

A Novel Method for Testing Digital to Analog Converter in Static Range

K. Hariharan, S. Gouthamraj, B. Subramaniam,
S.R. Venkatesh Babu and V. Abhaikumar

Department of Electronics and Communication Engineering,
Thiagarajar College of Engineering, Madurai, Tamil Nadu, India 625015

Abstracts: Problem statement: Linearity testing methods for DAC usually involves usage of non-linear analog components, which are indeed prone to various errors. Few other testing methodologies involve complex circuitry for measuring exactitude of DAC. Practically, it is difficult to build those as Built In Self Test (BIST) due to complexity of calculation, which demands more usage of ALU (or core of processing unit). This research aims to optimize and simplify the design of DAC testing scheme, while minimizing the computational overhead. Henceforth, the testing technique can be brought on to BIST level circuitry. **Approach:** A slope generator (more commonly known as integrator) produces a Ramp type of output voltage when it is fed with a DC voltage, slope of ramp depends upon the magnitude of DC-voltage. These varying slopes, when converted into a useful number, can provide some information, regarding voltage level of input. **Results:** In this research, we replaced the DC input of the Slope generator by analog output of DAC, which is under test. As the output of DAC varies according to the Digital code input, various slopes can be generated. These slopes are converted here into useful numbers called tick counts, by measuring the time taken by Ramp type output to cross a defined threshold voltage interval. The proposed method makes use of an integrator to produce a ramp signal of high precision and conditioned slope. The actual slope produced by the output of the DAC is compared with the expected slope by counting the number of clock ticks. **Conclusion:** This system of using Time Tick based BIST eludes the usage of high precision non-linear devices like ADCs to test DACs. Also this system reduces exigency of separate ALU for computing error.

Key words: Time tick BIST, DNL, INL, DAC error testing, DAC performance evaluation

INTRODUCTION

A novel test scheme for Digital-to-Analog Converter (DAC) is presented. Scientific and Industrial Instruments use data Converters like ADCs and DACs, which bridges the gap between digital computing unit and real world systems such as Computer Numerical Control machines (CNC). Systems using DAC depreciate as time proceeds, due to static error accumulation. When one such DAC is interfaced without calibration into any system it may lead to erroneous system response. For instance, a microprocessor based system for controlling cryogenic liquid flow may fail if such erroneous DACs are used. Hence there arises an exigency to test and calibrate DACs. Non-monotonic behaviors, offset error, gain error, Differential Nonlinearity (DNL) and Integral Nonlinearity (INL) (IEEE Xplore Press, 2009) are important specifications for testing DACs. BIST

approach is proposed to solve the above difficulty (Chen *et al.*, 2004). However, one major difficulty in testing these parameters is the requirement of high precision instruments to measure the very small output change under the change of the input code. The basic idea is to convert the DAC output voltages corresponding to different input codes into corresponding RAMP signals and further convert these RAMP signals to different time tick values. From the difference between Ideal and practically obtained ticks, evaluation parameters of a DAC, such as offset error, gain error, Differential Nonlinearity (DNL), Integral Nonlinearity (INL), could be effectively detected by simple digital circuits rather than complex analog or digital ones.

The existing technique to test DAC is to convert the DAC output voltages corresponding to different input codes into different oscillating frequencies through a Voltage Controlled Oscillator (VCO) and

Corresponding Author: K. Hariharan, Department of Electronics and Communication Engineering,
Thiagarajar College of Engineering, Madurai, Tamil Nadu, India 625015

further transferring these frequencies to different digital codes using a counter (Jiang and Agrawal, 2008a; 2008b; Wen and Lee, 1998). Other technique used far and wide is using ADC for testing DAC. The major drawbacks of these methodologies are the usage of non linear devices, Voltage Controlled Oscillator (VCO) and ADC. These non linear devices further lead to various other errors (Chang *et al.*, 2002; Huang *et al.*, 2000; Jiang and Agrawal, 2008b; Vargha *et al.*, 2001). The novelty of this method is to reduce the dependency on these non linear devices. Various evaluation parameters of DAC are discussed below:

DAC evaluation parameters:

Non-monotonic behavior testing: The scheme can easily test the non-monotonic fault of the DAC since, for the fault, the DAC will produce decreasing output voltage for an increasing input code. It can be easily detected by simply checking whether $D_{i+1} < D_o$ or not.

