IEICE TRANS. INF. & SYST., VOL.E99–D, NO.11 NOVEMBER 2016

PAPER

Reseeding-Oriented Test Power Reduction for Linear-Decompression-Based Test Compression Architectures

Tian CHEN[†a)], ***Member***, **Dandan SHEN**[†], **Xin YI**[†], **Huaguo LIANG**[††], ***Nonmembers***, **Xiaoqing WEN**[†††], ***Member***, ***and*** **Wei WANG**[†], ***Nonmember***

SUMMARY Linear feedback shift register (LFSR) reseeding is an effective method for test data reduction. However, the test patterns generated by LFSR reseeding generally have high toggle rate and thus cause high test power. Therefore, it is feasible to fill X bits in deterministic test cubes with 0 or 1 properly before encoding the seed to reduce toggle rate. However, X-filling will increase the number of specified bits, thus increase the difficulty of seed encoding, what's more, the size of LFSR will increase as well. This paper presents a test frame which takes into consideration both compression ratio and power consumption simultaneously. In the first stage, the proposed reseeding-oriented X-filling proceeds for shift power (shift filling) and capture power (capture filling) reduction. Then, encode the filled test cubes using the proposed Compatible Block Code (CBC). The CBC can X-ize specified bits, namely turning specified bits into X bits, and can resolve the conflict between low-power filling and seed encoding. Experiments performed on ISCAS'89 benchmark circuits show that our scheme attains a compression ratio of 94.1% and reduces capture power by at least 15% and scan-in power by more than 79.5%.

key words: *low power test, data compression, LFSR, X-filling*

1. Introduction

Due to the shrinkage of semiconductor feature size, the growth in circuit scale and the increase in clock speed, lots of defects related to the timing problem may come into being during the manufacture process, and cause faults in logic circuits.

In order to guarantee production quality, testing at clock frequencies equal to functional clock frequencies (at-speed testing) is essential. Unlike slow-speed test, at-speed testing suffers from poor test safety, test quality and test costs and most of these issues are related to high test power. According to reports of international technology roadmap for semiconductors (ITRS), the test power has become a significant challenge. Usually, test power of circuit is much higher than functional power as much as two to five times higher [1]. In some cases, the peak power of test even reaches up to 30 times higher of the functional power [2]. An excessive test power can create many problems such as temperature increase of circuit which may cause hot spots, IR drop and noise which may cause yield loss, structural damage of chip and reliability degradation. Plenty of schemes have been proposed to reduce test power, such as low-power test scheduling [3], [4], test-pattern ordering [5], low-power test data compression [6], [7], scan chain structural conversion [8], low-power automatic test pattern generation (ATPG) algorithm [9], [10] and low-power X-filling [11], [12] and so on. These schemes can reduce test power effectively, nevertheless some of them may result in poor test quality and the increase in test data. Researchers still have a long way to go to balance test cost and test safety.

On the other hand, as the circuit scale increases, more data is needed to test the whole chip. A large amount of data results in a long test time and requires better automatic test equipment (ATE) performance, higher test channel bandwidth and larger test storage. That's one of the main reasons why test cost rises. The main schemes for compressing test data include code-based scheme, broadcast-scan-based scheme and linear-decompression-based scheme. Code-based scheme involves Golomb code [13], EFDR code [14], RL-Huffman code [15] and dictionary code [16] etc. Broadcast-scan-based scheme contains Illinois scan [17], multiple-input broadcast Scan [18], reconfigurable broadcast scan [19] and virtual scan [20] etc. Linear-decompression-based technology utilizes linear finite state machine (such as LFSR, ring generator, EDT and folding counter etc.) to decompress test seeds and generate test patterns vectors on chip [21], [22]. Linear-decompression-based technology, especially LFSR, is adopted by most commercial tools, such as TestKompress of Mentor Graphics, SOC BIST of Synopsys and Smart BIST of Cadence etc. for its high compression ratio, low hardware overhead and simple implementation.

Lots of compression schemes may bring about serious power problem. The reason is that the test pattern generated by decompression tends not to consider the low-power constraints thereby result in high toggle rate of the nodes in the circuit under test (CUT). Test methods based on linear-decompression-based structure may face more severe power consumption. The problem is due to the conflicts among compressing seed and shift power reduction and capture power reduction. It is generally believed that is a challenge for test methods based on linear-decompression-based structure to simultaneously optimize shift power and capture

Manuscript received July 22, 2015.
Manuscript revised June 4, 2016.
Manuscript publicized July 25, 2016.

[†]The authors are with the School of Computer Science and Information, Hefei University of Technology, Hefei 230009, China.
[††]The author is with the School of Electronic Science and Applied Physics, Hefei University of Technology, Hefei 230009, China.
[†††]The author is with the Faculty of Computer Science and Systems Engineering, Kyushu Institute of Technology, Iizuka-shi, 820–8502 Japan.
a) E-mail: ct@hfut.edu.cn

DOI: 10.1587/transinf.2015EDP7289

power.

This paper presents a low-power linear-decompression-based scheme which operates X-ize specified bits and X-filling in turn to enhance the efficiency of seed encoding and reduce shift power and capture power simultaneously. In addition, the scheme yields an overall framework on the basis of reseeding-oriented X-filling which solves test compression and power problem simultaneously.

2. Motivation

Testing of scan circuits has three operation modes: normal mode, shift mode and capture mode. In shift mode, test patterns are shifted into scan chains, and this phase takes up most of the test time. So, in the general case, shift power determines the average power of the whole test and high average power will lead to high temperature of circuit. In capture mode, the scan cells are used to capture the test responses from the combinational logic. Although the capture phase lasts for a short time, low correlation between test responses and test stimulus will result in a high toggle rate of nodes in the CUT. Generally, capture power is much higher than shift power and the peak power of test often occurs during capture mode. Excessive peak power during testing can cause ground bounce and power supply droop. This instantaneous drop in power supply causes excessive delay and a path-delay problem. This excessive power supply noise may erroneously change the logic state of some circuit nodes and accordingly lead to manufacturing yield loss.

