Analog Integrated Circuits and Signal Processing (2022) 112:355–365
https://doi.org/10.1007/s10470-022-02055-6

A built-in self-test for analog reconfigurable filters implemented in a mixed-signal configurable processor

Emanuel Dri[1] · **Gabriela Peretti**[1,2] · **Eduardo Romero**[1,2]

Received: 8 February 2022 / Revised: 23 March 2022 / Accepted: 9 May 2022 / Published online: 2 June 2022

Abstract
This work introduced a new BIST scheme under a functional test approach. It solves the problem of determining in-field the specifications of lowpass filters embedded in PSoC1 devices with zero hardware overhead. The user can implement the BIST we propose here because it just requires the information and resources freely offered by the manufacturer, the nominal specifications of the filter, and its location within the resources array of the chip. It consists of a software routine that reconfigures the resources available in the device to synthesize all the circuitry needed for the test: stimuli generation, filter response measurement and analysis, data processing, and a communication interface to output the test results. The proposal is based on the transient response analysis method (TRAM), a test strategy for analog filters used to determine the functional parameters of the circuits under test. The BIST performance was evaluated by comparing it against frequency response and TRAM measurements. The laboratory results showed low errors and good repeatability, validating the proposal.

Keywords Built-in self-test · Mixed-signal reconfigurable hardware · Programmable · System on chip · Switched-capacitor filters · Transient response analysis method

1 Introduction

Configurable analog sections (CASs) have resources like operational amplifiers, switches, analog multiplexers, and programmable arrays of capacitors and resistors, all interconnected through a configurable network. A typical example of a device that contains one or more CAS is the field-programmable analog array or FPAA, considered the analog counterpart of the broadly known field-programmable gate array (FPGA). Moreover, some modern configurable processors (CPs) offer interconnection between CASs and Configurable Digital Sections (CDSs) added to the processing capabilities, thus enabling the implementation of mixed-signal modules. As a result, an increasing number of applications use circuits with CASs, like those recently reported in [1–5].

CPs (also FPAAs) reduce inventory costs, power requirements, size, weight, and time to market. These characteristics, added to the usually low-cost and easy-to-use development tools for CPs, allow small and medium-scale industries to improve their market opportunities.

Despite their benefits, the test of systems configured in CPs is challenging. Test procedures are essential for high-quality production processes because they prevent the consumer from receiving faulty products. Additionally, they offer vital information about malfunctions in manufacturing operations. However, there is another aspect of testing usually overlooked by technicians in small industries: the competitive advantage that in-field tests can provide to their products. In-field test applies during the device or system operational life to guarantee their fault-free condition, and it is essential for a wide range of critical applications like medical, automotive, and industry, which demand periodic testing of electronic systems.

✉ Gabriela Peretti
gperetti@frvm.utn.edu.ar; gabriela.peretti@unc.edu.ar

Emanuel Dri
emanueldri@frvm.utn.edu.ar; emanueldri@ymail.com

Eduardo Romero
eromero@frvm.utn.edu.ar; eduardo.abel.romero@unc.edu.ar

[1] Mechatronics Quality Research Group, Universidad Tecnológica Nacional – Facultad Regional Villa Maria, Av. Universidad 450, Villa María, Argentina

[2] Electronics and Instrumental Development Group, Universidad Nacional de Cordoba – Facultad de Matemática Astronomia y Fisica, Av Medina Allende, Ciudad Universitaria, Córdoba, Argentina

When an application uses CPs, it is usual to implement in the CASs a significant amount of the analog hardware. Therefore, it becomes relevant to provide test methodologies for them, easily implemented by design engineers in the industry. It should be highlighted that the configurable nature prevents the vendor from knowing the circuit configured in the application. Therefore, the user is responsible for developing and validating test strategies for the circuits configured in the ACSs. This task implies a set of challenges different from the ones that test engineers must face when designing test schemes in integrated-circuits production environments, mainly because the user does not have detailed information about the internal circuitry of the chips.

Due to proprietary reasons, the vendors do not disclose detailed structural information about CASs. In addition, the user must frequently deal with partial characterizations and (in some cases) lack of simulation models. These problems complicate the test planning and make the typical difficulties that analog testing exhibits even more severe, such as the complex nature of the involved signals and the impossibility of establishing a clear signal flow (as with digital circuits). Consequently, the test of circuits with CASs is very challenging, with a relatively low number of publications addressing this topic [6–13].

In this work, we use the Transient Response Analysis Method (TRAM), proposed in [14], to test second-order filters. This method has been applied to a few FPAAs circuits in the last 20 years, showing promising results. The authors of [8] test via TRAM the ACSs of two commercial FPAAs focusing on detecting structural-level faults without measuring the functional parameters of the circuits under test (CUT). They do not consider statistical variations in the test parameters, which could compromise the fault detection ability. In [10], TRAM is applied to obtain the functional specifications of second-order filters implemented on a Lattice device, but without reports about experimental data or variability considerations. The authors of [12] present a TRAM-based Design-for-Test (DfT) scheme for testing filters embedded in PSoC1 devices with a functional approach. This DfT requires equipment and technicians for its operation. The same filters are studied in [13], but using a structural approach instead and similarity measures to analyze the test signals.

We propose a novel Built-in-self-Test (BIST) based on TRAM for switched-capacitor (SC) filters embedded in PSoC1. We use a functional approach to determine the specifications of the filters under test, considering only the public information published by the vendor for test formulation. The proposed strategy runs autonomously using only the device's internal resources, introducing solutions to circumvent hardware limitations avoiding hardware overheads and external equipment. The obtained results show that the strategy is competitive. Also, the validation method is entirely experimental and overcomes the problems derived from the reduced information disclosed by the vendor.

