, assume the analog circuit under test requires a single power supply $V_{DD}$. Ground symbols in Figure 4 represent signal ground, which is typically at the level of $\frac{V_{DD}}{2}$. With the above conditions, we can apply $\frac{V_{DD}}{4}$ and $V_{DD}$ to $C_A$ and $C_B$ inputs, respectively. Comparing to signal ground, the voltage of $\frac{V_{DD}}{4}$ is a negative level and its magnitude is half of the signal magnitude applied at $C_B$ input.

The accuracy of the comparison results produced by the above circuit is predominately determined by the input offset voltage of the opamp. When considering the opamp input offset voltage $V_{OS}$, the mismatch between charge transferred from $C_A$ and $C_B$ can be expressed as:

$$\Delta Q = C_B \cdot (V_1 - V_{OS}) - C_A \cdot (-\frac{V_1}{2} - V_{OS}) \qquad (1)$$

Assuming $C_A$ and $C_B$ are two adjacent binary-weighted capacitors in a PCA, the values of $C_A$ and $C_B$ are given as $2^N \cdot C$ and $2^{N-1} \cdot C$, where $C$ is the unit capacitor value. If one unit capacitor in $C_A$ has an open fault, the realized value of $C_A$ will be $(2^N - 1) \cdot C$. Substituting these capacitor values into the above equation, we have:

$$\Delta Q = -3 \cdot 2^{N-1} \cdot V_{OS} \cdot C + \frac{V_1}{2} \cdot C \qquad (2)$$

The first term on the right hand side of Equation 2 is due to opamp input offset voltage. The second term is caused by the open fault. In order to correctly detect the open fault, we have to make the magnitude of the second term is at least $M$ times larger than that of the first term. This is:

$$\frac{V_1}{2} \cdot C > 3 \cdot M \cdot 2^{N-1} \cdot V_{OS} \cdot C \qquad (3)$$

From this inequality, we obtain the upper bound of comparison accuracy in terms of number of bits, which is given as:

$$N < log_2 \frac{V_1}{6 \cdot M \cdot V_{OS}} + 1 \qquad (4)$$

In order to get a general idea about the accuracy of the above testing circuit, we assume that the PCA under test works with a 3.3V power supply. Also, assume signal ground is 1.65V and voltage applied at $C_B$ input is 3.3. Therefore, $V_1$ is 1.65 comparing to signal ground. Furthermore, if we select $M$ as 5 and assume the opamp input offset voltage is 2mV. Then, the maximum accuracy that can be achieved by the proposed method is 5-bit. However, testing PCAs with large programmable ranges may require accuracy higher than 5-bit. To solve this problem, we can apply opamp input offset cancellation techniques [27] in the testing circuit as shown in Figure 5. Before performing integration, the circuit is initialized by closing switches $s_1 \sim s_3$. In this phase, the opamp is configured as a unit-gain buffer and it charges capacitor $C_p$ to the level of $V_{OS}$. After the initialization phase, we open $s_1 \sim s_3$ and close $s_4$. Thereafter, capacitor $C_p$ functions as a voltage shifter to cancel the opamp input offset voltage.

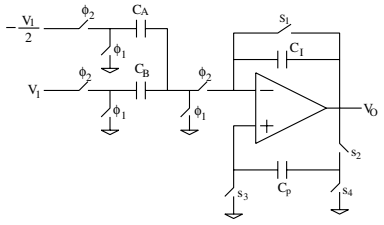


**Figure 5. PCA testing circuit with opamp input offset cancellation.**

Figure 6 shows the proposed circuit for detecting if two capacitors have the same value. Its operation can be analyzed similarly by tracing charge transferred from input capacitors to integration capacitor $C_I$. Note that opamp input offset voltage also affects the accuracy of this circuit. Without opamp input offset cancellation, the accuracy, in terms

of number of bits, that can be achieved by this circuit is:

$$N < log_2 \frac{V_1}{M \cdot V_{OS}} - 1 \quad (5)$$

Even with the help of opamp input offset cancellation, $V_{OS}$ cannot be completely cancelled due to other non-ideal effects, such as channel charge injection, clock feedthrough, clock skew, and leakage. Assume that the residual input offset voltage is $V'_{OS}$ after applying input offset cancellation techniques. Substituting $V'_{OS}$ into Equations 4 and 5, we can estimate the accuracy that can be achieved by the proposed circuits with opamp input offset cancellation.