The test cube set generated by the ATPG algorithm usually contains lots of X bits, and a typical test cube set has more than 95 percent X bits. At present, X-filling is the main way to reduce capture power, such as X-filling based on probabilistically-estimation [12], preferred-fill based on probability [23], JP-fill based on adjustment [24] etc. X-filling can also be used to reduce shift power, such as 0-filling, 1-filling, minimum transition filling [25] and adjacent filling [26] etc. However, in the test based on linear decompression structure, the hardware overhead of linear decompression and the success of compressing the deterministic test set are determined by the number of specified bits in test cubes. If plenty of X bits are filled with specified values, we may fail to compress the test cube. Taking for example the deterministic test based on LFSR reseeding, test cubes should be compressed into seeds and LFSR is taken as the decompressor to generate test patterns by extending seeds. Supposing that the degree of characteristic polynomial series of LFSR is k, if there are s specified bits in a given test cube C, the seed can be got by solving an equation set with s equations and k unknowns, where each specified bit in C corresponds to an equation in the equation set. We resort to the widely used Gaussian-Jordan elimination method to solve the equation set. C can be compressed into a seed if there exists a solution to the equation set, otherwise can't. The more specified bits, the more equations, the less the probability of encoding seed successfully. Whether a LFSR can compress a test set or not is determined by the test cube with the most specified bits. When LFSR decompresses seeds into test patterns, X bits in the test cube are set to pseudo-random values generated by LFSR. Thus these test patterns have high toggle rate and power consumption. However, excessive X-filling before computing seed for power reduction not only introduces additional hardware overhead, but also reduces the probability of successful seed encoding. Therefore, there is a conflict between low-power X-filling and encoding LFSR seed.

On the other hand, shift filling and capture filling often fill one bit with conflicting values for shift power and capture power reduction respectively.

Based on the above analysis, a reseeding-oriented low-power test compression scheme is proposed in this paper, which can X-ize specified bits and resolve conflict between X-filling and encoding seed, while increasing compression ratio. To optimize capture power and shift power simultaneously and resolve the conflict between them, we decide X-filling value according to the influence of these two kinds of power and perform shift filling and capture filling in turn.

3. Low-Power Test Method Based on LFSR Reseeding

3.1 Overall Test Framework

As shown in Fig. 1, the proposed scheme first performs reseeding-oriented X-filling to fill X bits in the test set generated by ATPG, then encodes the filled test set using the CBC. Next the test set is compressed and stored in ATE. During test, the test set is decompressed and fed into scan chains by LFSR. Last, test responses are compressed and stored in ATE.

In the test, the shift power determines the average power of the whole test which is in direct proportion to the heat dissipation of the CUT. Thus a high average power will enhance the temperature of the circuit. Researches show that, during the test, the failure probability of circuit doubles for each ten-degree rising of temperature [27]. The reliability of the overall system may consequently degrade significantly even fail. Therefore, we should reduce shift power as much as possible. Although capture power is generally not an issue compared to shift power, it will lead to power supply noise, over-testing and yield loss. Consequently, we

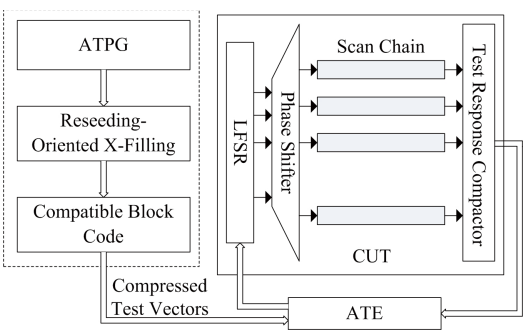


Fig. 1 LFSR-based compression infrastructure

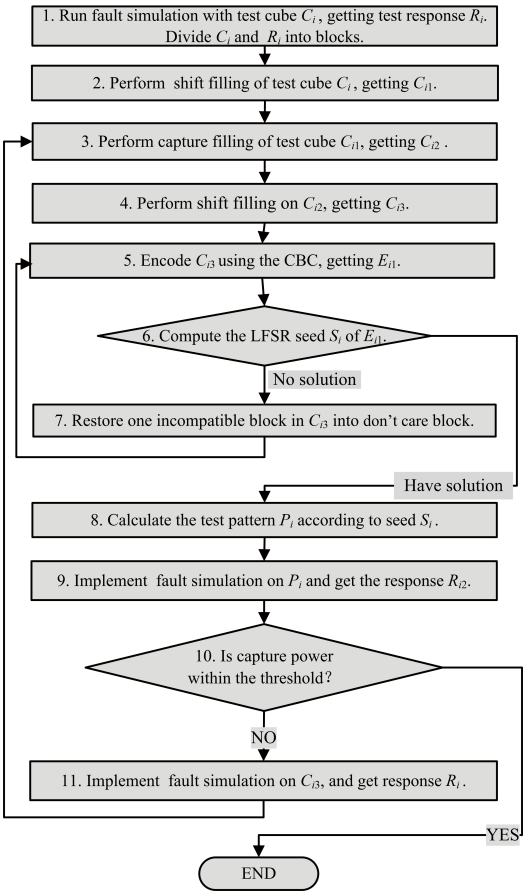


Fig. 2 Frame diagram of reseeding-oriented low power test

should reduce capture power to keep it (especially during launch) under the safe threshold. Thus, during the whole test process, the basic principle of X-filling is to reduce the shift power as much as possible without compression ratio loss, while keeping the capture power under the safe threshold.

As shown in Fig. 2 the process of the proposed method is as follows:

1) Run fault simulation with the test cube C_i to get test response R_i which contains X bits. Divide C_i and R_i into data blocks of the same length;

2) Perform shift filling of the test cube C_i and get C_{i1}. Namely find out all 0-compatible blocks and 1-compatible blocks in C_i and fill the X bits with 0 and 1 respectively.

3) Perform capture filling of C_{i1} and get C_{i2};

4) Perform shift filling of C_{i2} and get C_{i3}. Since during the capture filling process, some new 0-compatible blocks and 1-compatible blocks are generated. In order to reduce shift power, it is necessary to perform shift filling again.

5) To realize X-ize specified bits, encode C_{i3} using the compatible block code (CBC) and get the test cube E_{i1};

6) Compute the LFSR seed S_i of E_{i1}. According to characteristic polynomial, build an equation set whose equation number is equal to the number of specified bits in E_{i1}. Solve the equation set through Guass-Jordan elimination. If there exists a solution, go to 8), otherwise, go to 7);

7) Restore one incompatible block B_{ij} in C_{i3} into don't care block. B_{ij} is don't care block before X-filling, but more specified bits are generated after X-filling. Thus B_{ij} is encoded as incompatible block in E_{i1}. Go to 5);

8) generate the test pattern P_i by extending seed S_i;

9) Run fault simulation with P_i to get the response R_{i2};

10) Calculate the weighted switching activities (WSA) of capture power of circuit nodes. If capture power is within the threshold, go to 12). Otherwise, go to 11);

11) Run fault simulation with C_{i3} and get response R_i then go to 3).

12) Take S_i as seed, end.