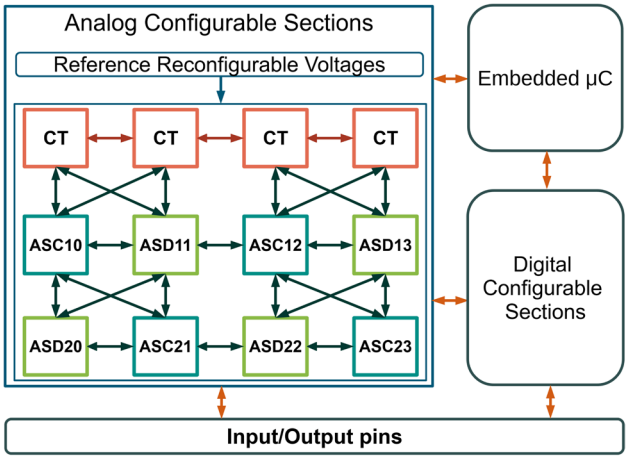

Fig. 1 PSoC1 platform resources

2 Device under test

2.1 PSoC1 platform

PSoC1 offers the typical resources of a modest microcontroller (µC): an 8 bits processing core, embedded RAM, a flash ROM, and communication interfaces. Nevertheless, its power resides in the embedded ACSs and DCSs for implementing complex functions and systems. The use of a single configurable module or a combination of them facilitates the synthesis of a wide variety of circuits, like discrete-time amplifiers or delta-sigma ($\Delta\Sigma$) converters, for mentioning only two of them. In this way, a broad range of schemes for signal processing, control, and synchronization can be implemented directly on the configurable hardware, reducing the demands of the processing core.

Figure 1 shows a conceptual diagram of the primary resources inside the platform. For this work, the blocks in the analog array are of particular interest. These are Continuous Time (CT), as well as Switched Capacitors (SC) of types C (ASC) and D (ASD). The arrows in the diagram represent the configurable interconnections available in the chip, which impose constraints for testing.

2.2 Biquads filters configured in PSoC1

PSoC1 implements filters using second-order sections, also called biquads. By cascading these sections, it is possible to design high-order filters. Each biquad is configured in two blocks (ASC and ASD) with the schematic shown in Fig. 2. Here, $\phi 1$ and $\phi 2$ are the two non-overlapping clock

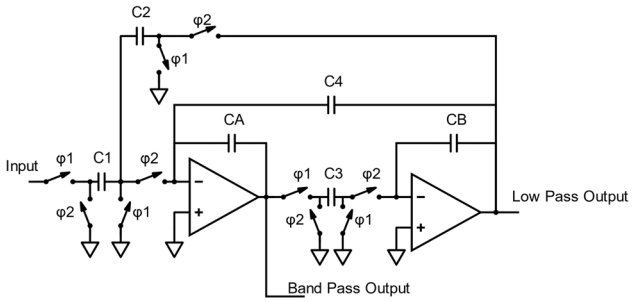

Fig. 2 Biquad filter implemented in PSoC1

Table 1 Lowpass filter resources

Filter	Input block	Output block
F0HA|B	ASC10	ASD11
F1HA|B	ASC21	ASD20
F2HA|B	ASC12	ASD13
F3HA|B	ASC23	ASD22
F0VA	ASC10	ASD20
F1VA|B	ASC21	ASD11
F2VA	ASC12	ASD22
F3VA|B	ASC23	ASD13

signals with a frequency fs_{nom}, and the capacitor values are digitally programmable.

The filters can be placed in several positions in the analog array and configured as lowpass or pass-band [15]. Their transfer functions are characterized by the following functional parameters: pole frequency (ω_p), pole quality factor (Qp), and gain (K).

In this work, we consider only the lowpass implementation. For easy comparison with [12], this paper uses the same nomenclature for the filters. The name assigned to a given filter has the structure FNOT, where F denotes filter, N is the number of the filter, O its orientation (vertical or horizontal), and T the type of filter (A, B or A|B that means that could be A or B). Table 1 shows the resources used by the 14 possible implementations in the device used as a platform, the CY8C29466PXI. As the filters use these blocks, they cannot be employed to implement the test resources. Therefore, the circuits required for implementing the test schemes must be located on different positions of the analog array, depending on the filter under the test.

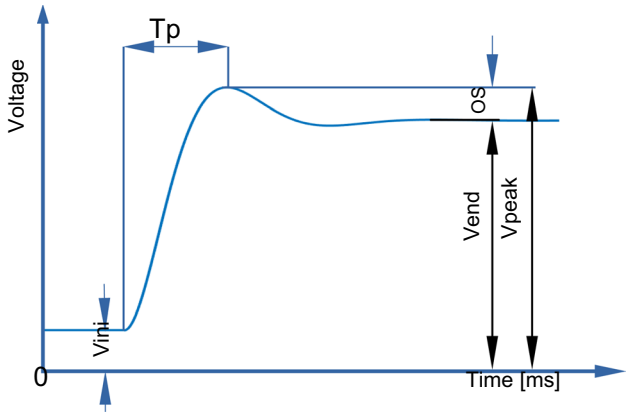

Fig. 3 TRAM response attributes

3 Test proposal

3.1 TRAM and software-based test: basic concepts

Our test combines the principles of the software-based test (SWBT) and TRAM. An SWBT strategy takes advantage of the processing core and the hardware resources already present in the system to test some of the internal circuits of the chip without hardware overhead [16]. The test process uses the principles of TRAM, a well-known test method targeted for second-order filters. It is based on applying a suitable test stimulus to the input filter (step, ramp, or parabola) to induce an underdamped response [14].

Figure 3 shows a typical TRAM test response with the primary parameters usually measured: the peak time (Tp) and the overshoot (OS). It also shows V_{in}, V_{end}, and V_{peak}, the voltages needed to get the value of OS%, according to (1). With these measurements and the amplitude of the input step signal (V_{input}), are obtained the functional parameters of the filter K, ω_p, and Qp through expressions (2) to (4) [17].

$$OS\% = \frac{V_{peak} - V_{in}}{V_{end} - V_{in}} \times 100\% \tag{1}$$

$$K = \frac{V_{end} - V_{in}}{V_{input}} \tag{2}$$

$$Q_p = \frac{1}{2}\sqrt{\left(\frac{\pi}{\ln\left(\frac{OS\%}{100}\right)}\right)^2 + 1} \tag{3}$$

