Algorithms for ADC Multi-site Test with Digital Input Stimulus

Xiaoqin Sheng, Hans Kerkhoff
Dept. of EEMCS
University of Twente
Enschede, the Netherlands
{x.sheng, h.g.kerkhoff}@utwente.nl

Amir Zjajo, Guido Gronthoud
NXP Semiconductors
HighTech Campus 37
Eindhoven, the Netherlands
{amir.zjajo, guido.gronthoud}@nxp.com

Abstract— **This paper reports two novel algorithms based on time-modulo reconstruction method intended for detection of the parametric faults in analogue-to-digital converters (ADC). In both algorithms, a pulse signal, in its slightly adapted form to allow sufficient time for converter settling, is taken as the test stimulus reliving the burden placed on accuracy requirement of excitation source. The objective of the test scheme is not to completely replace traditional specification-based tests, but to provide a reliable method for early identification of excessive parameter variations in production test that allows quickly discarding of most of the faulty circuits before going through the conventional test. The efficiency of the methods is validated on a 6-bit flash ADC.**

I. INTRODUCTION

Multi-site testing is an effective approach to reduce production test time by increasing the parallel efficiency of testing multiple devices-under-test (DUT) on one tester. Nevertheless, for an ADC test, the increasing number of DUT in parallel usually requires high-quality analogue signal sources (such as ramp or a sine wave) [1] weakening the gains acquired with multi-site testing [2]. With advance of CMOS technology more and more data converters are integrated into platform-based designs, which are mostly used for video, audio and high-speed communication systems. The standardized architecture of these platforms is usually composed of memories, RF and mixed-signal front ends and importantly the multiple-processor cores, which offer the possibility of inexpensive on-chip digital waveform generation in its pure or slightly adapted form. In general, dynamic parameters of the A/D converter are evaluated through the conventional post-processing methods, such as histogram or FFT analysis, which exploit sine wave stimulus efficiency, i.e., negligible distortion and highly accurate and stable frequency. Employing similar methods to adapted pulse wave diminishes advantages of these methods. The spectrum of a pulse wave is distorted with harmonics related to the pulse rise and fall times, making accurate determination of A/D converter parametric faults complex and time excessive [3]. In this paper, we present two new algorithms for the post-processing using the pulse wave as the test stimulus.

Recently, several efforts have been made to further decrease the cost or the requirement of the accurate analogue stimulus generators for ADC testing. In [4], a white noise signal, which requires low analogue area overhead on chip, is used to estimate linearity of an ADC by the dynamic parameters. In [5], a staircase-like exponential waveform is used as the test input signal, which is generated by a pulse-width modulation (PWM) signal followed by an off-chip *RC* filter. With this method, the *3rd* harmonic distortion of an ADC up to 20-bits can be tested with a 3rd order polynomial fitting algorithm. The accuracy of the result is primarily limited by the linearity of the off-chip capacitor. The authors in [6] present an approach to simultaneously test the gain, offset, *2nd* and *3rd* harmonics and signal-to-noise ratio of ΣΔ ADCs. The results are obtained by only applying a fully binary PWM test signal and post processing the digital samples at the output. A self calibrated BIST architecture utilizing its on-chip processor is presented in [7]. It deals with two test methods that use the *RC* charging exponential waveform to test the static parameters of a high-resolution ADC. The first method relies on the static analysis to reduce the noise effects and test the gain, offset and linearity errors. The second one obtains the ADC static parameters by using an exponential curve fitting. The test stimulus is generated from a reference voltage, requiring four resistors and one capacitor. However, the generated waveform is quite dependent on process variations. The references [8-10] are continuous works with regard to the usage of two imperfect ramp signals with constant offset to test a high resolution ADC. A stimulus error identification and removal (SEIR) algorithm [8-10] is described for relaxing the linearity requirement of the test signal.

In this work, we give a preliminary solution for ADC multi-site testing. An ADC is tested by a pulse wave stimulus and the testing results are obtained by analyzing the output of the ADC in time domain. It is supposed to be a pre-test filtering out the faulty devices before the conventional testing. As the test stimulus and post-processing algorithm of this method are very suitable for multi-site testing, it has the high potential to reduce the ADC production test cost.