Offset error testing: Offset error is the difference between the ideal and actual DAC output values when the zero level digital input code is applied. It can be evaluated by: (unit: LSB):

$$\text{Offset error} = (D_o - D_{\min}) / TR$$

Gain error testing: Gain error is the difference between the measured output and the ideal output when a full-scale input code is applied. To make the gain error independent of offset error, offset error should be subtracted from the difference. It can be computed by: (unit: LSB):

$$\text{Gain error} = (D_2^n - D_{\max}) / TR - \text{offset error}$$

DNL testing: DNL is a measure of the deviation between the actual analogue output change and the theoretical change of 1 LSB. It can be evaluated by: (unit: LSB):

$$\text{DNL} (i) = (D_i - D_{i-1}) / TR - 1$$

INL testing: INL is defined by measuring the deviation of the actual converter output from the straight line of the ideal DAC transfer function. It is the cumulative effect, for any given input, of DNL and can be computed by: (unit: LSB) (IEEE Xplore Press, 2009; Jasper, 2007):

$$\text{INL}_i = \sum_{j=1}^i \text{DNL}_j$$

MATERIALS AND METHODS

The time-tick system proposed to test static errors in DAC is shown below (Fig. 1).

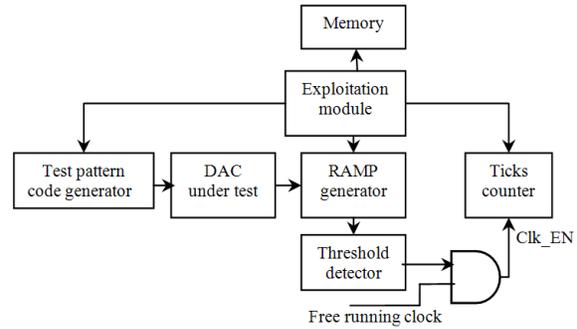


Fig. 1: System block diagram of test scheme for DAC testing

Test core module: Test core module includes:

- Code generator module
- RAMP generator
- Threshold detector
- Ticks counter
- Exploitation module

Detailed description of each of these is given below.

Code generator module: Test pattern code Generator provides the digital input data for DAC. The test pattern code generator produces digital output code on being given a signal by exploitation module. The counter is set to zero during the initiation of the test. On test initiation, the digital bin is given as the input to DAC. The digital data output of the Test Pattern Code Generator (TPCG) (Carni and Grimaldi, 2009) is incremented following the completion of ticks calculation for that data. The completion of the ticks calculation is recognized by the exploitation module and it instructs the counter to be incremented (Fig. 2).

RAMP generator: Ramp signal generator is implemented by means of an operational amplifier circuit operating as an integrator. Processing can be performed in the continuous-time (analog) domain or approximated (simulated) in the discrete-time (digital) domain. An integrator will have a low pass filtering effect but when given an offset it will accumulate a value building it until it reaches a limit of the system or overflows. Hence integrator can effectively used as a Ramp generator (Huang *et al.*, 2000). A switch is used to discharge the capacitor at the end of each ramp signal cycle. The ramp voltage at any time can be predicted by the equation:

$$V_o = -\frac{1}{C} \int_0^T \frac{V_{in}}{R} dt \quad (1)$$

when, the V_{in} is constant with fixed time period, the equation becomes:

$$V_o = -\frac{1}{C} \times \frac{V_{in}}{R} \times T \quad (2)$$

Threshold detector: Threshold detector (comparator) is implemented to detect the ramp from the integrator within two threshold ranges, used to determine time-ticks per code (Fig. 3).