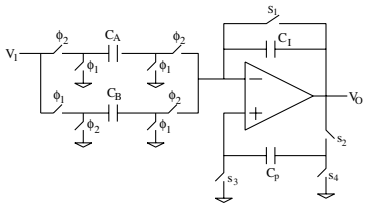


**Figure 6. Circuit technique to detect if $C_A = C_B$.**

## 4. Proposed PCA BIST method

The proposed BIST method aims to efficiently testing a large number of PCAs used in programmable analog circuits. It is comprised of two testing phases. In the first phase, a single PCA is tested to make sure all binary-weighted capacitors have proper values and none of the faults discussed in Section 2 occurs in the selected PCA. In the second phase, the tested PCA is used as a reference to examine other PCAs.

To more clearly explain the proposed method, we assume a scenario that the proposed BIST method is applied to testing PCAs in an SC-based FPAA circuit [28, 29]. The FPAA is comprised of configurable analog blocks (CABs), programmable interconnects, I/O cells, and configuration memories. PCAs that need to be tested are located in CABs, whose structure is sketched in Figure 7. It consists of one opamp, five PCAs, and a number of switches.

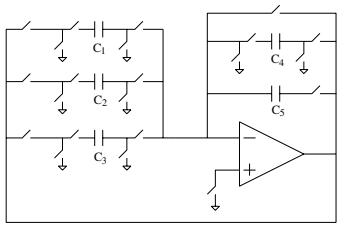


**Figure 7. FPAA CAB structure.**

In this discussion, we arbitrarily select $C_1$ as the reference PCA, which is to be tested first. Also, without losing generalities, we assume each PCA contains four binary-weighted capacitors. Following the technique depicted in Figure 5, the BIST circuit for PCA $C_1$ is constructed as shown in Figure 8. Switches $T_2 \sim T_4$, $SW_1$, $SW_2$, and their associated interconnects are extra resources along with the offset cancellation circuit that is added for implementing the proposed BIST scheme. The rest of the BIST circuit are implemented by existing resources in the CAB. In Figure 8, CAB resources that are not used in the BIST circuit are drawn by dash lines. To test the ratio between $1C$ and $2C$, we close switches $B_1$ and $T_2$ during integration. Similarly, we close $B_2$ and $T_3$ or $B_3$ and $T_4$ to examine the ratio between $2C$ and $4C$ or the ratio between $4C$ and $8C$, respectively.

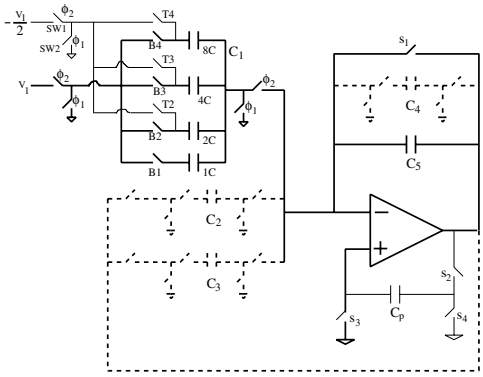


**Figure 8. Capacitor ratio testing.**

After the reference PCA is tested, we use the technique described in Figure 6 to examine the other PCAs. In this process, each binary-weighted capacitor in the PCA under test is compared with the corresponding capacitor in the reference PCA. The configured BIST circuit is very similar to the circuit shown in Figure 6. In the original CAB design, PCAs $C_4$ and $C_5$ are fixed as opamp feedback branches. Therefore, they cannot be tested using the above method. To address this problem, we propose to add additional programmable switches in the CAB such that $C_4$ and $C_5$ can be used as opamp input branches as well. Also, $C_5$ should have the same switch network as the other PCAs.