II. DETECTION OF A FAULTY ADC BY USING A PULSE WAVE SIGNAL

An adapted pulse wave can be expressed in time domain as:

$$\begin{aligned} x(t) &= \frac{A}{T_r} t(u(t) - u(t - T_r)) \\ &+ A(u(t - T_r) - u(t - (T_r + T_h))) \\ &+ A(1 + \frac{T_h + T_r - t}{T_f})(u(t - (T_r + T_h)) - u(t - T_r + T_h + T_f))) \end{aligned} \tag{1}$$

1530-1877/09 $25.00 © 2009 IEEE
DOI 10.1109/ETS.2009.17

IEEE computer society

Authorized licensed use limited to: UNIVERSITEIT TWENTE. Downloaded on January 28, 2010 at 03:41 from IEEE Xplore. Restrictions apply.

, where A, T_r, T_f and T_h denote the amplitude, rising and falling time of the signal and time of high-level, respectively as shown in Figure 1. The amplitude of the signal excites the A/D converter full input range, while the rising and falling time of the adapted digital stimuli are determined by the DUT bandwidth constrains [3]. In the previous work, we explored using an adaptive pulse wave to test ADCs in frequency domain. The spectrum of adapted pulse wave can be found as:

$$F\{x(t)\} = \frac{A}{\omega^2 T_r}(e^{-j\omega T_r} - 1) - \frac{A}{\omega^2 T_f}(e^{-j\omega T_h} - 1)e^{-j\omega(T_r + T_h)} - \frac{2jA}{\omega^2 T_r}\sin\omega\frac{T_r}{2}e^{-j\omega\frac{T_r}{2}} + \frac{2jA}{\omega^2 T_f}\sin\omega\frac{T_f}{2}e^{-j\omega(T_r+T_h+\frac{T_f}{2})} \quad (2)$$

, where $F_s=1/T_s$. One can see that the spectral representation of adapted pulse wave is not only a function of sampling frequency and amplitude of the signal, as for sine wave stimuli, but a periodic function of a pulse rising and falling times as well. When combined with non-linear response of the converter-under-test, well-controlled and accurate determination of converter's parametric faults through conventional methods become complex and time excessive [3]. In this case, one tries to obtain a signature result to distinguish the faulty from the fault-free devices by a pulse wave stimulus in time domain. The basic concept of our method is that by comparing the results' similarity between the digital outputs of the golden device and the DUT, one can detect the faulty devices from a large amount of DUTs in a multi-site test environment. A simple and fast pre-test can be carried out by this method before testing the specific dynamic and static parameters of the ADC. Most of the faulty devices can be discarded by this pre-test. Hence the number of devices, which are tested by the complicated and time-costly conventional test, will be reduced efficiently. As a result, it will reduce the test time when there is a large volume of DUTs to be tested.

In order to make the time-domain results clearer and easier to process, a technique of modulo time plot [11] is applied in both algorithms. It is a process that reorders all the sampling points of the output. As shown in Figure 1, after applying this algorithm, the output of the ADC is converted from several-periods pulse wave into one-period pulse wave. The reconstructed waveform shows errors of the ADC more visibly and intuitively [11].

A. Compare the deviation by using the amplitude of the ADC output

The overview flow of the algorithm is shown in the following table 1.

Initialization and data collection: As shown in Figure 1, a pulse wave test stimulus is applied to the golden devices, which are a collection of the examples of the fault-free devices, and the DUT respectively during a number of periods. To obtain the fault-free range later, all the corner cases (such as fast and slow cases) are required to be tested on the golden devices.

