Essential principles for practical analog BIST

Steve Sunter, Mentor Graphics - November 04, 2010

For almost 20 years, researchers and semiconductor manufacturers have been trying to develop a practical analog BIST (built-in self-test) for mixed-signal ICs. By enabling mixed-signal IC testing on digital testers and simpler multisite testing, this technique potentially reduces IC-test costs and time to market. Other anticipated benefits include faster test development and in-system self-test.

Most IC-design engineers understand the operation principles of digital BIST. It generates pseudorandom bit patterns with an LFSR (linear-feedback shift register) and applies them to a circuit under test using the flip-flops configured temporarily into serial shift registers. Digital BIST also captures responses using the same flip-flops, compresses the shifted-out result into a digital signature, and then bitwise-compares it to a correct signature. Although the details of industrial-logic BIST are more complex, the basic principles still apply for accommodating multiple clock domains, multicycle delay paths, and power-rail noise, for example.

The principles of analog BIST, however, remain enigmas to most engineers, especially analog-circuit designers. This fact became evident at the 2009 Design Automation Conference when an experienced PLL (phase-locked-loop)-circuit designer asked, “If your BIST for PLLs is so accurate, why don’t you design PLLs?” To answer that question, you must first understand the principles of practical analog BIST.

How do you define “analog”?

An “analog” circuit means different things to different people. You can consider a PLL or a SERDES (serializer/deserializer) to be digital-, analog-, or mixed-signal. BIST tests for these units can be purely digital because these functions have only digital inputs and outputs. For example, some ICs measure their PLL’s output frequency with an on-chip frequency counter—which counts the number of oscillation cycles in a known number of cycles of a reference frequency—that fails the test if any bit in the count differs from what you expect. Many ICs test their SERDES transceiver’s performance by looping back pseudorandom data and failing it if it detects a bit error. However, testing analog circuits, such as ADCs or DACs, clearly requires BIST circuitry that can generate or capture analog signals—signals whose instantaneous voltage is always relevant. Traditional analog circuits, such as filters and linear voltage regulators, have analog inputs and outputs, although many have digital-control signals or clocks. The purest analog circuits, such as RF circuits, may have no digital aspects at all.

In testing, an analog circuit has at least one signal whose instantaneous voltage is nondeterministic. Testing comprises checking that the signal, which might be a digital word, is between two voltages, digital values, or time thresholds; that a signal statistic is between limits; or that a mathematical computation involving the signal produces a value between limits. You should apply analog-test
principles to all circuits that have any analog signals.

Responses from purely digital circuits are deterministic, so an acceptable output signal is one that you need to sample only once. If you look at digital-circuit signals in enough detail, such as millivolts or picoseconds, however, every circuit is analog. This consideration is nontrivial for nanometer CMOS processes in which power-rail noise, jitter, temperature, and parametric variations are significant effects relative to 1V power rails and subnanosecond clock periods. BIST circuitry that tests analog is subject to these effects, even if the BIST is almost entirely digital, which is the reason that many analog designers wonder how any analog BIST can be more accurate than an analog circuit under test on the same chip.

The challenge of designing analog BIST

Designing BIST for analog circuits involves more than accurately delivering and capturing analog signals. The variety of signals and parameters requiring testing is much larger than that of the logic zeros and ones that digital BIST handles. Analog stimuli and responses can range from dc voltages, linear ramps, and impulses to sine waves and frequency modulation. To complicate the challenge, the stimulus and response might belong to different domains. For example, a dc voltage input might generate a frequency output. The parameters requiring analysis further add to the challenge because they may range from amplitude, phase delay, and SNR (signal-to-noise ratio) to dc voltage, peak-to-peak jitter, and duty cycle.

Test equipment usually must be an order of magnitude more accurate than the circuit under test. So the most daunting challenge for analog BIST is how to economically achieve greater accuracy than the circuit under test, which presumably has achieved the best possible accuracy for its silicon area and technology. The range of signal amplitudes can be enormous. ADCs and DACs handle on-chip analog signals with dynamic ranges as high as 2^24, equivalent to eight orders of magnitude.