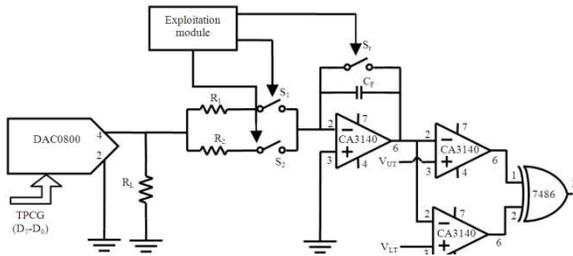


Fig. 2: Schematic of test core

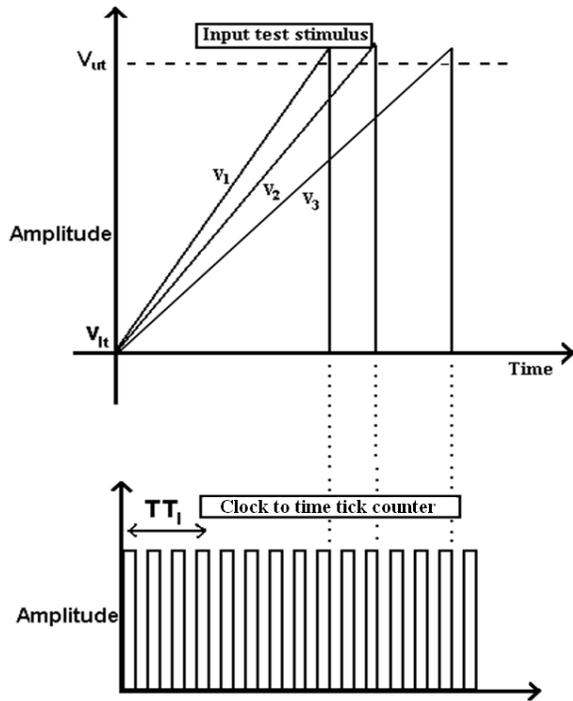


Fig. 3: Ticks value counted for various voltages. Here $V_1 > V_2 > V_3$

The lower comparator senses the ramp voltage when it traverses the reference Voltage (V_{LBT}). When the ramp traverses past the reference Voltage (V_{UBT}), the counter is disabled. The comparator outputs are connected to an EX-OR gate. The output of the EX-OR gate then acts as an active high enable for the counter. Thus the counter is enabled only when the ramp voltage is between the threshold voltage ranges.

Ticks counter: A counter of $(\log^2 (R \cdot (2^N - 1)))$ bits (where N-code width of DAC and R is resolution of the system) is used to count the number of ticks for the time period of the ramp between the two thresholds. The down counter is enabled during this period. The down counter output is supplied as clock for tick counter. The down counter is loaded with the dividing factor for the corresponding code input to the DAC. Dividing factors for each code is stored in memory. When carry over occurs in down counter, the ticks counter gets incremented once. Thus, the total number of counts per code is scaled down to meet the resolution condition of the ticks counter. The scaling factor depends upon the time taken by ramp to cross the threshold value. When ticks for the digital data have been computed it is transferred to memory. This is controlled by exploitation module. Dividing factor for each digital stage is given by formula:

$$D_x = \text{floor} \left(\frac{TT_{(max)}}{100 \times x^2} \right) \quad (3)$$

Exploitation module: After counting the tick value for a code, exploitation module discharges the capacitor used in ramp generator (Fig. 4). When both the comparator gives logic1 as output, the switch, used to discharge the capacitor, is powered ON by signal1. The same signal (signal1) passes through delay element.

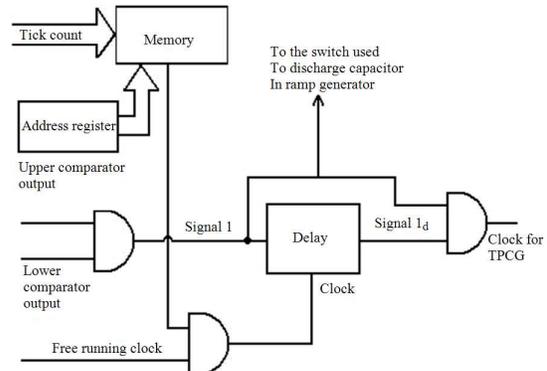


Fig. 4: Exploitation module

These two signals (signal1, signal1d) are given to an AND gate and output of this and gate acts as the clock for the test pattern code generator. TPCG generates next code. Meanwhile, the signal1 acts as write signal for memory. Tick count for each code is stored into memory. Address register gives consecutive address for the memory.