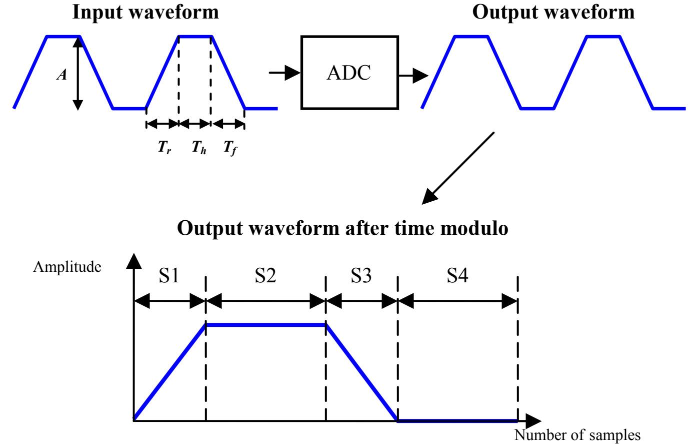


Figure 1: The reconstructed pulse wave of the ADC output

TABLE 1: The overview flow of the algorithm comparing the deviation by amplitude

Algorithm 1
Initialization
- Initialize the amplitude array *Am* of each sampling point
- Initialize the input stimuli
Data collection
- Collect *N* sampling points instants for each calculation
Main Body
1. Calculate the reconstructed signal according to [11]
2. Divide reconstructed output into four sections
3. Obtain the acceptable range of the output amplitude [$Am_{min}(i)$,
4. If $Am_{DUT}(i)>Am_{max}(i)$ obtain $\Delta Am(i)= Am_{DUT}(i)-Am_{max}(i)$
5. Increase the index, *i*, and repeat previous step for best estimate
6. If $Am_{DUT}(i)<Am_{min}(i)$ obtain $\Delta Am(i)= Am_{min}(i)-Am_{DUT}(i)$
7. Increase the index, *i*, and repeat previous step for best estimate
8. Calculate the out-of-amplitude-range percentage *P_am*

Step 1: The technique of time modulo plot [11] is used to reconstruct the output waveform. The output results, which are a pulse wave with a number of periods, are converted into a pulse wave with one period. From the output waveform after time modulo in Figure 1, one can see that the x-axis denotes the number of samples while the y-axis denotes the amplitude of the output.

Step 2: In Figure 1, the converted output pulse wave is divided into 4 sections. The starting sampling point and ending sampling point of each section are not required to be selected very accurately. However, the output of the golden devices and the DUT must be divided with the same starting and ending sampling points. The rising and falling edges are the most interesting part. The following steps are executed for each section respectively.

Step 3: For the result of each corner case of the golden device, an array of amplitudes *Am* can be obtained from each section of the reconstructed output. Each element *Am (i)* is the amplitude of one sampling point. By comparing the results of all the cases, one can obtain the maximum value $Am_{max}(i)$ and minimum value $Am_{min}(i)$ of every element *Am (i)*.

Authorized licensed use limited to: UNIVERSITEIT TWENTE. Downloaded on January 28, 2010 at 03:41 from IEEE Xplore. Restrictions apply.

Step 4-7: For the DUT, an array of the amplitude Am_{DUT} of the sampling points can be obtained from the reconstructed output. For each element $Am_{DUT}(i)$, it is verified whether it is within the range $[Am_{\min}(i), Am_{\max}(i)]$. If $Am_{DUT}(i) > Am_{\max}(i)$, the amplitude deviation $\Delta Am(i)$ is defined as:

$$\Delta Am(i) = Am_{DUT}(i) - Am_{\max}(i) \tag{3}$$

Similarly, if $Am_{DUT}(i) < Am_{\min}(i)$, we define the amplitude deviation $\Delta Am(i)$ as:

$$\Delta Am(i) = Am_{\min}(i) - Am_{DUT}(i) \tag{4}$$

For the case $Am_{\min}(i) < Am_{DUT}(i) < Am_{\max}(i)$, the amplitude deviation $\Delta Am(i)=0$.

Step 8: It is assumed that the total number of sampling points is N. The average out-of-amplitude-range percentage P_am of the whole curve is calculated as:

$$P_am = \frac{\sum_{i=1}^{N} \Delta Am(i)}{\sum_{i=1}^{N} Am_{\max}(i) - \sum_{i=1}^{N} Am_{\min}(i)} \tag{5}$$

It is employed to evaluate the faults in the ADC, which will be shown later on.

B. Compare the deviation by using the angle of the ADC output

An overview of the flow of the algorithm is shown in the following table:

TABLE 2: Overview of flow of the algorithm comparing the deviation by angle

Algorithm 2
Initialization
- Initialize the angle array $\angle\varphi$ of each sampling point
- Initialize the input stimuli
Data collection
- Collect N sampling points instants for each calculation
Main Body
1. Calculate the reconstructed signal according to [11]
2. Divide reconstructed output into four sections
3. Obtain the acceptable range of the angle deviation $[\angle\varphi_{\min}(i), \angle\varphi_{\max}(i)]$
4. If $\angle\varphi_{DUT}(i) > \angle\varphi_{\max}(i)$ obtain $\Delta\angle\varphi(i) = \angle\varphi_{DUT}(i) - \angle\varphi_{\max}(i)$
5. Increase the index, i, and repeat previous step for best estimate
6. If $\angle\varphi_{DUT}(i) < \angle\varphi_{\min}(i)$ obtain $\Delta\angle\varphi(i) = \angle\varphi_{\min}(i) - \angle\varphi_{DUT}(i)$
7. Increase the index, i, and repeat previous step for best estimate
8. Calculate the out-of-amplitude-range percentage $P\angle\varphi$

The initialization, data collection and the first two steps are completely the same as the previous algorithm. So we will not explain them in detail.