3.2 Reseeding-Oriented X-Filling

Reseeding-oriented X-filling contains three steps: shift filling, capture filling and the secondary shift filling. For the sake of encoding, it is necessary to divide the test cubes into blocks and then fill each block according to its type. We need to reduce capture power until it is under the safe threshold and reduce shift power as much as possible. Furthermore, the encoding of data blocks which are filled for shift power reduction will generate much less specified bits and this will make for X-ize specified bits. Hence, shift filling takes priority over capture filling.

This paper defines four types of data blocks: 0-compatible block, 1-compatible block, incompatible block and don't care block.

Definition 1: Let $B = \langle b_0, b_1, \cdots, b_{m-1} \rangle \in \{0, \mathrm{X}\}^m$ be a given test sequence, and $S(B)$ be the set of specified bits in B. If $S(B) \neq \varnothing$, B is a 0-compatible block.

Definition 2: Let $B = \langle b_0, b_1, \cdots, b_{m-1} \rangle \in \{1, \mathrm{X}\}^m$ be a given test sequence, and $S(B)$ be the set of specified bits in B. If $S(B) \neq \varnothing$, B is a 1-compatible block.

Definition 3: Let $B = \langle b_0, b_1, \cdots, b_{m-1} \rangle \in \{\mathrm{X}\}^m$ be a given test sequence, then B is a don't care block.

Definition 4: If B is none of the above, then B is an incompatible block.

Shift filling contains 0-filling and 1-filling, namely all don't care bits in a 0-compatible block are set to 0 and all don't care bits in a 1-compatible block are set to 1. 0-filling and 1-filling reduce both shift power and capture power. That's because there will be many 0 bits in test response if AND gate is in majority in the circuit, while there will be many 1 bits if OR gate is in majority.

In the test, shift filling should be executed twice. The first time is before capture filling for 0-compatible and 1-compatible blocks and the second time is after capture filling. Some don't care blocks may turn to 0-compatible blocks and 1-compatible blocks during the capture filling process. Hence, it is necessary to perform shift filling once more for a further shift power reduction. Namely, the secondary shift filling fills blocks which are don't care blocks before capture filling however are filled and become compatible block during capture filling. Table 1 illustrates why shift filling should be performed twice. Assuming the length

Table 1 An example of reseeding-oriented X-filling

Block	B_1	B_2	B_3	B_4
Test response	XXX0XXXX	X1XXXXXX	XXXXX0XX	XXXXXXXX
Original test cube	0XXX0XXX	X1XXXXX0	XXXXXXXX	XXXX1XXX
Shift filing(first time)	00000000	X1XXXXX0	XXXXXXXX	11111111
Capture filling	00000000	X1XXXXX0	XXXXX0XX	11111111
Shift filing(second time)	00000000	X1XXXXX0	00000000	11111111

of block is 8. B_3 is a don't care block at the beginning thus isn't filled during the first-time shift filling. Nevertheless it turns to a 0-compatible block during the capture filling, performing its shift filling for a second time can reduce shift power.

Our method fills X bits in the test cube with specified bits in the corresponding position in the corresponding test response to reduce capture power. First of all, we run fault simulation with the test cube to get the test response. If a bit r_i in the test response is specified bit (r_i is the i-th bit in the test response, $r_i \in \{0, 1\}$) and the corresponding i-th bit in the test cube is X bit, then we fill the X bit with r_i. Specified bits in the test response only relate to specified bits in the test cube rather than unspecified bits. The deterministic test cube has a few specified bits as a result there will be a few specified bits in the test response. So capture filling won't cause sharp increase of specified bits in the test cube.

3.3 The CBC Oriented to Reseeding

During capture filling, the inclusion relation among X bits in a test cube may cause the failure of seed encoding of the test cube. What's more, the compression ratio is relevant to the amount of specified bits in the test set. Thus it is necessary to encode the test cube using the proposed CBC to reduce specified bits.

The CBC first divides test cubes into data blocks of the same length which is calculated according to theorems and corollaries in Sect. 3.4, and then encodes these blocks according to their types. The process of the CBC is as follows:

(1) The CBC first divides test cubes into data blocks. Suppose a test set $T_D = \{C_1, C_2, \cdots, C_n\}$, and a test cube $C_i = \langle b_{i0}, b_{i1}, \cdots, b_{i(m-1)}\rangle \in \{0, 1, X\}^m$ of T_D, where m is the length of test cubes. The process of dividing C_i into blocks of length l: taking the top l bits as the first block (namely b_{i0}, b_{i1}, $\cdots$, $b_{i(l-1)}$), the l+1-th bit to 2l-th bit as the second block (namely b_{il}, $b_{i(l+1)}$, $\cdots$, $b_{i(2l-1)}$), and so on. After that, C_i is divided into $\lceil m/l \rceil$ blocks denoted as B_{i1}, B_{i2}, $\cdots$, B_{ij}, $\cdots$, $B_{i\lceil m/l\rceil}$, $B_{i1} = \langle b_{i0}, b_{i1}, \cdots, b_{i(l-1)}\rangle$, $B_{i2} = \langle b_{il}, b_{i(l+1)}, \cdots, b_{i(2l-1)}\rangle$, $\cdots$, $B_{ij} = \langle b_{i((j-1)l)}, b_{i((j-1)l+1)}, \cdots, b_{i(jl-1)}\rangle$, $\cdots$, $B_{i\lceil m/l\rceil} = \langle b_{i((\lceil m/l\rceil-1)l)}, b_{i((\lceil m/l\rceil-1)l+1)}, \cdots, b_{i(m-1)}\rangle$. $B_{i\lceil m/l\rceil}$is the last block and its length needn't be l.

(2) Then the CBC identifies the type of each block. Taking the j-th block of C_i, $B_{ij} = \langle b_{i((j-1)l)}, b_{i((j-1)l+1)}, \cdots, b_{i(jl-1)}\rangle$ for example, the CBC first identifies the type of B_{ij} based on above Definition 1 to Definition 4, then, adds a bit of block flag in the front of B_{ij} according to its type. New test cube after being encoded comprises block flag and data bits in turn. Supposing B'_{ij} is got by encoding B_{ij} using the CBC. Though there are four types of data blocks, only one bit is added to mark the types of blocks by extending block flag. That's to say, B'_{ij} comprises l+1 bits: one bit of block flag and l bits of data bits. But if the block flag is 0, extend the block flag by putting the second bit of B'_{ij} in the block flag. Namely, the first two bits of B'_{ij} is the extending block flag and the remaining l-1 bits are data bits. That's what we call extending block flag.