Methodology: The digital output of the TPCG is incremented once for every iteration. The binary data from TPCG is fed to DAC. Step size or the LSB value of a DAC analog output is given as:

$$1\text{LSB} = \left(\frac{V_{\text{REF}}}{2^N} \right) \tag{4}$$

This analog DAC output is fed to a ramp generator. Ramp generator converts this analog DAC output into linearly increasing ramp voltage. Ramp voltage has a fixed slope value for each analog voltage input which is given by formula:

$$\frac{dV_o}{dt} = \left(\frac{1}{RC} \right) \times V_{\text{in}} \tag{5}$$

This Ramp voltage is then fed to Ramp Threshold Detector (RTD) block which comprises threshold level detectors. The level detectors as explained already have individual threshold levels Upper bound Threshold Voltage (V_{UBT}) and Lower bound Threshold Voltage (V_{LBT}). When the ramp voltage crosses V_{LBT} , the time tick based counter is enabled. The counting process continues until the RTD provides valid output. The count value is inversely proportional to slope of the ramp signal, which in turn is proportional to the V_{out} from DAC under Test.

The count value is scaled using a preload down counter. The down counter is loaded with the pre-calculated dividing factor corresponding to that particular digital bin. The scaling value can be any integer value. The scaling factor for x^{th} LSB input is:

$$D_x = \text{floor} \left(\frac{TT_{(\text{max})}}{127 \times x^2} \right) \tag{6}$$

$TT_{(\text{max})}$ is the maximum number of ticks:

$$TT_x = \text{floor} \left(\frac{\text{count value}}{D_x} \right) \tag{7}$$

x is 0, 1,, (2^N-1)

The slope of the ramp signal for the first few steps of the DAC will be very low; hence the number of values counted by Time Ticks counter will be too high. On the other hand tick value to which it is scaled down is low. This in turn means that the dividing factor required for scaling the count values is too high. So in order to reduce the number of counted values, time and the dividing factor, two tick ranges are chosen.

Design of test parameters: No of ticks for a digital data input is:

$$TT_{(x)} = \frac{T}{T_{\text{clk}}} \text{Ticks} \tag{8}$$

Equation for output of the integrator is from Eq. 5:

$$\frac{dV_o}{dt} = \left(\frac{1}{RC} \right) \times V_{\text{in}}$$

Also the time taken for output of the integrator to change from a lower voltage V_{LT} to higher voltage V_{UT} is:

$$T = \frac{(V_{\text{UT}} - V_{\text{LT}}) \times RC}{V_{\text{in}}} \tag{9}$$

where $V_{\text{in}} = x \times \text{LSB}$ $x = 1, 2, 3, \dots, (2^n - 1)$.

The difference between upper and lower threshold level is selected to be 1 LSB:

$$\text{i.e., } (V_{\text{UT}} - V_{\text{LT}}) = 1\text{LSB}$$

The tick count is the ratio of time taken for the output of the integrator to change from a lower voltage V_{LT} to higher voltage V_{UT} to the time period of the clock:

$$TT_{(x)} = \left(\frac{(V_{\text{UT}} - V_{\text{LT}}) \times RC}{V_{\text{in}}} \right) \times \left(\frac{1}{T_{\text{clk}}} \right) \text{Ticks} \tag{10}$$

$$\therefore TT_{(x)} = \left(\frac{\text{LSB} \times RC}{x \text{LSB}} \right) \times \left(\frac{1}{T_{\text{clk}}} \right) \text{Ticks} \tag{11}$$

This on further simplification:

$$\therefore TT_{(x)} = \frac{RC}{x \cdot T_{\text{clk}}} \tag{12}$$

Assuming maximum available clock frequency to be 40 MHz, we get the time period of clock:

$$T_{\text{clk}} = \frac{1}{40 \times 10^6} \text{ s} \quad (13)$$

$$\therefore TT_{(\text{max})} = 40 \times 10^6 \times RCTicks$$

The number of ticks for $x \times \text{LSB}$ output from DAC is $(x \times 100)$.