Step 3: As shown in Figure 2, the i-1, i and i+1 are three adjacent points on the output curve. If one connects two adjacent points i and i-1 with a straight line, then an angle $\angle\varphi(i-1)$, which is between the connected line and x-axis, is obtained. In this way, with a curve of N sampling points, an array of angles $\angle\varphi(1), \angle\varphi(2) \ldots \angle\varphi(i) \ldots \angle\varphi(N-1)$ can be obtained, which describes the deviation of the *trend* of a curve.

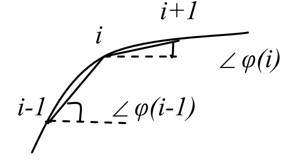


Figure 2: The sampling points on the output curve of the ADC

Similar to the Am, an array of angles $\angle\varphi$ can be obtained from each section of the output curve. By comparing $\angle\varphi(i)$ of all the corner cases, the maximum value $\angle\varphi_{\max}(i)$ and minimum value $\angle\varphi_{\min}(i)$ can be obtained for each element $\angle\varphi(i)$.

Step 4-7: An array of the angle $\angle\varphi_{DUT}$ of the sampling points from the DUT output can be obtained in the same way as in Step 3. For each element $\angle\varphi_{DUT}(i)$, it is verified whether it is within the range $[\angle\varphi_{\min}(i), \angle\varphi_{\max}(i)]$.

If $\angle\varphi_{DUT}(i) > \angle\varphi_{\max}(i)$, the angle deviation $\Delta\angle\varphi(i)$ is defined as:

$$\Delta\angle\varphi(i) = \angle\varphi_{DUT}(i) - \angle\varphi_{\max}(i) \tag{6}$$

For the case $\angle\varphi_{DUT}(i) < \angle\varphi_{\min}(i)$, we define the angle deviation $\Delta\angle\varphi(i)$ as:

$$\Delta\angle\varphi(i) = \angle\varphi_{\min}(i) - \angle\varphi_{DUT}(i) \tag{7}$$

If $\varphi_{DUT}(i)$ is within the range $[\angle\varphi_{\min}(i), \angle\varphi_{\max}(i)]$, the angle deviation $\Delta\angle\varphi(i)=0$.

Step 8 At the end, the average out-of-angle-range percentage $P\angle\varphi$ of the whole curve is found as:

$$P\angle\varphi = \frac{\sum_{i=1}^{N-1} \Delta\angle\varphi(i)}{\sum_{i=1}^{N-1} \angle\varphi_{\max}(i) - \sum_{i=1}^{N-1} \angle\varphi_{\min}(i)} \tag{8}$$

$P\angle\varphi$ is used to evaluate the faults in the ADC, which will be illustrated later on.

III. THE DEVICE UNDER TEST AND FAULT INJECTION

Both algorithms have been evaluated on the transistor-level design of 6-bit flash ADC illustrated in Figure 3. Flash ADC is by far the fastest and conceptually the most simple conversion process, where an analogue input is applied to one side of a comparator circuit and the other side is connected to the proper level of reference from zero to full scale. The threshold levels are usually generated by resistively dividing one or more references into a series of equally spaced voltages, which are applied to one input of each comparator. The total number of comparator required is 2^n-1, where n is the resolution of the A/D converter. The large number of comparators causes various detrimental effects: large die size which implies high cost, large device count leading to low yield, complicated clock and single distribution with significant capacitive loading, large input capacitance requiring high power dissipation in the S/H driving the A/D converter and degrading dynamic linearity,

Authorized licensed use limited to: UNIVERSITEIT TWENTE. Downloaded on January 28, 2010 at 03:41 from IEEE Xplore. Restrictions apply.

high power supply noise due to large digital switching current and significant errors in threshold voltages caused by comparator input bias current flowing through the resistive reference ladder. These factors make implementation of flash converters above eight bits very difficult, especially if low power dissipation is required. A S/H amplifier for sampling of the input signal is not a necessary component for the flash A/D converter, however, since the CMOS high-speed comparator usually contains a differential amplifier at its input, the insertion of a S/H amplifier in front of the comparator array can help avoiding improper signal racing among the differential amplifiers of the parallel connected high-speed comparators, reduce the input impedance, and increase the analogue bandwidth of the whole conversion system.