Table 2 The CBC encoding table

Type of Blocks	Block Flag	Data bits
incompatible block	1	original block data
0-compatible block	00	XXXXXXX
1-compatible block	01	XXXXXXX
don't care block	X	XXXXXXXX

Supposing the length of block is l. If B_{ij} is a incompatible block, the one-bit block flag of B_{ij} is 1, new block $B'_{ij} = \langle 1, B_{ij}\rangle$. If B_{ij} is a 0-compatible block, the two-bit block flag of B'_{ij} is 00 and the data bits are l-1 bits of X, new block $B'_{ij} = \langle 0, 0, X, \cdots, X\rangle$. If B_{ij} is a 1-compatible block, the two-bits block flags of are 01 and data bits are l-1 bits of X, new block $B'_{ij} = \langle 0, 1, X, \cdots, X\rangle$. If B_{ij} is a don't care block, then the block flag of B_{ij} is X and data bits are l bits of X, $B'_{ij} = \langle X, X, \cdots, X\rangle$. The combination of all encoded blocks in order is the new test cube C'_i, $C'_i = B'_{i1} \cup B'_{i2} \cup \cdots \cup B'_{i\lceil m/l\rceil}$. Table 2 is the CBC encoding table. The length of block l in it is assumed to be 8.

Table 3 is an example of the CBC. Supposing thirty-two-bit test cube C could be divided into four blocks $C = \langle B_1, B_2, B_3, B_4\rangle$. Encode each block and add block flags for them. There are 19 specified bits in the original test cube, but only 12 specified bits in encoded test cube. It can be observed the proposed method in this paper is able to lessen specified bits and increase the successful probability of seed encoding. As for B_1 and B_2, decoding circuit generates all-0 and all-1 test pattern blocks according to their block flags and then feeds them into scan chains. These blocks have no transitions, and this is beneficial for reducing shift power.

3.4 Method of Calculating the Optimal Block Length

Partitioning the test cube suitably is the key to enhancing compression ratio. So we will analyze the relations among test data volume, the size of LFSR and data block length below.

Supposing the size of LFSR is k, and the test cube $C \in \{0, 1, X\}^m$ has s specified bits, the probability that C can be encoded into k-bit seed of the LFSR is $P_{seed}(k, s)$,

Table 3 An encoding example of the CBC

Block	Original Data	Block Type	Flag	Encoded
B_1	0X00X000	0-compatible block	00	00XXXXXXX
B_2	X1X11111	1-compatible block	01	01XXXXXXX
B_3	0X110100	incompatible block	1	10X110100
B_4	XXXXXXXX	don't care block	X	XXXXXXXXX

and $P_{noseed}(k, s) = 1 - P_{seed}(k, s)$ is the probability that C can't be encoded into k-bit seed of the LFSR. According to literature [28], when $2^k - 1 \geq m$, namely the period of LFSR $2^k - 1$ is greater than the length of the test cube m, we consider establishing a set of linear equations as constructing a Markov chain $(X_t)_{1 \leq t \leq s}$ over the state set $S = \{0, 1, \cdots, k\}$. As shown in Fig. 3, the t-th equation is generated in t-th step. Assuming that the rank of the set with t equations is d, when $1 \leq d \leq t \leq s$ and $X_t = d$ there exists a unique solution to the set, while $X_t = 0$ means there exists no solution.

A Markov chain is determined by its initial distribution and the transition probabilities. Its initial distribution is as follows:

$$\begin{cases} P(X_1 = 1) = 1, \\ P(X_1 = d) = 0, \quad (d \neq 1). \end{cases} \tag{1}$$

Transition probabilities:

$$\begin{cases} P(X_{t+1} = d + 1 \mid X_t = d) \\ \qquad = \begin{cases} \dfrac{2^k - 2^d}{2^k - 1 - t}, (1 < d + 1 \leq k), \\ 0, otherwise; \end{cases} \\ P(X_{t+1} = d \mid X_t = d) \\ \qquad = \begin{cases} \dfrac{1}{2} \cdot \dfrac{2^d - 1 - t}{2^k - 1 - t}, (d > 0, t + 1 \leq 2^d - 1), \\ 0, otherwise; \end{cases} \\ P(X_{t+1} = 0 \mid X_t = d) = P(X_{t+1} = d \mid X_t = d), (d > 0); \\ p(X_{t+1} = 0 \mid X_t = 0) = 1. \end{cases} \tag{2}$$

Theorem 1 [29]: Hypothesizing $C \in \{0, 1, X\}^m$ is a test pattern with s specified bits, and $h(x)$ is a k-degree primitive polynomial. Let $(X_t)_{1 \leq t \leq s}$ be a Markov chain defined above, the possibility of encoding C successfully is given by:

$$P_{seed}(k, s) = \sum_{d=1}^{s} P(X_s = d) = \sum_{d=\lceil log_2(s+1) \rceil}^{min(k,s)} P(X_s) = d \tag{3}$$

$P(X_s = d)$ is gotten by recursive computation, $P(X_t = d) = P(X_{t-1} = d) \cdot P(X_t = d \mid X_{t-1} = d) + P(X_{t-1} = d - 1) \cdot P(X_t = d \mid X_{t-1} = d - 1)$.

Corollary 1: Let $T \in \{0, 1, X\}^m$ be a test set with n test vectors and $h(x)$ a characteristic polynomial of degree k. Then the probability that the original T can be encoded successfully into seed of LFSR of size k is given by:

$$P_{setsucc} = \prod_{i=1}^{n} P_{seed}(k, s_i) \tag{4}$$

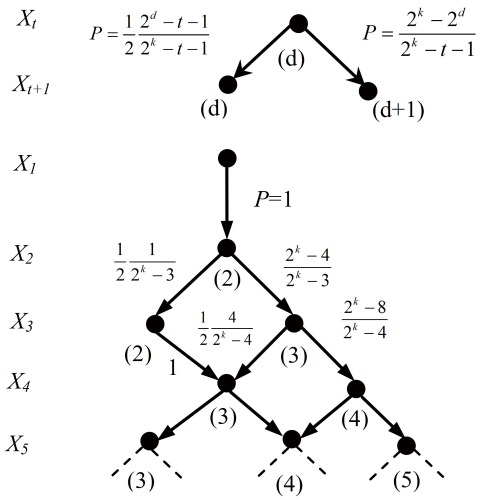


Fig. 3 Structure of the Markov chain $(X_t)_{1 \leq t \leq s}$

Where s_i is the number of specified bits in the i-th test vector.