Hence the obtained tick value has to be rounded of to some integral value. So we divide the obtained tick value by some value known as dividing ratio (D_x). The dividing ratio obtained is rounded off to its integer value:

$$D_{(x)} = \text{floor} \left(\frac{TT_{(\text{max})}}{100 \times x^2} \right) \quad (14)$$

Hence the tick count value finally obtained is:

$$TC_{(x)} = \text{floor} \left(\frac{TT_{(x)}}{D_{(x)}} \right) \quad (15)$$

In order to have two tick values we change the resistance used:

$$R = \frac{TT_{(\text{max})}}{C \times f_{\text{clk}}} \quad (16)$$

Assuming $C = 1 \mu\text{F}$:

$$f_{\text{clk}} = 40 \times 10^6 \text{ Hz}$$

We get:

$$R = 640 \ \Omega \quad \text{for } T_i = 25,600 \text{ ticks and}$$

$$R = 162.5625 \text{ K}\Omega \text{ for } T_i = 6,502,500 \text{ ticks}$$

Components used: Circuit construction was done in a separate PCB using components:

- DAC0800, CA3140 (Ramp and threshold detection) CD4066 for Switching and 74LS86
- Exploitation module, which controls and monitors entire operation of system, was described in Verilog and implemented in ALTERA DE1 FPGA Board

Implementation: The proposed Time Tick based test method is implemented to test the DAC0800 using ALTERA DE1 board. The proposed test scheme hardware is implemented in Verilog and the built code is loaded onto the FPGA as shown in Fig. 8. When the ramp input is applied to the DAC, the error values for each code are acquired into the SRAM. For

performance analysis of this method, the error values are stored in SRAM and then transferred to PC for further processing.

RESULTS

Non-linearity errors we computed using this technique. On comparing with the result obtained by conventional technique (checking the values of each code using high precision multimeter), we get similar results. Test data for first ten code indices of DAC are listed in Table 1.

DISCUSSION

Accuracy analysis: The accuracy of the test scheme is at least 0.01 LSB in each case which is five times greater than 0.05 LSB which is described in Chang *et al.*, 2002. The accuracy may even increase for certain codes but the system is designed to maintain a minimum of 0.01 LSB. The dividing factor calculated includes some floating point values. But usually a memory stores integer values. This means that the dividing factor for each step should be an integer i.e. it should be rounded off. Ticks should ideally be an integral multiple of 100. But practically it is not possible as we scale the dividing factor. So the accuracy that was mentioned earlier varies according to the curve plot shown in Fig. 5.

For instance the dividing ratio for code index 147 is actually 3.0123. As we floor it, we get dividing ratio as 3. There is not much a difference between two values. Hence the resolution here is 0.0099593 LSB which is approximately 0.01 LSB.

Let $TC_{id(x)}$ be the ideal number of ticks counts obtained and $TC_{p(x)}$ be the actual number of ticks counts obtained:

$$\text{Resolution} = \frac{x \times 100}{TC_{p(x)}} \times 0.01 \text{ LSB} \quad (17)$$

Table 1: Test results obtained for first 10 codes-through time-tick Bist method

Code index	Dividing ratio	Ideal tick counts	Obtained ticks counts	Difference	Error (MV)
1	128	100	92	8	0.0008
2	42	203	195	8	0.0008
3	21	304	294	10	0.0010
4	12	426	421	5	0.0005
5	8	533	526	7	0.0007
6	6	609	602	7	0.0007
7	4	800	792	8	0.0008
8	3	948	944	4	0.0004
9	2	1280	1272	8	0.0008
10	2	1163	1156	7	0.0007