The above testing plan is capable of detecting all the faults discussed in Section 2. This is because all of these faults affect charge transfer in the BIST circuits and, consequently, lead to abnormal circuit outputs. Besides its excellent fault-detection capability, the proposed BIST method introduces small hardware overhead. In particular, it adds several switches, a small capacitor $C_p$. Note that switches $T_2 \sim T_4$ are added to the reference PCA only. Furthermore, the above discussion assumes that the circuit under test is an FPAA circuit. Similarly, the proposed BIST techniques can be easily applied to SC-based programmable filters as well as other reconfigurable analog circuits.

## 5. Experimental results

To demonstrate the validity of the proposed BIST techniques, circuit simulations are performed to detect PCA faults using the proposed testing method. Each PCA contains 8 binary-weighted capacitors and its value can be programmed from 1 to 255 unit capacitance (which is 250fF). All the switches used in PCAs are CMOS transmission gates with the same size for simplicity reasons. The transistor sizes are NMOS-$9\mu/0.32\mu$ and PMOS-$22.5\mu/0.32\mu$. The clock frequency used in the experiment is 1MHz, and the power supply of the circuit is 3.3V. To reduce simulation time, an opamp macromodel is used in simulation. Its key performance parameters are summarized in Table 1.

**Table 1. Opamp parameters used in simulation.**

| Opamp parameters | Values |
|---|---|
| Low frequency gain | 80dB |
| Unit-gain frequency | 10MHz |
| Common mode rejection ratio | 70dB |
| Input offset voltage | 4mV |
| Slew Rate | $20V/\mu s$ |
| Settling time (0.1%) | $0.5\mu s$ |
| Power supply | 3.3V |
| Output swing range | $0.18V \sim 3.1V$ |

In simulation, PCA faults are injected by using fault models shown in Figure 2 and 3. Also, the integration capacitor in BIST circuits is simply assigned to 16 unit capacitance. The number of clock cycles that integration is performed is 10. If there are no faults occurred in the PCA under test, the BIST circuit output should be 1.65V, which is the voltage level of signal ground. However, due to parasitic effects, the BIST output is always slightly away from its ideal value. We define the tolerant range as $\pm 0.45V$. Therefore, the PCA under test is fault-free if the BIST circuit output is within the range from 1.2V to 2.1V. Otherwise, the PCA is faulty.

Simulation results showed that all short, open, stuck-on, and stuck-off faults can be easily detected by the proposed method. Figure 9 compares BIST circuit outputs in faulty and fault-free scenarios when two capacitors, whose expected values are 64C and 32C, are tested. The bottom curve is the BIST circuit output in the fault-free scenario. The top curve is the BIST circuit output when an unit capacitor belonging to the $64C$ capacitor has an open fault. Since faults occurring at different positions in a PCA may affect charge transfer with different degrees, the difficulty to detect faults at different positions also varies. Table 2 summarizes BIST circuit outputs when short, open, stuck-on, and stuck-off faults take place in positions that make them most difficult to be detected. For comparison purposes, Table 2 also lists the corresponding BIST circuit outputs in fault-free scenarios. Simulation results also indicate the selection of $\pm 0.45V$ tolerant range is very conservative. In fault-free scenario, the maximum variation between simulated BIST outputs and their ideal value is only 0.15V.

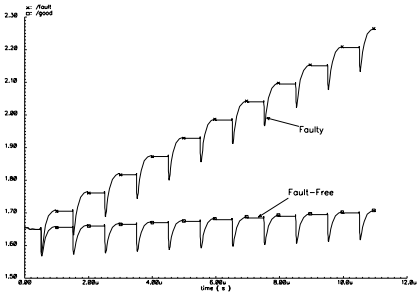


**Figure 9. BIST circuit output when an open fault occurs in the PCA under test.**

**Table 2. Comparison of BIST circuit outputs.**

| Fault Scenario | BIST Circuit Output | |
|---|---|---|
| | Faulty PCA | Faulty-Free PCA |
| Short fault | 0.182V | 1.67V |
| Open fault | 1.05V | 1.50V |
| stuck-on fault | 0.637V | 1.68V |
| stuck-off fault | 2.69V | 1.67V |