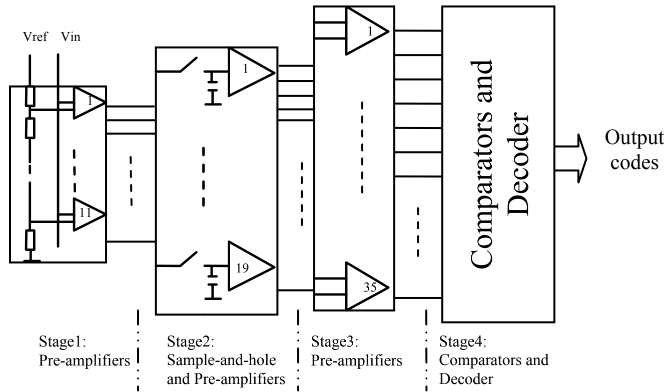


Figure 3: The block level diagram of the 6-bit flash ADC

The performance of a low-resolution flash A/D converter is limited primarily by the accuracy of the comparators and secondarily by the accuracy of the reference. To ease the problem of the large input capacitance, the difference between the analogue input and each reference voltage can be quantized at the output of each preamplifier, which is possible because of preamplifier finite gain (non-zero linear input range). This indicates that interpolating between the outputs of preamplifiers can increase the equivalent resolution of a flash stage [12]. The gain in the pre-amplifiers reduces the required accuracy and thereby the power consumption of the comparators.

In this work, we focus on the parametric faults in the analogue circuitry of the ADC. The input of the first stage is the original input stimulus, while the input of the following stages is amplified and divided into several resolution levels through resistor ladder. Similarly, any faulty behaviour in the first stage will be amplified and affect the following stage. As a result, the performance of the A/D converter is the most fault-sensitive in the first stage. In our case, we inject the parametric faults into the first stage, which is composed of 11 pre-amplifiers. The first stage pre-amplifiers are randomly chosen and injected by three type of faults respectively: *i)* offset fault - by inserting a dc voltage source into the gate of one of the input pair transistors, *ii)* gain fault - by varying the load resistor value of the amplifier and *iii)* bandwidth fault - by inserting an extra capacitor at the output of the amplifier.

IV. Simulation Results and Analysis

All simulations have been performed with an adapted pulse wave of an input frequency f_{in}=7MHz with 2.1ps jitter, a sample frequency f_s=300MHz, number of sampling periods M=70, amplitude A=0.46V, offset voltage V_{offset}=0, rising/falling time T_r / T_f=10ns, temperature T=25°C, and a power supply V_{DD}=1.2V. To limit the simulation time, the digital decoder is omitted from the analysis.

After executing the first three steps of the proposed algorithms, the reconstructed output waveform of the ADC can be obtained as shown in Figure 4. The x-axis denotes the number of sampling points while the y-axis denotes the amplitude of the output. There are 3002 sampling points in total. On both rising and falling edges there are around 220 sampling points. As the falling and rising edges are symmetrical and the results are quite similar, only the results of the rising edge are shown.

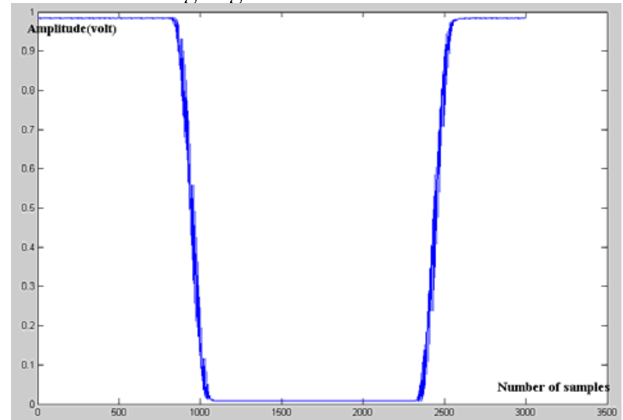


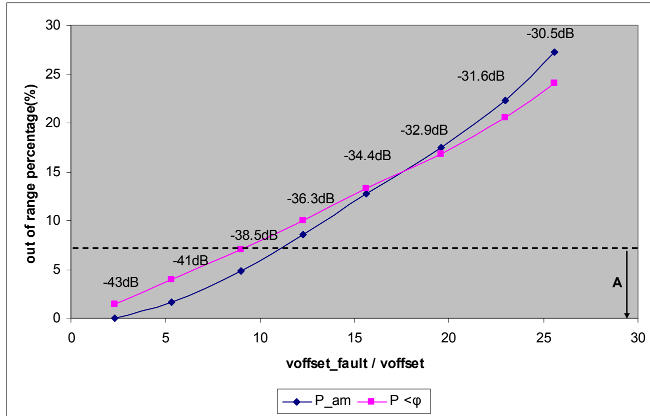
Figure 4: The reconstructed waveform of the ADC output