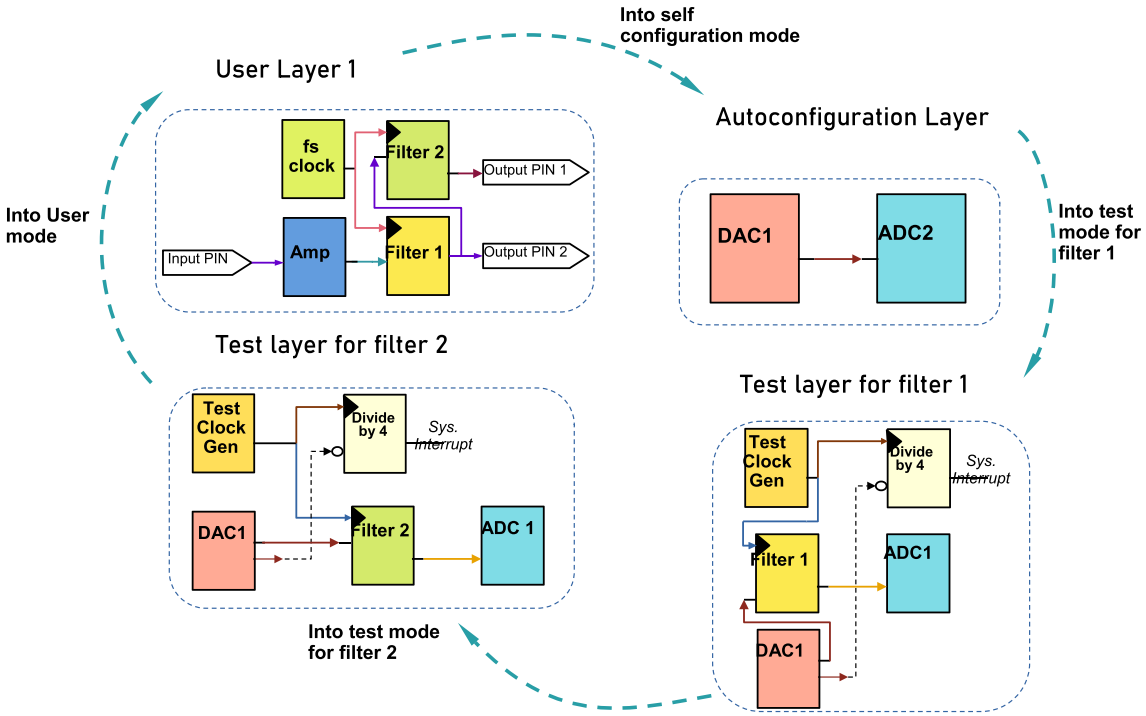

Fig.4 Arrangement of layers for implementing the test strategy

$$\omega_p = \frac{\pi}{Tp\sqrt{1 - \frac{1}{4 \cdot Q_p{}^2}}} \tag{4}$$

3.2 Dynamic reconfiguration of the hardware for testing

The dynamic reconfiguration ability of PSoC1 allows changing the functions of the ACSs and DCSs by software (SWBT). The test proposal presented in this paper takes advantage of this characteristic and generates several design layers with block configurations that an embedded routine can change. One of these layers is the user application (that runs in normal mode), and the remaining ones are needed to test the filters. In addition, one more layer is specifically intended to determine the amplitudes of the signals delivered by the test stimuli generators. Finally, the test firmware performs all the reconfigurations required by the BIST and the operations that determine the functional parameters of the CUTs (ω_p, Qp, and K). An advantage of dynamic reconfiguration is that it is very straightforward to repeat the concept several times to test different filters according to the user's needs.

For the sake of clarity, Fig. 4 shows an example with the hardware resources arranged in the layers. First, the User Layer1 has resources for the regular operation of the system, in this case, an amplifier and a filter. Next, the Autoconfiguration Layer rearranges the hardware to instantiate a DAC (DAC1) and an ADC (ADC2) and make a self-calibration routine. Finally, the remaining two layers (test layer for filter1 and filter2) incorporate the blocks for testing Filter1 and Filter 2.

Table 2 ADCs and DACs resources

Name	Resources
ADC1	ASC12–ASD22
ADC2	ASC21–ASD11
ADC3	ASD11–ASC21
ADC4	ASC23–ASD13
DAC1	ASD20
DAC2	ASC12
DAC3	ASD11
DAC4	ASD22
DAC5	ASD13

3.3 Resources used by the BIST

One of the goals of BIST is to solve test stimuli generation and test response measurements with the internal resources of the processor. For this purpose, test stimuli generation is implemented by 6-bits digital to analog converters (DACs), while 14-bits ΔΣ analog-to-digital converters (ADCs) measure the test responses. ADCs and DACs are configured with on-chip SC blocks, shown in Table 2, and some CDSs.

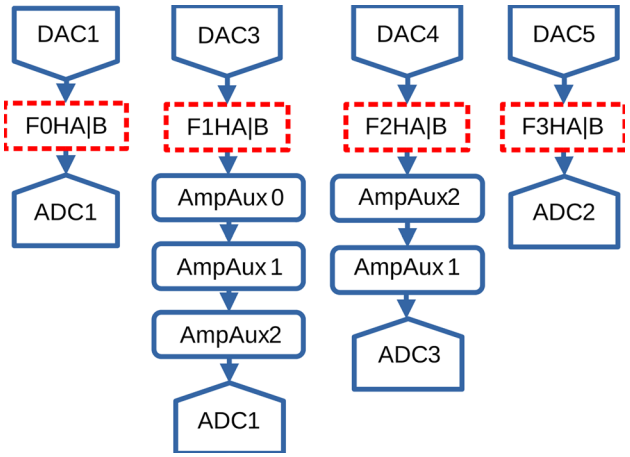

Fig. 5 Structures needed to test horizontal filters

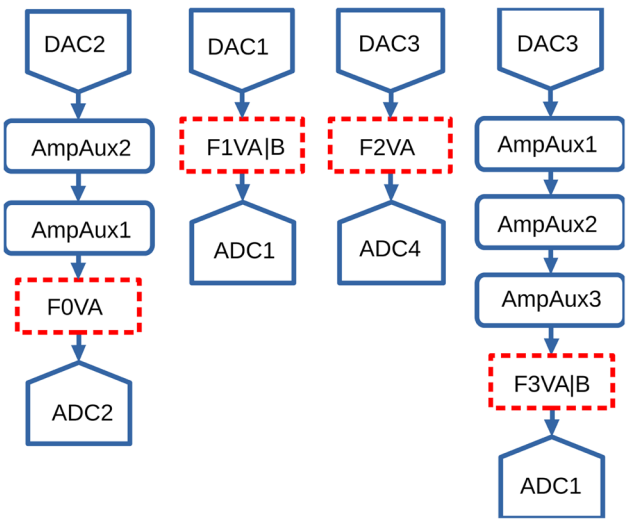

Fig. 6 Structures needed to test vertical filters

The filters able to be implemented in the device use different chip resources (Table 1). Therefore, the CASs available for implementing the DACs and ADCs differ according to the filter under test. For instance, if we consider the filter named F0HA as the CUT, it would be impossible to use ADC2 and DAC3 for the test because both employ the block ASD11 (see Table 2), also necessary to implement the filter. Additionally, it is necessary to use extra blocks to drive the test signals to the CUTs and bring their responses to the ADCs because the programmable interconnection network exhibits limitations. Consequently, each possible filter under test requires an ad-hoc test layout and resources, leading to a particular test solution that must be experimentally evaluated.