You can compare digital BIST to a student who is grading his own multiple-choice test. The student places a template over the answer sheet and counts the number of correct answers. Analog BIST, on the other hand, compares with a student who is grading his own essay answers. It’s not a simple, objective procedure. Now that you understand the magnitude of the challenge, consider the fundamental circuit principles that you must apply for analog BIST to be practical.

Fundamental circuit principles

The first principle is that the test mechanism itself must be testable by applying timing-insensitive digital-test patterns, clocks, and dc voltages without requiring off-chip linear ac signals or measurements. ATE (automatic-test equipment) undergoes extensive calibration and testing before it leaves the factory. For BIST to be an alternative to using mixed-signal ATE, you must calibrate and test it before using it. The purely digital portions of analog-BIST circuitry should be testable using scan-based tests, including logic BIST. If the digital circuitry includes delay lines or delay-matched circuitry, then you should test the delays and delay increments. You can measure a delay by including or configuring the delay line into a ring oscillator and measuring the oscillation frequency using an on-chip frequency counter.

Testing a purely analog portion of analog BIST is more complex. Some researchers have proposed using an ADC or a DAC in their analog BIST, implicitly assuming that ATE can test it first; however, mixed-signal ATE would still be necessary, thus eliminating many benefits of BIST.

Perhaps the oldest BIST technique is to loop back a DAC output to an ADC input or a modulator
output to a demodulator input to permit an entirely digital test. This approach is analogous to using an untested circuit to test another and is insensitive to compensatory faults. For example, the DAC might have excessive nonlinearity for which similar nonlinearity in the ADC compensates, making the two together look better than either one alone.

The second principle of analog BIST is undersampling—sampling slower than the Nyquist rate, which means slower than twice the highest frequency of interest—which is necessary to permit slower analysis of a signal. Slower sampling also facilitates making the BIST circuitry smaller than the circuit under test.

In some self-calibration schemes, a low-speed ADC undersamples a high-speed ADC or DAC analog signal. First-order sigma-delta modulators are small, simple analog circuits that can convert analog signals into digital bit streams with arbitrary resolution if bandwidth decreases. A modulator might sample a signal 16 million times/second to produce 16 million 1-bit samples; the modulator can digitally filter these samples to produce 1 million 4-bit resolution samples/second or 16,000 16-bit samples/second, in each case decreasing the usable bandwidth by a factor of 16. Undersampling permits a narrower bandwidth of interest to center on the original signal’s frequency and allows its translation to a low frequency, at which it is easier to analyze. However, undersampling involves a trade-off of aliasing effects, which you must consider.

Another example of sampling is a PLL BIST that uses the PLL’s input-reference-clock edges to sample the PLL’s output (Figure 1a). In this case, a reference clocks a latch through an adjustable delay line, and the latch performs the sampling. The latch’s output counts for, say, 1000 clock cycles, and then the delay increments. This action repeats until the latch obtains the cumulative distribution function (b).

Another example of sampling is a PLL BIST that uses the PLL’s input-reference-clock edges to sample the PLL’s output (Figure 1a). In this case, a reference clocks a latch through an adjustable delay line, and the latch performs the sampling. The latch’s output counts for, say, 1000 clock cycles, and then the delay is incremented. This action repeats until the latch obtains the cumulative distribution function (Figure 1b). The PLL’s output frequency could be many times higher than its reference frequency. This BIST cannot detect jitter between reference-clock edges, but another technique that uses a slightly offset sampling frequency can sample at all points in the output phase (Figure 2).
These two techniques show an important principle of time measurement: Controlling the time at which a signal is sampled requires either a constant time offset from an adjustable delay or a constant frequency offset from an adjustable oscillator, such as a PLL. Low-jitter delays are increasingly more difficult to achieve in nanometer CMOS, but low-jitter frequency offsets are increasingly easier to achieve.