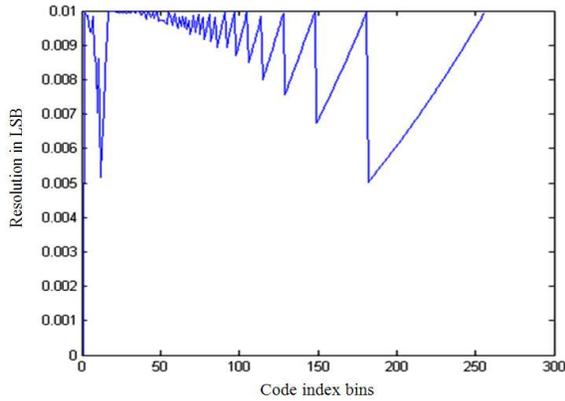


Fig. 5: Accuracy graph giving resolution Vs code

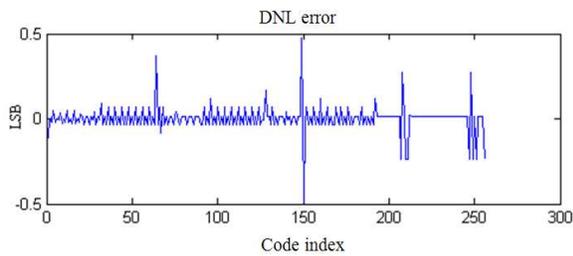


Fig. 6: DNL error calculated by conventional method

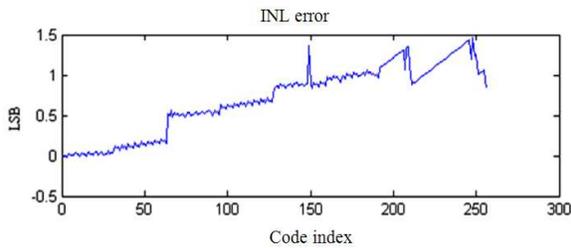


Fig. 7: INL error calculated by conventional method

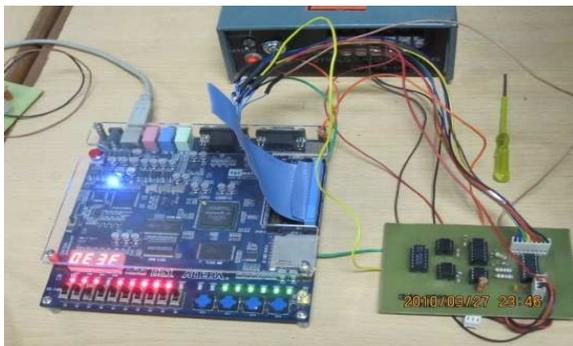


Fig. 8: Workbench setup for proposed test scheme

where, $LSB = \frac{V_{REF}}{2^n}$.

But for the next code (148), the dividing ratio is 2.9719 and the floor of the value is 2. Hence the resolution here is 0.00673LSB which is a greater resolution than 0.01 LSB. Thus, it can be inferred that the system maintains a minimum resolution of 0.01 LSB for all the test values.

Error obtained here is a function of difference between ideal number of counts and obtained number of counts, ideal counts and accuracy. It may be defined as:

$$Error = \frac{(TC_{p(x)} - TC_{id(x)})}{TC_{id(x)}} \times resolution (V) \quad (18)$$

The output of the DAC0800 was observed manually and the non linearity errors were plotted. Figure 6 shows the plot for Differential Non Linearity error (DNL) of the DAC under test. Figure 7 is the plot for Integral Nonlinearity error (INL) of the DAC under test.

Performance analysis: The plots below shows the error calculated in two different methods. The first plot shows the difference between the ideal voltage to be obtained and the voltage actually obtained. The second plot is the error calculated by time tick based methods. These entire plots are normalized with respect to LSB. The accuracy of the system is maintained at least 0.01 LSB. The plot below shows the DNL error for the output data taken manually of the digital to analog converter DAC0800. The output is processed by both the conventional methodology and time ticks based test scheme. The difference in error as calculated by two methodologies is given below.

This DAC is found to have the DNL errors obtained from the Time tick based test method as shown in Fig. 9. The corresponding INL error values are shown in Fig. 10 and the difference in errors obtained from the tick based method and that obtained from manual testing is given in Fig. 10-12.