After considering different alternatives, the resulting schemes for testing horizontal and vertical filters are depicted in Figs. 5 and 6, respectively. In the figures, red boxes are the CUTs, test stimuli generators are denoted as DAC1 to DAC5, and the ADCs as ADC1 to ADC4. The arrows represent the interconnection resources needed, while AmpAux1 and AmpAux3 are programmable gain amplifiers (PGAs). The selection of PGAs as intermediate modules to overcome the limitations of the interconnection network relies on the fact that exists an SWBT method for these modules, previously reported in [18]. This approach can be easily incorporated into the test routine presented here.

3.4 Strategy for the measurement of the TRAM parameters

The use of ADCs for measurement limits the maximum frequency of the filters under test because each ADC has a maximum sample rate of 7812 Hz, becoming a limit incompatible with our proposal. To overcome this problem, we propose to reduce the sampling frequency of the filters during the test phase, taking advantage of their discrete-time nature. Under these conditions, the gain and OS% remain unchanged because they are independent of the sampling frequency. However, the value of Tp is obtained from the following expression:

$$Tp = Tp_{meas} \cdot \frac{fs_{meas}}{fs_{nom}} \tag{5}$$

As observed in (5), the actual value of Tp is the measured peak time (Tp_{meas}) at the frequency fs_{meas}, which is lower than the nominal frequency of the filter (fs_{nom}).

We use CDSs to generate the clock signal for the filter under test to provide an additional degree of flexibility with no hardware overhead. For this task, a pulse width modulator (PWM) embedded in the DCSs generates the clock signal for the filter with four times the frequency $fs_{meas,}$ a setting required by the vendor [15]. Additionally, to synchronize the measurements with fs_{meas}, we also implement (in the CDSs) a divider by four enabled by the test stimulus generator. The signal, with frequency fs_{meas}, generates an interruption that makes the test routine running in the processor (μC) capture the ADC output. Figure 7 depicts a conceptual diagram that illustrates the interconnection between the referred blocks.

3.5 Test stimuli configuration

Depending on the filter locations and the operating temperature, the filters exhibit an output offset as high as 640 mV [19]. To mitigate its effects in the test process and optimize the utilization of the dynamic range of the ADCs, we include in the BIST a routine to configure the test stimuli. This routine runs before the test measurements.

The input step signal is generated as a high to low level transition for obtaining a positive excursion at the output

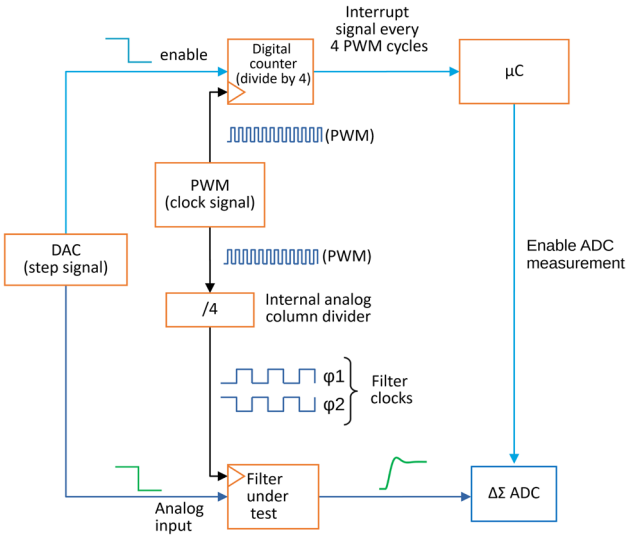

Fig. 7 Generation of signals in test mode

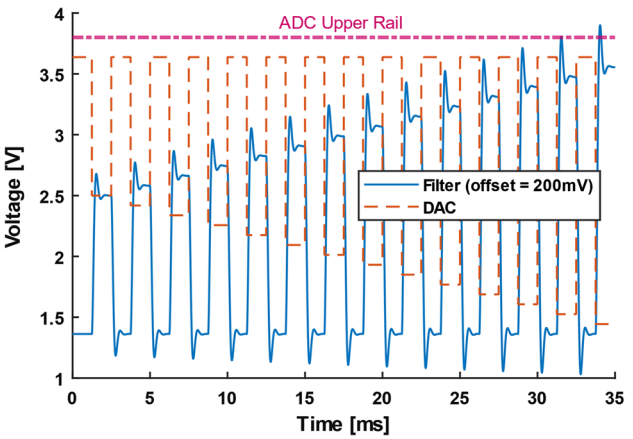

Fig. 8 Simulation of a DAC autoconfiguration process – seek of the input step final state

because the gain of the filters is negative. The high level of the input step is determined by incrementing the input of the DAC until the value reported by the ADC that measures the filter output reaches zero. Then, the algorithm selects the previous DAC state as the initial value of the stimulus.

The search process for determining the DAC's final value is similar, with the difference that the complete filter output transient response must be measured to ensure that the ADC's input does not saturate. The routine uses the high value previously determined during the exploration and modifies the low one. Figure 8 illustrates the concept, showing the search for the final value. In this figure, the filter output rises with a decrement in the low value of the DAC. The process continues until the maximum value admitted by the ADC is surpassed. Then, the value chosen is the previous one. The procedure repeats for each filter to be tested.

To accurately determine the voltage produced by the DAC, we resort to a different design layer, which has the DACs outputs directly connected to a 14 bits ΔΣ ADC without any intermediate block for the measurement to diminish signal degradations and noise. The values obtained in this procedure are adopted to determine the test parameters.

3.6 BIST operations summary

The operations made by the complete routine are depicted in Fig. 9. After setting the test stimulus, the device is reconfigured in test mode, adopting one of the configurations depicted in Figs. 5 or 6. The duration of the step is programmed to be at least four times the nominal settling time of the filter (at 1% of the final value). As mentioned above, the filter's response is measured using the corresponding ADC.