Another principle of analog BIST is subtracting systemic errors to improve accuracy. When measuring voltages, for example, you must cancel the offset voltage of any comparator or operational amplifier. If these circuits have negligible offset, then you must measure the offset to verify that it is negligible; otherwise, you must subtract its value. It is simpler to assume that the offset is significant and subtract it. When measuring delays, you must subtract the delay of the test-access path to the input of the circuit under test from the delay to the output to ensure that the access path’s delay cancels. ATE often uses both multiplication and subtraction to perform analog self-calibration, but this operation requires too much circuitry to be economical for BIST. Low-frequency effects can appear as fluctuating systematic errors, such as an offset that changes at 50 or 60 Hz because of power-line noise.

You can improve precision by adding samples to compute an average. Random noise in a signal or in the measurement circuitry limits the repeatability of any measurement of the signal’s properties. As you include more samples in a measurement, the test variance and repeatability improve. Analog-measurement circuitry typically accomplishes averaging with lowpass filtering or charge integration in a capacitor.

You can use full adders in analog BIST’s digital circuitry, but, in many cases, you can more efficiently accomplish averaging with binary counters. You cannot cancel noise that is not random, such as interference from nearby synchronous logic or 60-Hz power, by simple averaging or subtraction. You can, however, reduce its impact by sampling synchronously to the interference (Reference 1) or by integrating for an integer number of cycles of the interfering frequency.

To be cost-effective, BIST circuitry must have higher yield than the circuit under test. In the case of digital BIST, this requirement means simply that its area must be smaller than that of the circuit under test. For analog BIST, however, this principle also means that the BIST must achieve its required linearity, noise, and bandwidth without affecting yield. In a case study, only 70% of small analog-BIST circuits on a test chip achieved the required measurement accuracy. This BIST’s yield impact on an SOC (system on chip) would be the same as a circuit occupying 30% of the whole SOC.

The best way to implement BIST so that it has higher yield than an analog circuit under test is to minimize the amount of analog circuitry in the BIST—that is, make it digital. You can reduce the relative area of BIST circuitry by sharing one BIST circuit among many functions. Digital BIST can easily achieve this task, but analog BIST cannot because of the diversity of functions requiring testing. Such was the reasoning behind MadBIST, a scheme that MF Toner and Gordon W Roberts developed (Reference 2). In MadBIST, one DSP first tested an ADC and then a DAC. MadBIST, the ADC, and the DAC then tested other analog circuitry.

A problem with using a shared analysis block is conveying the analog signal of interest to the block. Analog busses commonly perform this task, but they introduce loading, noise, and nonlinearity, and
they reduce bandwidth. An alternative is to locally convert the signal to some form of digital representation and then use a digital bus.

Analog BIST must be able to apply specification-based structural tests. In other words, the applied stimulus and response analysis must permit correlation of the results with the analog circuit’s functional specifications, but they must also target manufacturing defects to facilitate diagnosis and minimize test time. Defect-oriented testing strives to accomplish this task but does not normally try to use functionlike tests. Philips (now NXP) in 1995 first performed one of the few published industrial comparisons between conventional specification-based analog testing and defect-oriented test (Reference 3). It concluded that defect-oriented test achieved faster testing for similar defect coverage when the design specifications had significant margin and the process was well-controlled. Otherwise, specification-based testing was necessary for maintaining test coverage and yield.

Digital BIST inherently applies a functionlike stimulus because almost any pattern of ones and zeros represents the input signals in function mode, including pseudorandom data. Delivering a functionlike stimulus to an analog circuit can be more complex. Pseudorandom noise is an enticing analog stimulus that addresses many potential defects and is easy to generate. A resistor and a capacitor can filter the output of the LFSR in digital BIST to produce an analog waveform. Multipliers and adders can cross-correlate the response of the analog circuit under test to its pseudorandom input.

Another more easily implemented approach is to reconfigure the circuit into an oscillator by connecting its output to its input, adding gain or inversion if necessary, and measuring the resulting oscillation frequency. The technique is area-efficient. Unfortunately for both these approaches, determining whether the circuit under test meets specifications has proved difficult because the measurement is too insensitive to noise and nonlinearity, and diagnosis is impractical.