Corollary 2: Let $T \in \{0, 1, X\}^m$ be a test set with n test vectors. Then, among the sizes of all the LFSRs that can encode T, the probability that the minimum size is k is given by:

$$P_{mindegree}(k) = P_{setsucc}(k) \times \prod_{i=1}^{k-1} P_{setunsucc}(i) \tag{5}$$

Corollary 3: Let $T \in \{0, 1, X\}^m$ be a test set with n test vectors, d_{min} the minimum of sizes of all the LFSRs that can encode T. The expectation of d_{min} is given by:

$$E(d_{min}) = \sum_{i=\lceil log_2(m+1) \rceil}^{S_{max}+20} (i \times P_{mindegree}(i)) \tag{6}$$

According to [30], the probability of unsuccessful seed encoding is less than 10^6 when the size of LFSR reaches to $S_{max} + 20$. Therefore, the upper limit of i is $S_{max} + 20$. Meanwhile, since the constraint of Theorem 1 holding is $2^k - 1 \geq m$, we take the value of $\lceil log_2(m + 1) \rceil$ as the lower limit of i. Mathematical expectation of the storage of seed is obtained by multiplying $E(d_{min})$ by the number of seed.

To determine the optimal block length, we first calculate $E(d_{min})$ under different given block lengths according to the above theorems and corollaries. Different block lengths lead to different $E(d_{min})$ and the length leads to the minimum $E(d_{min})$ is adopted.

Taking s13207 for example, divide the test set for

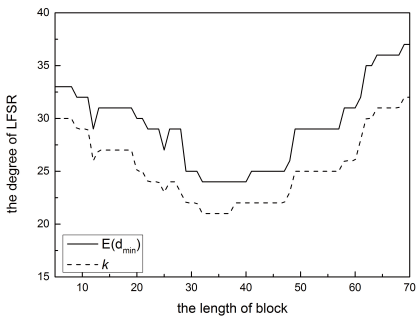


Fig. 4 Relationship between the length of block and the size of LFSR

s13207 with different block lengths from 5 to 70. For every block length, fill and encode the test set first using the proposed scheme then calculate $E(d_{min})$ and the actual size k of LFSR. As shown in Fig. 4, the horizontal axis represents the length of block and the vertical axis represents the size of LFSR. The solid line represents $E(d_{min})$ while the dotted line indicates k. As can be seen, although there is error between $E(d_{min})$ and k, the variation trends of them are consistent. Thus, when $E(d_{min})$ is the minimum, k is the minimum too.

4. Hardware Implementation

Figure 5 is the hardware implementation of our scheme. There are two flip-flops (FFs) in front of each scan chain, F1 and F2 which are used to store the first and the second bit of block flag respectively. F1 and F2 are control signals of two multiplexers (MUXs) respectively. The value of F1 determines whether a block is compatible or incompatible, while the value of F2 decides whether a block is a 0-compatible or a 1-compatible block. LFSR decompresses seed into test pattern that contains block flags and data bits. Supposing the length of each block is l+1 firstly, feed F1 and F2 with the first and the second bit of flag generated by LFSR. If F1=1, feed scan chain with l bits of data generated by LFSR, if F1=0 and F2=0, with l bits of 0, if F1=0 and F2=1, with l bits of 1. Though the test patterns which are shifted in by blocks contain block flags, test sequences loaded into scan chains are free from block flags and compatible with the deterministic test cubes generated by ATPG.

5. Experimental Results

To evaluate the proposed scheme, experiments were performed on six largest ISCAS'89 benchmark circuits for 100% fault coverage. We stitched scan flip-flops (SFFs) into scan chains according to their orders in the circuit file, then divide each scan chain into n sub chains of the same length without according to its original order. We adopted the hard fault set which set aimed at faults hard to detect by the pseudorandom test [31]. We used test cubes set which ATPG generated for hard faults remaining after 10000 times of pseudo-random test.

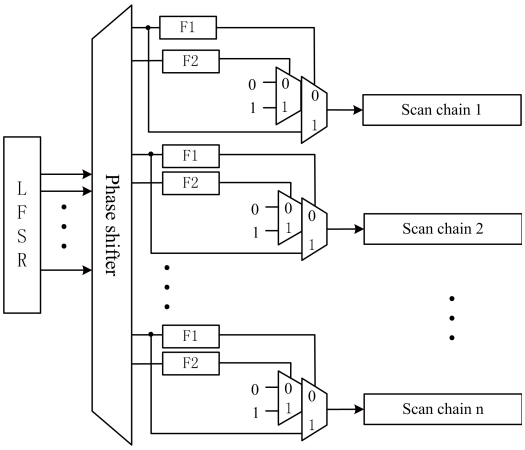
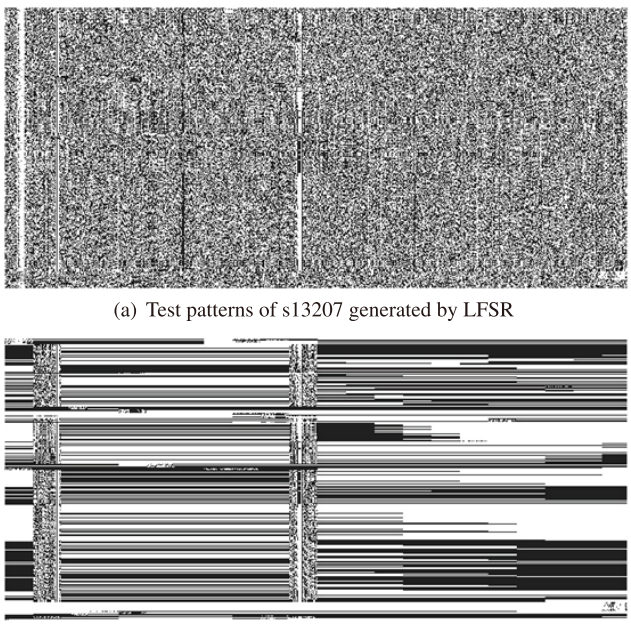


Fig. 5 Hardware implementation



(a) Test patterns of s13207 generated by LFSR

(b) Test patterns of s13207 generated by the proposed scheme

Fig. 6 Visual compare of test pattern set of s13207 before and after optimized

Taking for example s13207, Fig. 6 visually compares test patterns generated by the proposed scheme and traditional scheme based-on LFSR. In Fig. 6, black spots indicate 1 while white spots indicate 0. The vertical axis represents test patterns and the horizontal axis represents each bit in test patterns. For example, a black spot whose coordinates are (i, j) indicates the i-th bit of the j-th test pattern is 1. Figure 6 (a) is the visual expression of the deterministic test set decoded by traditional LFSR. We can see that its switching activity rate is close to 50%, which indicates a high shift power. Figure 6 (b) is the visual expression of test set generated by the proposed method, where we can see lots of sequential identical bits (0 or 1) thus the transition ratio reduces observably during scan in.