The nominal filter sampling frequency (fs_{nom}) is reduced to $fs_{meas,}$ as depicted in Section D. The signal from the DAC excites the filter and triggers the counter that generates the interrupt for acquiring the filter ADC's output.

A peak detection routine establishes the maximum value of the ADC. The sample numbers corresponding to the maximum value are also recorded. The value of Tp_{meas} is then calculated as the average of the sample numbers.

Next, the test measures the filter's response at four times the nominal settling time to get the V_{end} value, with the filter operating at fs_{nom}.

The routine measures T_{pmeas}, $V_{peak,}$ and V_{end} N times and then averages their values in the last step, when the µC calculates the functional parameters and transfers the results to an output port with a protocol specified by the user.

4 Experimental results

4.1 Case of study

We adopt the second-order lowpass filter previously reported in [12] as the test vehicle to facilitate result comparisons. Table 3 shows its main characteristics.

4.2 Determination of the test stimulus amplitude

The routine performs 100 consecutive measurements, using the values obtained in the self-calibration process, to establish with reasonable precision the amplitude of the input step delivered by the DACs. These measurements show low dispersions with a maximum value for the relation (standard deviation)/average or variation coefficient (VC) of 0.0126% for DAC4. Table 4 shows the average of our measurements

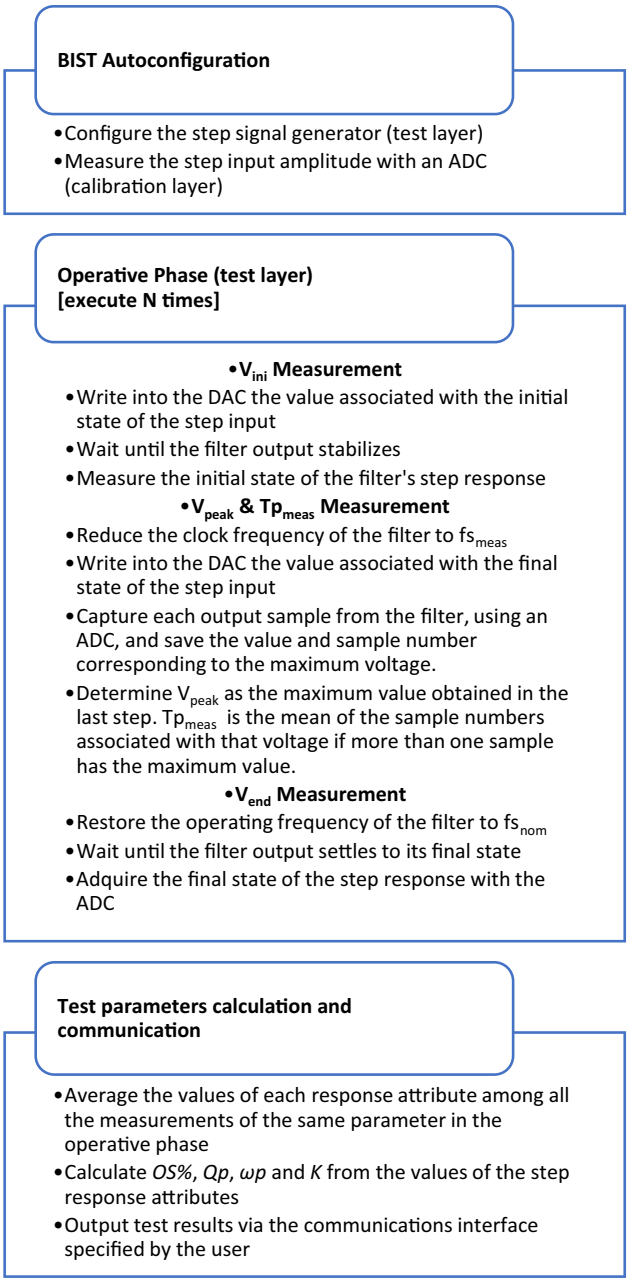

Fig. 9 Operations sequence of the test proposal

Table 3 Specifications of the filters under test

Feature	Value
Cut frequency − 0 dB cross (Hz)	1990.0
− 3 dB frequency (Hz)	2552.5
Bandpass ripple (V/V)	1.14
DC Gain (V/V)	1.0
ω_p (rad/s)	12,685
Qp	0.98
Sample frequency (kHz)	200

Table 4 DACs output amplitude

DAC	Theoretical step amplitude [V]	Measured step amplitude [V]	Relative difference
DAC1	1.991	1.984	0.33%
DAC2	1.991	1.982	0.42%
DAC3	2.194	2.181	0.57%
DAC4	2.072	2.064	0.37%
DAC5	2.194	2.184	0.45%

Table 5 Statistics for the parameter K (V/V)

Filter	Mean	Maximum	Minimum	VC
F0HA	1.00	1.00	1.00	0.02%
F0HB	1.01	1.01	1.01	0.02%
F0VA	0.99	0.99	0.99	0.03%
F1HA	1.00	1.00	1.00	0.02%
F1HB	1.00	1.00	1.00	0.02%
F1VA	1.00	1.00	1.00	0.03%
F1VB	1.00	1.00	1.00	0.03%
F2HA	1.00	1.00	1.00	0.01%
F2HB	1.01	1.01	1.01	0.01%
F2VA	1.00	1.00	0.99	0.02%
F3HA	1.00	1.00	1.00	0.04%
F3HB	1.00	1.00	1.00	0.04%
F3VA	1.00	1.00	0.99	0.06%
F3VB	0.99	0.99	0.99	0.07%

and the expected theoretical value according to the transfer function of the DAC [20]. The table shows that the measured values are very close to the theoretical ones.

4.3 Test scheme performance characterization

Our characterization procedure repeats the test of each filter 100 times to determine the stability of the on-chip measurement scheme (measurements repeatability). The results obtained for the parameters K, Qp, and ω_p, are reported in Tables 5, 6, and 7, respectively. These tables show the average as a measure of central tendency, VC as a measure of dispersion, and the minimum and maximum obtained values. These results show good repeatability, with maximum VC values for K, Qp, and ω_p of 0.07%, 0.14%, and 0.19%, respectively. In [12], the DfT scheme obtained VCs of 0.147%, 0.14% and 0.346% respectively.