A. The ADC with offset faults

As explained in section III, the 6-bit flash ADC is injected with offset faults by inserting a voltage source to the gate of one of the input pair transistors. The two test algorithms in section II are applied to the DUT respectively and the results are shown in Figure 5. The x-axis denotes the ratio of the faulty offset voltage *voffset_fault* to the fault-free offset voltage *voffset*. The y-axis denotes the values of the out-of-range percentage from the two algorithms. The THD value of each faulty case is also shown in the figure, which is one of the most important conventional dynamic parameters for all types of ADCs. The fault-free range of the THD value is smaller than -38.7dB. From these results, one can see that if the ratio of *voffset_fault* to *voffset* changes from 9 to 25.6, the out-of-amplitude-range percentage changes from 0 to 27.3% and the out-of-angle-range percentage changes from 1.4%~24.1%. For both algorithms, P_am and $P\angle\varphi$ increase as the change of the offset becomes larger. When P_am and $P\angle\varphi$ increase to larger than 5% and 7% respectively, the DUT is out of the fault-free range of the THD. Moreover, when the ratio of *voffset_fault* to *voffset* changes from 2.33 to 9, it can not

Authorized licensed use limited to: UNIVERSITEIT TWENTE. Downloaded on January 28, 2010 at 03:41 from IEEE Xplore. Restrictions apply.

detect the fault from the THD value. However, P_am and $P\angle\varphi$ are not zero. This means that our method can detect the offset fault in a more sensitive manner than the conventional method of measurement of the THD value. Comparing the two curves in Figure 5, one can see that the curve of the first algorithm is slightly steeper compared to the second algorithm. This means that the first one is a little bit more sensitive to the offset fault deviation.



A is the fault-free range with regard to the THD value

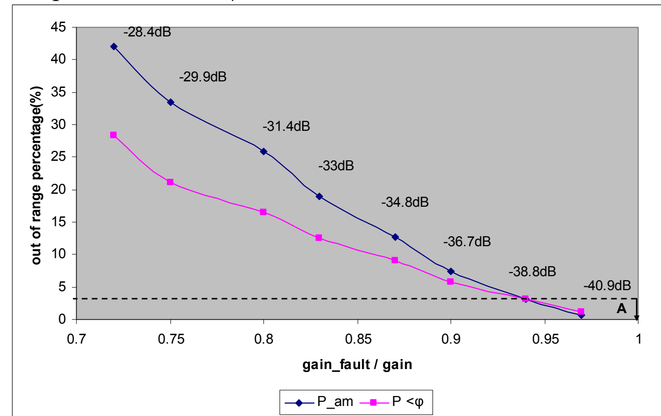
Figure 5: The results of P_am and $P\angle\varphi$ with offset faults in one single first stage pre-amplifier

B. The ADC with gain faults

When testing the ADC for gain faults, the results of the two algorithms are as shown in Figure 6. The x-axis denotes the ratio of faulty gain value *gain_fault* to the fault-free gain value *gain*. The y-axis denotes the out-of-range percentage by the amplitude or angle. The THD value for each faulty case is also shown in the figure as a reference. While the ratio of *gain_fault* to *gain* decreases from 0.97 to 0.72, P_am increases from 0.7 % to 42.1% and $P\angle\varphi$ increases from 1.1% to 28.3%. When P_am becomes larger than 7% or $P\angle\varphi$ is larger than 5%, the DUT is out of the fault-free range of the THD. This means that our algorithms can detect the gain fault, which can also be detected by the THD value. Furthermore, they can detect the gain faults when *gain_fault / gain* is from 0.94 to 0.97 while the THD value can not. So they are even more sensitive to the gain fault than the measurement of the THD value. In Figure 6, the curve obtained from the first algorithm is much steeper than the second one. Hence the algorithm by comparing the deviation of the amplitude is more sensitive as compared to the angle with respect to the gain fault deviation.