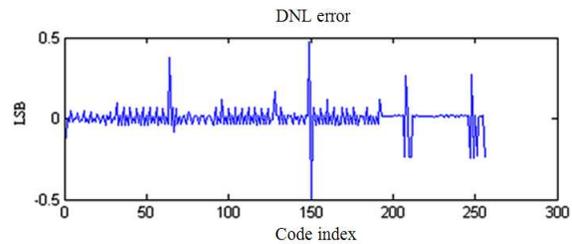


Fig. 9: DNL error calculated by time ticks based test scheme

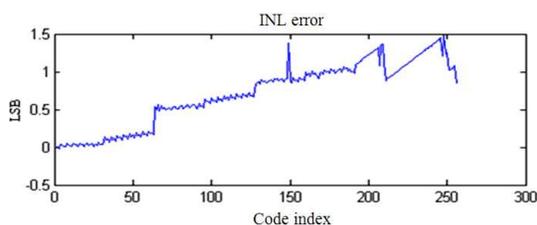


Fig. 10: INL error calculated by time ticks based test scheme

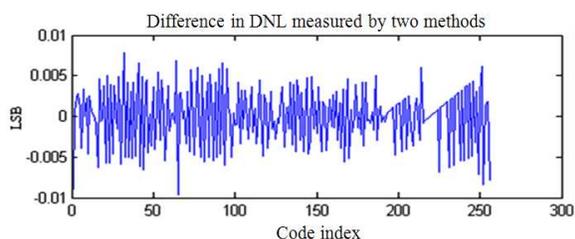


Fig. 11: DNL error difference between two methodologies

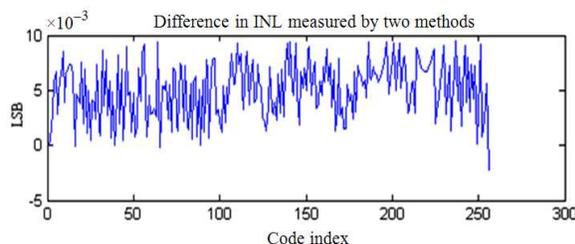


Fig. 12: INL error difference between two methodologies

It can be evidently seen that the values obtained from both the conventional and Time Tick based method seem to be approximately equal. Difference in values calculated from both the methods lies between 0.009 LSB at the maximum. So this method forms an alternative testing technique with comparatively good precision.

CONCLUSION

This test scheme will be executed every time when SoC starts up, to get up-to-date characteristics and errors of on-chip DAC. The time-tick based test scheme approach has been verified by simulation and shows significant improvements in effective error testing in noisy on-chip DACs. The main advantages are the proposed test scheme architecture does not require the existence of both AD and DA converters, which makes it feasible for most mixed-signal IC's. We show how the desired test accuracy can be achieved for a given hardware configuration and validate our ideas with numerical simulation results.

Our future work will be obtaining ideal time tick values to be a constant for all codes, thereby avoiding the usage of dividing ratio.

REFERENCES

- Carni, D.L. and D. Grimaldi, 2009. Comparative analysis of different acquisition techniques applied to static and dynamic characterization of high resolution DACs. Proceeding of the 19th IMEKO World Congress Fundamental and Applied Metrology, Sept. 6-11, IMEKO, USA., pp: 541-545. http://www.imeko2009.it.pt/Papers/FP_599.pdf
- Chang, S.J., C.L. Lee and J.E. Cheng, 2002. BIST scheme for DAC testing. *Elect. Lett.*, 38: 776-777. DOI: 10.1049/el:20020530
- Chen, G.X., C.L. Lee and J.E. Chen, 2004. A new BIST scheme based on a summing-into-timing-signal principle with self calibration for the DAC. Proceedings of the 13th Asian Test Symposium, Nov. 15-17, IEEE Xplore Press, USA., pp: 58-68. DOI: 10.1109/ATS.2004.10
- Huang, J.L., C.K. Ong and K.T. Cheng, 2000. A BIST scheme for on-chip ADC and DAC testing. Proceeding of the Conference and Exhibition on Design, Automation and Test in Europe, Mar. 27-30, IEEE Xplore Press, USA., pp: 216-220. DOI: 10.1109/DATE.2000.840041
- IEEE Xplore Press, 2009. Draft IEEE standard for terminology and test methods for digital-to-analog converters. IEEE. http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5291665
- Jasper, B., 2007. Practical telecom DAC testing. Test Edge Inc. http://www.analogchips.com/source/korean/body/custom/board/178_dactest.pdf
- Jiang, W. and V.D. Agrawal, 2008a. Built-in adaptive test and calibration of DAC. Proceeding of the IEEE International Test Conference ITC, Dec. 2008, IEEE, USA., pp: 1-6.
- Jiang, W. and V.D. Agrawal, 2008b. Built-in self-calibration of on-chip DAC and ADC. Proceeding of the IEEE International Test Conference, Oct. 28-30, IEEE Xplore Press, Santa Clara, CA., pp: 1-10. DOI: 10.1109/TEST.2008.4700638
- Vargha, B., J. Schoukens and Y. Rolain, 2001. Static nonlinearity testing of digital-to-analog converters. *IEEE Trans. Instrum. Measure.*, 50: 1283-1288. DOI: 10.1109/19.963198
- Wen, Y.C. and K.J. Lee, 1998. BIST structure for DAC testing. *Elect. Lett.*, 34: 1173-1174. http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=702351