ATE widely uses a linear ramp and a single-tone sine wave as test stimuli to efficiently test linearity and aid diagnosis of ADCs and DACs. The most robust way to generate a pure ramp or sine on-chip is to store a periodic sigma-delta bit stream in a circulating shift register, though this approach may require thousands of logic gates plus analog filtering. Fortunately, one stimulus block may be sufficient for all analog functions in an SOC and can efficiently convey the serial digital bit stream to all areas of a chip.

The easiest useful stimulus to generate is a digital square wave, which you can use to test a step or an impulse response. Surprisingly, an accurate dc voltage is a difficult stimulus or reference for a sampling comparator to generate unless you resort to analog techniques that require more self-test. Lowpass filtering of a programmable-duty-cycle digital waveform produces a mostly dc waveform for which the average voltage depends on the duty cycle and, at high switching frequencies, on the mismatch in the rise and fall times of the digital signal.

Reducing the switching frequency reduces the dc voltage’s sensitivity to this mismatch but increases the peak-to-peak variation of the dc voltage. In analog functions, such as voltage regulators, additional active lowpass filtering reduces this noise. Analog BIST using this approach must test the filtering, however. A technique that is more suitable for BIST was recently presented at the Workshop on Test and Verification of High-Speed Analog Circuits (Reference 4).

The last principle of analog BIST is that it must output its result as a digital measurement and its pass/fail bits resulting from comparison to upper and lower test limits. An analog-voltage result would be subject to corruption if you conveyed it off-chip for characterization and would require mixed-signal ATE. A digital result, without on-chip comparison to limits, would require the ATE to
capture and analyze digital words instead of single bits, preventing the use of the most common test-pattern languages, WGL (waveform-generation language) and STIL (Standard Test Interface Language), and many low-cost testers. A pass/fail result alone would prevent characterization of parameters and measurement repeatability, which is an essential step toward setting test limits.

**PL-BIST accuracy?**

Taking a look at these essential principles helps answer the PLL designer’s question. Practical PLL BIST uses neither analog circuitry nor delay lines, making it less sensitive to noise than the PLL under test. PLLs must generate a low-jitter edge every nanosecond, for example, and minimize jitter accumulation. However, PLL BIST can undersample edges using a pretested low-jitter clock conveyed through a few digital inverters that have fast transitions to minimize added jitter.

If a pretested clock is not available, then one PLL can sample the edges of another PLL on the same chip, operating at a slightly asynchronous frequency. The resulting jitter measurement is the sum of the two jitter levels; random jitters cannot cancel each other. Adding many such samples in a histogram reduces the impact of spurious noise, and sampling at the same rate as any interference can further reduce this impact.

**The need for analog BIST**

Few analog-BIST techniques that anyone has proposed in the past 15 years embody all of the principles noted here. Yet all of these principles are essential for the BIST to be practical and cost-effective. Developing a practical analog BIST has proved perhaps too challenging, but engineers will undoubtedly develop techniques that embody these principles because the need for them is increasing.

SOCs are including more system analog functions and higher pin and higher gate counts, all of which drive up tester costs and test times. The addition of embedded flash memories can greatly increase test time—to more than a minute—making multisite test essential, and this requirement drives the need for low-pin-count access and more analog test resources.

A significant roadblock that prevents adoption of analog BIST or any other new analog-test technique is the lack of an industry-accepted analog fault model. Fortunately, one outcome of a panel discussion at the 2009 International Test Conference ([Reference 5](#)) is that several of the panelists expressed interest in developing an IEEE-sponsored standard analog fault model. The panelists also agreed that more DFT (design-for-test) automation is necessary before the industry can adopt any new technique—a scenario that has happened for the digital portions of ICs. EDA companies can develop automation only when IC designers adopt systematic general techniques that can test many functions on an IC.

---

**References**