Table 4 provides the circuit and test set information.

Table 4 Circuit and test set information

Circuit	#SCs	Scan architecture	TD	0(%)	1(%)	X(%)	#Testable faults	#HDF	Hardware	
									#MUX	#FF
s5378	214	8×30	5992	3.82	3.67	92.51	4551	86	16	16
s9234	247	9×30	73112	5.54	4.85	89.60	6927	1096	18	18
s13207	700	11×64	220500	1.38	1.78	97.83	9815	701	22	22
s15850	611	9×75	163748	2.56	1.67	95.76	11725	1091	18	18
s38417	1664	17×100	2201472	0.87	0.98	98.15	31180	2451	34	34
s38584	1464	10×160	456768	0.75	0.35	98.90	36303	2057	20	20

Table 5 Shift-in and shift-out power under different thresholds of capture power

Circuit	10%				15%				20%			
	shift-in		shift-out		shift-in		shift-out		shift-in		shift-out	
	Ave.	Peak	Ave.	Peak	Ave.	Peak	Ave.	Peak	Ave.	Peak	Ave.	Peak
s9234	2045	7798	4746	10570	2104	8784	5498	11158	2345	9984	6746	12547
s13207	13589	37588	23401	57253	14599	46975	25878	58748	16548	59878	28481	59898
s15850	12598	30589	19060	52941	15916	37856	22871	54548	18487	45849	24890	57461
s38417	110477	289585	175658	374826	123843	317805	189568	407426	154876	389875	195564	415846
s38584	118475	280846	199652	382764	134298	311877	228642	405874	148774	379847	236949	414514

Table 6 Comparison with other solutions on compression ratio and the LFSR size

Circuit	9C-RLHC [32]			Hybrid coding [31]				Reseeding of Dividing [33]		Proposed			
	T_D (bits)	T_E (bits)	Comp (%)	T_D (bits)	S_{max}	T_E (bits)	Comp (%)	T_E(bits)	Comp(%)	Length	Size	T_E (bits)	Comp (%)
s5378	23754	6664	71.9	5992	18	897	85	504	91.6	10	19	532	91.1
s9234	39273	13176	66.5	73112	49	13927	81	10920	85.1	10	34	10064	86.2
s13207	165200	22553	86.3	220500	22	8574	96.1	6320	97.1	32	21	6615	97.0
s15850	76986	17629	77.1	163748	48	9478	94.2	9396	94.3	25	30	8040	95.1
s38417	164736	43097	73.8	2201472	66	73841	96.7	62580	97.2	50	43	56889	97.4
s38584	199104	47399	76.2	456768	54	10787	97.6	9251	98	40	31	9672	97.9
Average	111509	25086	77.5	520265	42.8	19584	91.8	16495	93.9			15302	94.1

Table 7 Comparison with LFSR reseeding on power

Circuit	Reseeding of Multi-Polynomial LFSR [28]			Proposed				
	Total	Ave.	Peak	Total	Ave.	Peak	Ave. redu.(%)	Peak. redu.(%)
s9234	3130263	10575	14005	622922	2104	8784	80.1	37.28
s13207	33542231	106483	125032	4595286	14599	46975	86.3	62.43
s15850	21116446	78793	99422	4265522	15916	37856	79.8	61.92
s38417	752149515	568089	700320	163968594	123843	317805	78.2	54.62
s38584	155765650	499249	554448	41900960	134298	311877	73.1	43.75
Average							79.5	52.0

Table 8 Comparison with other schemes on power and compression ratio

Circuit	Dual MP-LFSR reseeding [34]			9C-RLHC [32]			Low-power LFSR reseeding [35]			Proposed		
	T_E (bits)	Comp (%)	Ave. redu. (%)	T_E (bits)	Comp (%)	Ave. redu. (%)	T_E (bits)	Comp (%)	Ave. redu. (%)	T_E (bits)	Comp (%)	Ave. redu. (%)
s9234	19440	68	24	13176	66.5	79.9	10302	79	53	10064	86.2	80.1
s13207	11803	94	25	22553	86.3	94	10484	94	53	6615	97.0	86.3
s15850	14518	90	25	17629	77.1	83.2	11411	93	52	8040	95.1	79.8
s38417	66234	92	25	43097	73.8	80.3	32152	95	52	56889	97.4	78.2
s38584	23835	94	25	47399	76.2	83.7	31152	93	40	9672	97.9	73.1
Average	27166	87.6	24.8	28771	76	84.2	19100	90.8	50	18256	94.7	79.5

The second and third columns show the number of scan cell and the scan architecture (the number of scan chain × the length of scan chain). The third to seventh columns show test set size and the percentages of 0-bits, 1-bits and X-bits in the initial test set. #Testable fault and #HDF indicates the numbers of testable faults and hard-to-detected faults which are not detected by 10000 times of pseudo-random test. The last two columns are the numbers of 2-to-1 Muxs and FFs which are in direct proportion to the number of scan chains. These Muxs and FFs account for the major proportion of the additional hardware overheads of the proposed scheme compared with traditional LFSR-based scheme.

Table 5 shows shift-in and shift-out power of the proposed scheme under different thresholds of capture power. We use the weighted transitions metric (WTM) to estimate the power consumption of the CUT. The shift-in and shift-

out power is estimated by:

$$WTM = \sum_{i=1}^{n} \sum_{j=1}^{m-1} (S_{(i,j)} \oplus S_{(i,j+1)}) \times j$$

where n is the number of scan chains, m is the number of scan cells in a scan chain and $S_{(i,j)}$ represents the logic value of the j-th scan cell in the i-th scan chain. The notations Avg. and Peak indicate the average and peak power. As can be seen, the average and peak power increase with the increase of threshold. The reason is that more X-bits are filled into specified bits for more capture power reduction, thus affects shift-in and shift-out power.

As shown in Table 6, to demonstrate the effectiveness of the proposed scheme in achieving high test data compression, we compare the compression ratio of the proposed scheme with that of 9C-RLHC [32], hybrid coding [31] and reseeding of dividing [33] (The follow-up experimental results are attained under the threshold of 15%). Hybrid coding and reseeding of dividing use the same test set with our scheme. In Table 6, S_{max} is the largest number of specified bits in any test cube. The eleventh sub-column is the length of block used in the proposed scheme which is calculated according to Sect. 3.4. The twelfth sub-column is the size of LFSR of the proposed scheme. The compression ratio computational equation is given by:$Compression\ Ratio = (T_D - T_E)/T_D \times 100\%$, where T_D is the size of the original test set, while T_E is the size after being compressed. Experiments show our scheme equals the reseeding of dividing method while achieves better compression ratio than 9C-RLHC and Hybrid coding.