Regarding the test time, our proposal obtains an average of 2.906 s for the test process of a single filter, including calibration and communication of test results (%OS, Tp, K, Qp, and Wp) through a serial port. This time is reasonable for applications that can stop their regular system operation for doing the test. However, it should be remarked that the

Table 6 Statistics for the parameter Qp

Filter	Mean	Maximum	Minimum	VC
F0HA	0.98	0.98	0.98	0.06%
F0HB	0.98	0.98	0.98	0.05%
F0VA	0.98	0.98	0.97	0.06%
F1HA	0.98	0.98	0.98	0.05%
F1HB	0.98	0.98	0.98	0.06%
F1VA	0.98	0.98	0.97	0.06%
F1VB	0.97	0.98	0.97	0.07%
F2HA	0.98	0.98	0.98	0.03%
F2HB	0.98	0.98	0.98	0.04%
F2VA	0.98	0.98	0.98	0.07%
F3HA	0.98	0.98	0.98	0.14%
F3HB	0.98	0.99	0.98	0.10%
F3VA	0.97	0.98	0.97	0.11%
F3VB	0.98	0.98	0.97	0.09%

Table 7 Statistics for the parameter ω_p (rad/s)

Filter	Mean	Maximum	Minimum	VC
F0HA	12,952	12,978	12,930	0.07%
F0HB	12,920	12,982	12,861	0.19%
F0VA	13,130	13,154	13,107	0.08%
F1HA	12,862	12,889	12,830	0.10%
F1HB	12,956	12,990	12,918	0.12%
F1VA	12,979	13,023	12,919	0.15%
F1VB	12,985	13,030	12,940	0.15%
F2HA	12,860	12,881	12,841	0.06%
F2HB	12,997	13,033	12,971	0.10%
F2VA	13,006	13,036	12,967	0.09%
F3HA	13,074	13,108	13,035	0.11%
F3HB	13,072	13,103	13,046	0.09%
F3VA	12,913	12,946	12,868	0.14%
F3VB	13,098	13,123	13,066	0.09%

test time strongly depends on the transient characteristics of the CUT, being higher for those with higher settling times. Additionally, the modest performance of the processing core of PSoC1 that precludes fast calculations negatively impacts the test time. Thus, an alternative for possible improvements is to consider using a faster processor for determining the functional parameters. This option is viable if PSoC1 is part of a more extensive system with other processors.

4.4 Comparison with other measurements

The user cannot access the internal nodes for performing measurements at the (internal) filter output. Instead, the filter response goes through the interconnection networks to the analog pins, altering the signal path. This fact makes estimating the BIST accuracy against a reference value (the measurement at the filter output) difficult or even impossible. An alternative to face this drawback is to compare the BIST results against similar determinations (indirect) or against theoretical values to provide enough confidence in the test scheme's measurements.

Table 8 BIST results against oscilloscope measurements

Filter	K	Qp	ω_p
F0HA	**3.36%**	0.14%	0.67%
F0HB	3.26%	0.06%	0.27%
F0VA	2.25%	0.55%	1.26%
F1HA	3.18%	0.56%	0.80%
F1HB	1.49%	0.54%	0.04%
F1VA	2.77%	0.00%	0.70%
F1VB	2.52%	0.04%	0.05%
F2HA	1.92%	0.25%	0.47%
F2HB	1.94%	0.20%	1.49%
F2VA	1.59%	0.07%	0.70%
F3HA	1.68%	0.14%	**2.02%**
F3HB	1.29%	**0.82%**	1.29%

Table 9 BIST results against theoretical values

Filter	K	Qp	ω_p
F0HA	0.66%	0.03%	2.10%
F0HB	0.85%	0.09%	1.85%
F0VA	0.28%	0.45%	**3.51%**
F1HA	0.93%	0.03%	1.40%
F1HB	0.81%	0.06%	2.14%
F1VA	0.56%	0.42%	2.32%
F1VB	0.51%	**0.55%**	2.37%
F2HA	0.64%	0.18%	1.38%
F2HB	**0.95%**	0.19%	2.46%
F2VA	0.05%	0.13%	2.53%
F3HA	0.56%	0.17%	3.07%
F3HB	0.13%	0.37%	3.05%
F3VA	0.09%	0.52%	1.80%
F3VB	0.59%	0.40%	3.25%

Table 8 shows the error for the average value of the BIST measurements relative to the average of the measurements performed with an oscilloscope (using a PGA as a test stimulus generator) [12]. This and the following tables highlight the highest deviations observed for each parameter in bold. In Table 8, the worst case is 3.36% for the K parameter.

Our characterization procedure also compares the BIST results with MatLab filter simulations at a transfer-function level. By using the capacitor values obtained in the design process, these simulations allow obtaining the nominal or theoretical functional parameters. Table 9 reports the results obtained from this comparison. This table shows the error of the average of each parameter relative to the value obtained

Table 10 Errors of BIST against frequency response measurements

Filter	K	Qp	ω_p
F0HA	0.16%	1.14%	1.39%
F0HB	0.15%	1.20%	1.19%
F3HA	0.70%	**1.56%**	**2.69%**
F3HB	0.29%	1.17%	2.35%
F0VA	0.61%	0.65%	2.28%
F3VA	0.88%	1.42%	1.04%
F3VB	**1.35%**	1.41%	2.44%

in MatLab. The worst-case was obtained for ω_p, reaching a deviation of 3.51%.

To get additional characterizations, we implemented a comparison with the frequency response of the filter experimentally gathered in the laboratory. This study is limited only to those filters that can be tested without resorting to additional internal blocks to reduce the possible sources of errors. For these comparisons, we consider the average of 100 BIST measurements. Table 10 reports the relative errors obtained by using both methods. As can be observed, the highest error is 2.69%, lower than 3.02%, presented in [12].

4.5 Comparison with previous work

This section discusses the results obtained via the proposed BIST and establishes differences with previous work.

The authors of [12] introduce a DfT scheme that requires external equipment and cannot operate autonomously. The BIST proposed here overcomes these limitations by using only the internal resources of the processor with zero hardware overhead, achieving better performance and higher repeatability than the previous DfT strategy.

The observation mentioned above deserves additional consideration. In [12], the signals to be processed are measured at pins located in a general-purpose development kit. For this purpose, the DfT scheme employs internal resources to drive the signals to processor analog pins and then to pins located in the board, using interconnections at the PCB level. These resources are susceptible to noise and distortion.