When introducing leakage, bridge, large on-resistance, and small off-resistance faults, we have to assign values to $R_{leak}$, $R_b$, $R_{on}$, and $R_{off}$, respectively. The ability of the proposed method to detect these faults are strongly affected by the resistor values. Through simulation, we found the ranges of $R_{leak}$, $R_b$, $R_{on}$, and $R_{off}$ that lead to detectable faults under the above experiment setup. These values are summarized in Table 3. In the search for these ranges, we always inject faults to positions such that these faults are most difficult to be detected. The data shown in Table 3 demonstrate that the proposed method can detect the above four types of faults with large parameter ranges.

**Table 3. Detectable ranges of PCA faults.**

| Fault name | Parameter | Range |
|---|---|---|
| Leakage fault | $R_{leak}$ | $0 \sim 0.85M\Omega$ |
| Bridge fault | $R_b$ | $0 \sim 1.6M\Omega$ |
| Large on-resist. fault | $R_{on}$ | $1.2M\Omega \sim \infty$ |
| Small off-resist. fault | $R_{off}$ | $0 \sim 1.7M\Omega$ |

## 6. Concluding remarks

This work studies the problem of testing programmable capacitor arrays, which are frequently used in reconfigurable analog circuits. Two circuit techniques are developed to

test PCA binary-weighted capacitors. The accuracy of the developed testing circuits is investigated, and closed-form equations are derived for estimating comparison accuracy that can be achieved by the proposed techniques. By using the proposed techniques, an efficient BIST plan for testing a large number of PCAs on reconfigurable analog platforms is presented. The proposed BIST method takes advantage of the programmability of the circuit under test, hence, introducing very small hardware overhead. Experiments have been conducted to investigate the effectiveness of the proposed method. It demonstrates that the proposed method detects all the short, open, stuck-on, and stuck-off faults. Simulation results also indicate that the proposed method has an excellent capability to detect a wide range of leakage, bridge, large on-resistance, and small off-resistance faults.

## 7. Acknowledgements

The authors would like to thank Tao Wu, Freescale Semiconductors, for the discussion about PCA implementation.

## References

[1] B. Calvo, S. Celma, and M. T. Sanz, "High-frequency digitally programmable gain amplifier," *IEE Electronics Letters*, vol. 39, no. 15, pp. 1095–1096, 2003.

[2] C. C. Hsu and J. T. Wu, "A highly linear 125-MHz CMOS switched-resistor programmable-gain amplifier," 2003.

[3] K. Nah and B. Park, "A 50-MHz dB-linear programmable-gain amplifier with 98-dB dynamic range and 2-dB gain steps for 3 V power supply," *IEEE Trans. VLSI*, vol. 11, no. 2, pp. 218–223, 2003.

[4] F. Dulger, E. Sanchez-Sinencio, and J. Silva-Martinez, "A 1.3-v 5-mw fully integrated tunable bandpass filter at 2.1 ghz in 0.35-/spl mu/m cmos."

[5] G. Bollati, S, Marchese, M. Demicheli, and R. Castello, "An eighth-order CMOS low-pass filter with 30-120 MHz tuning range and programmable boost," *IEEE J. Solid-State Circuits*, vol. 36, no. 7, pp. 1056–1066, 2001.

[6] J. I. Osa, A. Carlosena, and A. J. Lopez-Martin, "MOSFET-C filter with on-chip tuning and wide programming range," *IEEE Ttrans. Circuits and Systems II: Analog and Digital Signal Processing*, vol. 48, no. 10, pp. 944–951, 2001.

[7] T. Ndjountche and R. Unbehauen, "Improved Structures for Programmable Filters: Application in a Switched-Capacitor Adaptive Filter Design," *IEEE Trans Circuits and Systems II: Analog and Digital Signal Processing*, vol. 46, no. 9, pp. 1137–1147, 1999.

[8] P. Kinget and M. Steyaert, "A Programmable Analog Cellur Neural Network CMOS Chip for High Speed Image Processing," *IEEE Journal of Solid State Circuits*, 1995.