Table 7 shows the shift-in power of the proposed scheme and the reseeding method based-on multiple-polynomial LFSR [27]. The reseeding based-on multi-polynomial LFSR scheme encodes the deterministic test set into seeds directly and then extends the seeds into test patterns during scan-in. On average, the proposed scheme can reduce shift-in power effectively, average power and peak power by 79.5% and 52.0%, respectively. At the same time, our scheme meets safe threshold of capture power.

Table 8 shows the storage, compression ratio and the average power of the proposed scheme against Dual MP-LFSR reseeding [34], 9C-RLHC code [32] and low-power LFSR reseeding [35]. Experimental results show the proposed scheme precedes other scheme, except for the storage volume of s38417. Among these four scheme, 9C-RLHC coding reduces power to the greatest extent but it is not compatible with LFSR reseeding which is widely used by commercial tools, what'more, it has a larger storage.

Above experimental results show the proposed scheme not only enhances encoding efficiency of LFSR reseeding but also reduces the storage and power consumption effectively.

6. Conclusions

LFSR reseeding-based schemes can reduce test data volume effectively but usually have high power consumption. Moreover, its encoding efficiency is not ideal and the size of LFSR is too big when there are too many specified bits in the test cube set. To solve these problems, this paper proposes a reseeding-oriented low-power test compression scheme, and then designs an overall frame. The proposed scheme performs X-filling and X-ize specified bits in turn to reduce the shift and capture power simultaneously. Experiments show the proposed scheme can reduce power consumption effectively with high compression ratio while keeping the capture power under the safe threshold.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Grant Nos. 61204046, 61474035 and 61432004). This work was partially supported by JSPS KAKENHI Grant-in-Aid for Scientific Research (B) #25280016 and JSPS KAKENHI Grant-in-Aid for Scientific Research on Innovative Areas #15K12003.

References

[1] Y. Zorian, "A distributed BIST control scheme for complex VLSI devices," Proc. VLSI Test Symp., pp.4–9, 1993.
[2] P. Rosinger, B. Al-Hashimi, and K. Chakrabarty, "Rapid generation of thermal-safe test schedules," Proc. Design, Automation and Test in Europe, pp.840–845, 2005.
[3] D. Zhao and S. Upadhyaya, "Dynamically partitioned test scheduling with adaptive TAM configuration for power-constrained SoC testing," Proc. IEEE Trans. Comput.-Aided Des., vol.24, no.6, pp.956–965, June 2005.
[4] T. Chantem, X.S. Hu, and R.P. Dick, "Temperature-aware scheduling and assignment for hard real-time applications on MPSoCs," Proc. Conf. Design, Automation, and Test in Europe (DATE), pp.288–293, 2008.
[5] S. Bahukudumbi and K. Chakrabarty, "Power management using test-pattern ordering for wafer-level test during burn-in," Proc. IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol.17, no.12, pp.1730–1741, 2009.
[6] C.-Y. Lin, H.-C. Lin, and H.-M. Chen, "On reducing test power and test volume by selective pattern compression schemes," Proc. IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol.18, no.8, pp.1220–1224, Aug. 2010.
[7] D. Czysz, G. Mrugalski, J. Rajski, and J. Tyszer, "Low power embedded deterministic test," Proc. VLSI Test Symp., pp.75–83, 2007.
[8] Z. Chen, S. Seth, and D. Xiang, "A novel hybrid delay testing scheme with low test power, volume, and time". Proc. VLSI Test Symp., pp.307–312, 2010.
[9] L.-T. Wang, X. Wen, S. Wu, Z. Wang, Z. Jiang, B. Sheu, and X. Gu, "Test compression technology using combinational logic and one-pass ATPG," IEEE Des. Test of Compu., vol.25, no.2 pp.122–130, 2008.
[10] S. Wang and S.K. Gupta, "ATPG for heat dissipation minimization during test application," Proc. IEEE Trans. Computers, vol.47, no.2, pp.256–262, 1998.
[11] J. Li, Q. Xu, Y. Hu, and X. Li, "X-Filling for Simultaneous Shift- and Capture-Power Reduction in At-Speed Scan-Based Testing," Proc. IEEE Trans. Very Large Scale Integr. (VLSI) Syst, vol.18, no.7, pp.1081–1092, 2010.
[12] X. Wen, K. Miyase, T. Suzuki, Y. Yamato, S. Kajihara, L.-T. Wang, and K.K. Saluja, "A highly guided x-filling method for effective low-capture-power scan test generation," Proc. Int'l Conf. Computer