A typical output signal obtained under these conditions is shown in Fig. 10 (labeled as oscilloscope measurement). For comparison purposes, this figure also shows a typical output signal processed by the BIST scheme, measured with the internal ADC. This digital signal was reconstructed and superimposed to the former. The signal exhibits a low noise level because it is measured internally, with the filters operating at a lower sampling frequency. As can be deduced by comparing signals from DfT and BIST, the scenario for determining the test parameters is more favorable in the BIST scheme.

The authors in [12] compare their work with the reported one by Balen et al. [8] and show better performance. As the

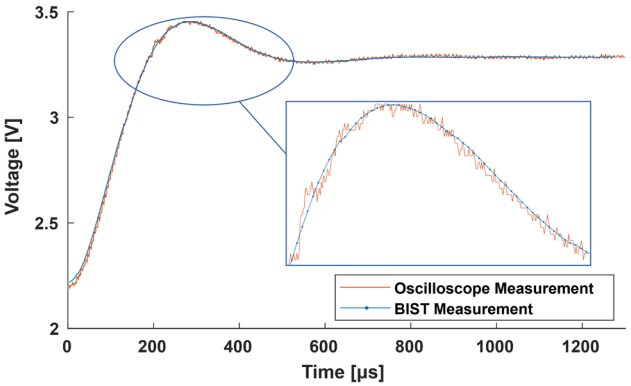

Fig. 10 Internal vs. oscilloscope measurement of the transient filter response

results of the BIST scheme presented here are better than [12], we avoid repeating the same comparison. However, the comparison may be unfair because the test solution [8] applies to a pretty different circuit with resources not comparable with PSoC1. Therefore, the authors of this paper consider that a sustainable comparison should be made between test schemes using similar resources and employed in comparable CUTs. However, The BIST proposed here has no precedents and is consequently very difficult to compare performances unquestionably.

Finally, a comparison with [13] is considered invalid because that work addresses the test of filters with a structural approach and demonstrates the method's suitability, but without providing an embedded test solution for any filter embedded in the analog array.

4.6 On-chip resources needed by the BIST proposal

The test requires two rows of digital configurable blocks and two to three analog columns (depending on the filter under test). However, the test resources are not reserved for test purposes. Instead, the device's dynamic reconfigurability allows rearranging these resources to any functionality required by the user when the test is not being executed. This way, the proposal has a zero-hardware overhead.

Nevertheless, the BIST has a memory overhead. The test routine reserves 11.5 KB of flash ROM (read-only memory), which represents 36% of the total nonvolatile memory available in the chip we used. This includes the configuration of the test modules, their libraries, and the floating-point (FP) library needed to calculate the test parameters. However, the ROM usage can be reduced to 6 KB if an external processor performs the floating-point operations. For example, this might be the case of larger systems where PSoC1 serves as a signal coprocessor.

Finally, the test software also needs 125 bytes of system RAM (random access memory) to run, which corresponds

to 6.1% of the total memory available in the chip (2 KB). This RAM space is reserved for implementing the variables of the BIST routine and cannot be freed after the execution of the test. However, these variables can be re-used by the user routines when the test is not running.

5 Conclusions

This work introduced a new BIST scheme (under a functional test approach) with zero hardware overhead that solves the problem of determining in-field the specifications of lowpass filters embedded in PSoC1 devices. The BIST we propose here can be implemented by the user, counting with the information and resources freely offered by the manufacturer. It just requires the user to specify the nominal specifications of the filter and its location within the resources array of the chip. This feature makes it very appealing for small and medium companies.

The experimental results showed excellent repeatability and low dispersions. What is more, the errors were minor than other techniques proposed for the same study case.

As future work, we will be evaluating an extension of the test scheme to make it compatible with bandpass filters and other reconfigurable platforms with similar resources to the one addressed here.

Funding This work was supported by Universidad Tecnológica Nacional and Universidad Nacional de Córdoba.

Data availability The datasets generated during and/or analyzed during the current study are available from the corresponding author on reasonable request.

Code availability The code generated during the current study is available from the corresponding author on reasonable request.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Consent for publication Not applicable.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

1. Suda, N., Suh, J., Hakim, N., Cao, Y., & Bakkaloglu, B. (2016). A 65 nm Programmable ANalog Device Array (PANDA) for analog circuit emulation. *IEEE Transactions on Circuits and Systems I: Regular Papers, 63*, 181–190.
2. Diab, M. S., & Mahmoud, S. A. (2020). Field programmable analog arrays for implementation of generalized nth-order operational transconductance amplifier-C elliptic filters. *ETRI Journal, 42*, 534–548.
3. George, S., Kim, S., Shah, S., Hasler, J., Collins, M., Adil, F., Wunderlich, R., Nease, S., & Ramakrishnan, S. (2016). A programmable and configurable mixed-mode FPAA SoC. *IEEE Transactions on Very Large Scale Integration Systems, 24*, 2253–2261.
4. Roel, D. D., & Kumar, M. (2020). Design and implementation of higher order sigma delta modulator circuits using FPAA. *Analog Integrated Circuits and Signal Processing, 104*, 169–182.
5. Rivera, J., Dueñas, I., Ortega, S., & Del Valle, J. L. (2018). Field-programmable analogue arrays for the sensorless control of DC motors. *International Journal of Electronics, 105*, 191–210.
6. Balen, T. R., Andrade, A. Q., AzaÏs, F., Lubaszewski, M., & Renovell, M. (2005). Applying the oscillation test strategy to FPAA's configurable analog blocks. *Journal of Electronic Testing, 21*, 135–146.
7. Laknaur, A., & Wang, H. (2005). A methodology to perform online self-testing for field-programmable analog array circuits. *IEEE Transactions on Instrumentation and Measurement, 54*, 1751–1760.
8. Balen, T. R., Calvano, J. V., Lubaszewski, M. S., & Renovell, M. (2007). Built-in self-test of field programmable analog arrays based on transient response analysis. *Journal of Electronic Testing, 23*, 497–512.
9. Pereira, G., Andrade, A., Balen, T. R., Lubaszewski, M., Azais, F., & Renovell, M. (2005). Testing the interconnect networks and I/O resources of field programmable analog arrays. In *23rd IEEE VLSI test symposium (VTS'05). IEEE Comput. Soc, Palms Springs, CA, USA* (pp. 389–394).
10. Lovay, M., Peretti, G., & Romero, E. (2015). Implementation of an adaptive filter using an evolvable hardware strategy. *IEEE Latin America Transactions, 13*, 927–934.
11. Laprovitta, A., Peretti, G., Romero, E., & Mourad, S. (2012). A low-cost configurability test strategy for an embedded analog circuit. *Microelectronics Journal, 43*, 745–755.
12. Dri, E., Peretti, G., & Romero, E. (2020). A low-cost test strategy based on transient response method for embedded reconfigurable filters. *International Journal of Electronics, 108*, 664–683.
13. Dri, E., Peretti, G., & Romero, E. (2021). Fault detection in configurable switched-capacitor filters using transient analysis and dynamic time warping. *Analog Integrated Circuits and Signal Processing, 108*, 291–304.
14. Calvano, J. V., Alves, V. C., Lubaszewski, M. (1999). Fault detection in systems with 2nd order dynamics using transient analysis. In *Proceedings. XII Symposium on Integrated Circuits and Systems Design. IEEE Comput. Soc, Natal, RN, Brazil* (pp. 110–114).
15. Van Ess, D. (2018) *AN2168—PSoC® 1—Understanding switched capacitor filters application note*. Retrieved from https://www.infineon.com/
16. Krstic, A., Lai, W.-C., Cheng, K.-T., Chen, L., & Dey, S. (2002). Embedded software-based self-test for programmable core-based designs. *IEEE Design and Test of Computers, 19*, 18–27.
17. Ogata, K. (2010). *Modern control engineering* (5th ed.). Prentice-Hall.
18. Lovay, M., Peretti, G., Romero, E., & Marqués, C. (2012). An adaptive amplifier system for wireless sensor network applications. *Journal of Electrical and Computer Engineering, 2012*, 1–14.
19. Infineon Technologies AG. (2013) T*wo-pole low-pass filter datasheet LPF2*. Retrieved from https://www.infineon.com/
20. Infineon Technologies AG. (2015) *6-Bit voltage output DAC datasheet DAC6*. Retrieved from https://www.infineon.com/