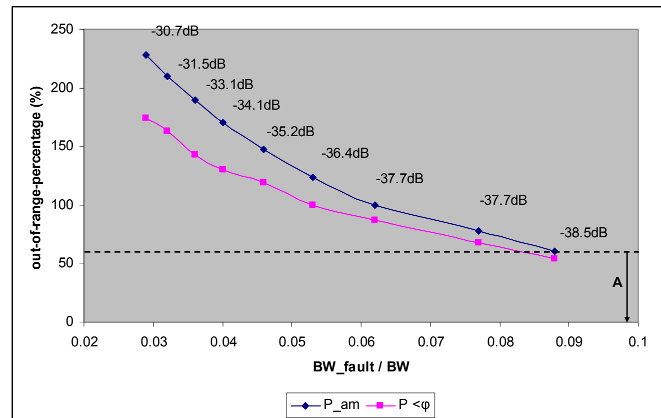
C. The ADC with bandwidth faults

The results of P_am and $P\angle\varphi$ with regard to bandwidth parametric faults are shown in Figure 7. The x-axis is the ratio of faulty bandwidth *BW_fault* to the fault-free bandwidth *BW* while the y-axis shows the values of the out-of-range percentage. From Figure 7, one can see that as the ratio of *BW_fault* to *BW* decreases from 0.077 to 0.029, the P_am increases from 77.9% to 228% and the $P\angle\varphi$ increases from 68% to 174%. When the THD value is increased by 1dB from the fault-free range due to the faulty bandwidth of 77MHz, P_am is as high as 77.9%, due to the high sensitivity of the rising/falling edge of the output to the bandwidth variations. From this result, one can also see that the P_am is more sensitive to the bandwidth deviation as compared to the $P\angle\varphi$.



A is the fault-free range with regard to the THD value

Figure 6: The results of P_am and $P\angle\varphi$ with gain faults in one single first stage pre-amplifier



A is the fault-free range with regard to the THD value

Figure 7: The results of P_am and $P\angle\varphi$ with bandwidth faults in one single first stage pre-amplifier

From the above results of the ADC injected with three different types of parametric faults, one can see that both of the algorithms of comparing the deviation of amplitude and angle can detect all the faults as well as the THD value, which is one of the most important conventional testing parameters of the ADCs. And they are even more sensitive to the faults than the THD values. Especially, in the case of bandwidth faults, it is more obvious.

V. Proposed Testing Method vs. Conventional Testing Method

1) Input test stimulus

The proposed testing method in this paper only requires a simple adapted pulse wave, which can be easily generated from the embedded processors in the case that the ADCs are integrated into a platform-based design. However, the conventional testing method requires an accurate analogue

Authorized licensed use limited to: UNIVERSITEIT TWENTE. Downloaded on January 28, 2010 at 03:41 from IEEE Xplore. Restrictions apply.

sine wave or ramp signal, which will result in silicon overhead if it is generated on-chip. Therefore the test stimulus of the proposed method is less expensive and more simple for a multi-site test environment.

2) Post-processing approach of the ADC output data

Compared to the proposed method, the post-processing method of the output data in a conventional test is much more complicated, like the FFT analysis. As a result, it requires much more time consumption and data processing power for calculation than the proposed method.

3) Testing result

Based on conventional mixed-signal tests, one can obtain all the accurate values of the dynamic or static parameters while one can only distinguish faulty devices from the fault-free devices by the proposed method. However, as its input test stimulus is very suitable for multi-site test environment and the post-processing method is much less time consuming, the proposed method can be a very quick and cheap pre-test. After that, most of the faulty devices can be discarded and only the devices, which pass the pre-test, take the complicated and time-costly conventional test. In this way, it can reduce the production test time and cost significantly.

4) Test time

We define the total test time of the conventional ADC test as:

$$T_c = t_{s_c} * \frac{N_{DUT}}{s_c} \tag{9}$$

, where t_{s_c} denotes the test time of conventional test for testing s_c sites, N_{DUT} denotes the total number of DUTs.

The total test time of the ADC test with the proposed pre-test can be defined as:

$$T_n = t_{s_p} * \frac{N_{DUT}}{s_p} + t_{s_c} * \left(\frac{N_{DUT} - N_f}{s_c} \right) \tag{10}$$

, where t_{s_p} denotes the time of pre-test for testing s_p sites, N_f denotes the number of faulty devices filtered out by the pre-test.