Design, pp.251–258, 2006.
[13] L. Li and K. Chakrabarty, "On using exponential-Golomb codes and subexponential codes for system-on-a-chip test data compression," Proc. J. Electron. Test Theory Appl., vol.20, no.6, pp.667–670, 2004.
[14] A.H. El-Maleh, "Test data compression for system-on-a-chip using extended frequency-directed run-length code," Proc. IET Comput. Digi. Tech., vol.2, no.3, pp.155–163, 2008.
[15] M.H. Tehranipour, M. Nourani, K. Arabi, and A. Afzali-Kusha, "Mixed RL-Huffman Encoding For Power Reduction and Data Compression In Scan Test," Proc. IEEE Intl. Symp. Circuits and Syst., pp.681–684, 2004.
[16] P. Sismanoglou and D. Nikolos, "Input Test Data Compression Based on the Reuse of Parts of Dictionary Entries: Static and Dynamic Approaches," IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol.32, no.11, pp.1762–1775, 2013.
[17] A.R. Pandey and J.H. Patel, "An incremental algorithm for test generation in Illinois scan architecture based designs," Proc. Design, Automation & Test in Europe, pp.368–375, 2002.
[18] M.A. Shah and J.H. Patel, "Enhancement of the Illinois scan architecture for use with multiple scan inputs," Proc. IEEE Computer Soc. Ann. Symp., VLSI, pp.167–172, 2004.
[19] A.R. Pandey and J.H. Patel, "Reconfiguration technique for reducing test time and test data volume in Illinois scan architecture based designs," Proc. 20th IEEE VLSI Test Symp., pp.9–15, 2002.
[20] L.T. Wang, X. Wen, H. Furukawa, F.-S. Hsu, S.-H. Lin, S.-W. Tsai, K.S. Abdel-Hafez, and S. Wu, "Virtual Scan: a new compressed scan technology for test cost reduction," Proc. Intl. Test Conf., pp.916–925, 2004.
[21] X. Lin, M. Kassab, and J. Rajski, "Using dynamic shift to reduce test data volume in high-compression designs," Proc. European Test Symp., pp.1–6, 2014.
[22] T. Chen, L.Y. Zheng, W. Wang, F.J. Ren, X. S Zhang, and H. Chang, "A low power BIST scheme based on block encoding," 5th Int. Conf. Comput. Commun. Netw. Technol. (ICCCNT), pp.1–7, 2014.
[23] S. Remersaro, X. Lin, Z. Zhang, Z. Zhang, S.M. Reddy, I. Pomeranz, and J. Rajski, "Preferred fill: A scalable method to reduce capture power for scan based designs," Proc. Intl. Test Conf., pp.1–10, 2006.
[24] X. Wen, Y. Yamashita, S. Kajiihara, L.-T. Wang, K.K. Saluja, and K. Kinoshita, "On low-capture-power test generation for scan testing" Proc. VLSI Test Symp., pp.265–270, 2005.
[25] R. Sankaralingam and N.A. Touba, "Controlling peak power during scan testing," Proc. VLSI Test Symp., pp.153–159, 2002.
[26] A. Chandra and R. Kapur, "Bounded Adjacent fill for low capture power scan testing," Proc. VLSI Test Symp., pp.131–138, 2008.
[27] C. Small, "Shrinking devices put the squeeze on system packaging," EDN, vol.39, no.4, pp.41–46, Feb. 1994.
[28] S. Hellebrand, S. Tarnick, J. Rajski, and B. Courtois, "Generation of vector patterns through reseeding of multiple-polynomial linear feedback shift regist," Proc. Intl. Test Conf., pp.120–129, 1992.
[29] S. Hellebrand, J. Rajski, S. Tarnick, S. Venkataraman, and B. Courtois, "Built-in test for circuits with scan based on reseeding of multiple-polynomial linear feedback shift regist," IEEE Trans. Comput., vol.44, no.2, pp.223–233, 1995.
[30] B. Könemann, "LFSR-coded test patterns for scan designs," Proc. Europ. Test Conf., pp.237–242, 1991.
[31] A. Wurtenberger, C.S. Tautermann, and S. Hellebrand, "A hybrid coding strategy for optimized test data compression," Proc. Intl. Test Conf., pp.451–459, 2003.
[32] S. Sivanantham, M. Padmavathy, G. Gopakumar, P.S. Mallick, and J.R.P. Perinbam, "Enhancement of test data compression with multistage encoding," Proc. Integr., VLSI J., vol.47, no.4, pp.499–509, 2013.
[33] H.G. Liang, K.H. Zhan, C.Y. Jiang, and M.X. Yi, "A scheme of LFSR reseeding based on dividing parity bits of test cubes," Chinese Journal of Comput., vol.30, no.10, pp.1689–1695, 2007.
[34] P.M. Rosinger, B.M. Al-Hashimi, and N. Nicolici, "Low power mixed-mode BIST based on mask pattern generation using dual LFSR re-seeding," Proc. Int. Conf. Comput. Des., pp.474–479, 2002.
[35] J. Lee and N.A. Touba, "LFSR-reseeding scheme achieving low-power dissipation during test," IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol.26, no.2, pp.396–401, 2007.

Tian Chen received the B.E., M.E., and Ph.D. in Computer Science and Technology from Hefei University of Technology (HFUT), China, in 1996, 2002 and 2011 respectively. Since 2010 she has been an Assistant Professor in Hefei University of Technology and the Vice Director of Department of Computer since 2013. Dr. Chen is a senior member of the China Computer Federation and a member of the IEICE. Her research interest is low power testing of VLSI, design for testability, System-on-Chip fault-tolerance and reliability.

Dandan Shen received her B.E. degree in Computer Science and Technology from Hefei University of Technology, Chain, in 2014.She has been a postgraduate student since 2014. Her research interest is low power test of VLSI and design for testability.

Xin Yi received his B.E. degree in Computer Science and Technology from Hefei University of Technology, Chain, in 2014. He has been a postgraduate student since 2014.His research interest is low power test of VLSI and design for testability.

Huaguo Liang Prof. Dr. Huaguo Liang is a full professor and the Dean of the School of Electronic Science & Applied Physics and Computer Engineering at the Hefei University of Technology (HFUT), China. Just the same, he is the Dean of the School of Microelectronics and the member of academic degrees committee at HFUT. From 1978 to 1982 he studied at the Hefei University of Technology, where he joined the faculty of the electronic engineering after graduation. Since 1995 he has been an associate professor at HFUT. He received his Ph.D degree from the University of Stuttgart, Germany, in 2003. Huaguo Liang has authored and co-authored more than 100 papers in the area of design, test, reliability and fault tolerance of circuits and systems. He is the vice director of Fault-Tolerant Computing Technical Committee, China Computer Federation.

Xiaoqing Wen received the B.E. degree from Tsinghua University, China, in 1986, the M.E. degree from Hiroshima University, Japan, in 1990, and the Ph.D. degree from Osaka University, Japan, in 1993. From 1993 to 1997, he was an Assistant Professor at Akita University, Japan. He was a Visiting Researcher at University of Wisconsin, Madison, USA, from Oct. 1995 to Mar. 1996. He joined SynTest Technologies, Inc., USA, in 1998, and served as its Chief Technology Officer until 2003. In 2004, he joined Kyushu Institute of Technology, Japan, where he is currently a Professor and the Director of Dependable Integrated Systems Research Center. His research interests include VLSI test, diagnosis, and testable design. He co-authored and co-edited two books: VLSI Test Principles and Architectures: Design for Testability (Morgan Kaufmann, 2006) and Power-Aware Testing and Test Strategies for Low Power Devices (Springer, 2009). He also holds 37 U.S. Patents and 14 Japan Patents on VLSI testing. He received the 2008 IEICE-ISS Best Paper Award for his pioneering work on X-filling-based low-capture-power test generation. He is a Fellow of the IEEE, a member of the IEICE, the IPSJ, and the REAJ.

Wei Wang received the B.E., and Ph.D. in Computer Science and Technology from Hefei University of Technology (HFUT), China, in 2001,and 2007 respectively. He has been an Associate Professor since 2009. Dr. Wang is a senior member of China Computer Federation. His main research interests is Very Large Scale Integrated circuit design and test, low power, System-on-Chip fault-tolerance and reliability.