Emanuel Dri received the degree in Systems Engineering from the Universidad Tecnológica Nacional, Facultad Regional Concepción del Uruguay, in 2014. Currently, he is pursuing a doctoral degree in Electrical Engineering from the Universidad Tecnológica Nacional, Facultad Regional Córdoba. His current research interests include testing of mixed-signal circuits embedded in configurable devices and analog signal processing.

Gabriela Peretti received the degree in Electronic Engineering and Doctor in Engineering from Universidad Tecnológica Nacional, Argentina, in 1998 and 2006, respectively. She is a Professor at Facultad Regional Villa María, Universidad Tecnológica Nacional, and at Facultad de Matemática. Astronomía y Fïsica, Universidad Nacional de Córdoba, Argentina. Her main research interests are digital and mixed-signal test and design for testability.

Eduardo Romero received the degree in Electrical/Electronic Engineering from Universidad Católica de Córdoba in 1987 and the degree of Doctor in Engineering from Universidad Tecnológica Nacional in 2005. He is a Professor at Facultad Regional Villa María, Universidad Tecnológica Nacional, and at Facultad de Matemática. Astronomía y Fïsica, Universidad Nacional de Córdoba, Argentina. His main research interests are analog and mixed-signal test and design for testability.

Terms and Conditions

Springer Nature journal content, brought to you courtesy of Springer Nature Customer Service Center GmbH (“Springer Nature”).
Springer Nature supports a reasonable amount of sharing of research papers by authors, subscribers and authorised users (“Users”), for small-scale personal, non-commercial use provided that all copyright, trade and service marks and other proprietary notices are maintained. By accessing, sharing, receiving or otherwise using the Springer Nature journal content you agree to these terms of use (“Terms”). For these purposes, Springer Nature considers academic use (by researchers and students) to be non-commercial.
These Terms are supplementary and will apply in addition to any applicable website terms and conditions, a relevant site licence or a personal subscription. These Terms will prevail over any conflict or ambiguity with regards to the relevant terms, a site licence or a personal subscription (to the extent of the conflict or ambiguity only). For Creative Commons-licensed articles, the terms of the Creative Commons license used will apply.
We collect and use personal data to provide access to the Springer Nature journal content. We may also use these personal data internally within ResearchGate and Springer Nature and as agreed share it, in an anonymised way, for purposes of tracking, analysis and reporting. We will not otherwise disclose your personal data outside the ResearchGate or the Springer Nature group of companies unless we have your permission as detailed in the Privacy Policy.
While Users may use the Springer Nature journal content for small scale, personal non-commercial use, it is important to note that Users may not:

1. use such content for the purpose of providing other users with access on a regular or large scale basis or as a means to circumvent access control;
2. use such content where to do so would be considered a criminal or statutory offence in any jurisdiction, or gives rise to civil liability, or is otherwise unlawful;
3. falsely or misleadingly imply or suggest endorsement, approval , sponsorship, or association unless explicitly agreed to by Springer Nature in writing;
4. use bots or other automated methods to access the content or redirect messages
5. override any security feature or exclusionary protocol; or
6. share the content in order to create substitute for Springer Nature products or services or a systematic database of Springer Nature journal content.

In line with the restriction against commercial use, Springer Nature does not permit the creation of a product or service that creates revenue, royalties, rent or income from our content or its inclusion as part of a paid for service or for other commercial gain. Springer Nature journal content cannot be used for inter-library loans and librarians may not upload Springer Nature journal content on a large scale into their, or any other, institutional repository.
These terms of use are reviewed regularly and may be amended at any time. Springer Nature is not obligated to publish any information or content on this website and may remove it or features or functionality at our sole discretion, at any time with or without notice. Springer Nature may revoke this licence to you at any time and remove access to any copies of the Springer Nature journal content which have been saved.
To the fullest extent permitted by law, Springer Nature makes no warranties, representations or guarantees to Users, either express or implied with respect to the Springer nature journal content and all parties disclaim and waive any implied warranties or warranties imposed by law, including merchantability or fitness for any particular purpose.
Please note that these rights do not automatically extend to content, data or other material published by Springer Nature that may be licensed from third parties.
If you would like to use or distribute our Springer Nature journal content to a wider audience or on a regular basis or in any other manner not expressly permitted by these Terms, please contact Springer Nature at

onlineservice@springernature.com