The difference between the total test time with and without pre-test can be calculated as:

$$T_c - T_n = t_{s_c} * \frac{N_f}{s_c} - t_{s_p} * \frac{N_{DUT}}{s_p} \tag{11}$$

Assuming $T_c > T_n$, (14) can be obtained as follows:

$$\frac{t_{s_c}}{t_{s_p}} * \frac{s_p}{s_c} > \frac{N_{DUT}}{N_f} \tag{12}$$

As analyzed before, $t_{s_c} > t_{s_p}$ and $s_p > s_c$. Therefore $T_c > T_n$ has the high potential to be satisfied.

VI. CONCLUSION

In this paper, the most simple digital waveform, a pulse wave, is used as the test stimulus for ADC testing. Instead of testing the conventional dynamic parameters, two algorithms are applied to obtain the out-of-amplitude-range and out-of-angle-range percentages. The basic concept of the algorithms is that of comparing the similarity between the outputs of golden devices and the DUTs by the amplitude or angle deviation. The method has been verified on a 6-bit flash ADC with gain, offset and bandwidth parametric faults in the first stage amplifiers. The results show that our method can detect the fault even better than the conventional parameter THD, especially in the case of bandwidth faults. Comparing these two algorithms, the out-of-amplitude-range percentage is more sensitive to the fault deviations. Since the digital waveform is the test stimulus, this method is very suitable for applying to a multi-site ADC test environment. Before testing the specific values of the conventional dynamic or static parameters, it can filter out most of the faulty devices in a simple and quick way. As a result, it will help to reduce the ADC production test time and costs significantly.

REFERENCES

[1] C. Wegener, M. P. Kennedy, "Innovation to Overcome Limitations of Test Equipment", *Circuit Theory and Design Conference*, pp. 309-314, 2005.

[2] M. Burns, G.W. Roberts, An Introduction to Mixed-Signal IC Test and Measurement, *Oxford University Press*, 2000.

[3] Xiaoqin Sheng, Hans Kerkhoff, Amir Zjajo, Guido Gronthoud, "Exploring dynamics of embedded ADC through adapted digital input stimuli", *IEEE Mixed-Signals, Sensors, and Systems Test Workshop,* pp. 130-136 , 2008

[4] M. da Bloria Flores, M. Negreiros, L. Carro, "INL and DNL Estimation based on Noise ADC Test", *Instrumentation and Measurement Technology Conference*, pp.1350-1353, 2003.

[5] A. Roy, S. Sunter, A. Fudoli, "High Accuracy Stimulus Generation for A/D Converter BIST", *International Test Conference*, pp. 1031-1039, 2002.

[6] D. De Venuto, L. Reyneri, "Full Digital Optimized Testing and Calibration Technique for ∑Δ ADCs", *International Symposium on Quality Electronic Design*, pp. 519-526, 2007.

[7] H.-K. Chen, C.-Hu Wang, C.-C. Su, "A Self Calibrated ADC BIST Methodology", *VLSI Test Symposium*, pp. 117-122, 2002.

[8] L. Jin, K. Parthasarathy, T. Kuyel, "Linearity Testing of Precision Analog-to-Digital Converters Using Stationary Nonlinearity Inputs", *International Test Conference*, pp. 218-227, 2003.

[9] L. Jin, C. He, D. Chen, R. Geiger, "Fast Implementation of a Linearity Test Approach for High Resolution ADCs Using Non-linear Ramp Signal", *International Symposium on Circuits and Systems*, pp. 932-935, 2004.

[10] L. Jin, K. Parthasarathy, T. Kuyel, "High-Performance ADC Linearity Test Using Low-Precision Signals in Non-Stationary Environments", *International Test Conference*, pp. 1182 – 1191, 2005.

[11] Irons, F.H.; Hummels, D.M., "The modulo time plot-a useful data aquisition diagnostic tool", *Instrumentation and Measurement*, pp. 734-738, 1996.

[12] M. Steyaert, R. Roovers, J. Craninckx, "A 100 MHz 8 bit CMOS Interpolating A/D Converter", *Proceedings of IEEE Custom Integrated Circuit Conference,* pp. 28.1.1–28.1.4, 1993.

Authorized licensed use limited to: UNIVERSITEIT TWENTE. Downloaded on January 28, 2010 at 03:41 from IEEE Xplore. Restrictions apply.