[9] M. A. A. El-Soud, R. A. AbdelRassoul, H. H. Soliman, and L. M. El-Ghanam, "Low-power CMOS circuits for analog VLSI programmable neural networks," in *Proc. 15th International Conference on Microelectronics*.

[10] D. D'mello and G. Gulak, "Design Approaches to Field-Programmable Analog Integrated Circuits," *Analog Integrated Circuits and Signal Processing*, vol. 17, no. 2, pp. 7–34, 1998.

[11] B. Pankiewicz, M. Wojcikowski, S. Szczepanski, and Y. Sun, "A Field Programmable Analog Array for CMOS continuous-time OTA-C filter applications," *IEEE Journal of Solid-State Circuits*, vol. 37, no. 2, pp. 125–136, 2002.

[12] M. Mar, B. Sullam, and E. Blom, "An Architecture for a Configurable Mixed-Signal Device," *IEEE Journal of Solid-State Circuits*, vol. 38, no. 3, pp. 565–568, 2003.

[13] Anadign, Inc. http://www.anadigm.com.

[14] Cypress Semiconductor. http://www.cypress.com.

[15] Lattice Semiconductor. http://www.latticesemi.com/.

[16] T. Slaughter and C. Stroud, "Fault Injection Emulation for Field Programmable Analog Arrays," in *Proc. Southwest Symposium Mixed-Signal Design*, pp. 212–216, 2003.

[17] R. S. Zebulum, D. Keymeulen, et. al., "Experimental Results in Evolutionary Fault-Recovery for Field Programmable Analog Devices," in *Proc. NASA/DoD Conf. on Evolvable Hardware*, 2003.

[18] C. A. Looby and C. Lyden, "Field Programmable Analogue Arrays: A DFT View," in *IEE Colloquium on Testing Mixed Signal Circuits and Systesm*, 1997.

[19] R. Sanahuja, V. Barcons, L. Balado, and J. Figueras, "X-Y Zoning BIST: An FPAA Experiment," in *Proc. Intl Mixed-Signal Test Workshop*, 2002.

[20] R. Sanahuja, V. Barcons, L. Balado, and J. Figueras, "Testing Biquad Filters under Parametric Shifts using X-Y Zoning," in *Proc. Intl Mixed-Signal Test Workshop*, 2003.

[21] T.Balen, A.Andrade Jr, F.Azais, M.Lubaszewski, M. Renovell, "An Approach to the Built-In Self-Test of field Programmable Analog Arrays," in *Proceeding of IEEE VLSI Test Symposium*, 2004.

[22] H. Wang, S. Kulkarni, and S. Tragoudas, "Circuit Techniuqes for Field Programmable Analog Array On-Line Testing," in *Proc. 10th International Mixed-Signal Testing Workshop*, pp. 237–244, 2004.

[23] A. Andrade, G. Vieira, M. Lubaszewski, etc, "Testing Global Interconnects of Field Programmable Analog Arrays," in *Proc. 10th International Mixed-Signal Testing Workshop*, pp. 231–246, 2004.

[24] C. Dufaza and H. IHS, "Test Synthesis for DC Test and Maximal Diagnosis of Switched-Capacitor Circuits," in *Proc. VLSI Test Symposium*, pp. 252–260, 1997.

[25] B. Vinnakota and R. Harjani, "DFT for Digital Detection of Analog Parametric Faults in SC Filters," *IEEE Trans CAD*, vol. 19, no. 7, pp. 789–798, 2000.

[26] R. Rodriguez-Montanes, D. Munoz, L. Balado, and J. Figueras, "Analog Switches in Programmable Analog Devices: Quiescent Defective Behaviours," 2002.

[27] P. Allen and D. Holberg, *CMOS Analog Circuit Design*. Oxford University Press, 2002.

[28] A. Bratt and I. Macbeth, "DPAD2 – A Field Programmable Analog Array," *Analog Integrated Circuits and Signal Processing*, vol. 17, no. 2, pp. 67–89, 1998.

[29] Motolora Inc., *EasyAnalog Design Software User's Manual*.