A Study On Signature Analyzer For Design For Test (DFT)

Abu Khari bin A'ain, *Member, IEEE*, *Chek T Lim, *Ng, Kok Hong, *Ng, Sheng Kwang, Liew Eng Yew

Department of Electronics (INSEED)
Faculty of Electrical Engineering
Universiti Teknologi Malaysia
81310 Skudai, Johor, MALAYSIA

Abstract **This paper takes a look at the use of linear feedback shift registers (LFSR's) as test pattern generators (TPG's) and signature analyzers for built-in self-test (BIST). We also propose a method to generate pseudorandom test patterns. The proposed method can generate longer sequences of the same set of test patterns.**

Keywords: LFSR, PRBS, BIST, ATPG

I. INTRODUCTION

THE process of testing a digital circuit will consist of applying successive sets of values to the primary inputs, and of observing the resulting values appearing at the primary outputs. Each individual test, consisting of a set of input values together with the corresponding set of correct, fault-free output values, is known as a test pattern or test vector. A complete sequence of test patterns is called a test set. The central problem of testing is the derivation of an adequate test set for any particular circuit. This process is described as test pattern generation.

Built-in self-test (BIST) denotes the ability of a circuit or system to test itself. It can potentially eliminate the need for external test equipment and introduces the capability for testing devices after the circuit is integrated in a system in the field (online testing). BIST is gaining acceptance in the VLSI industry because of its many advantages [1]. For BIST in general, test patterns are generated on-chip by a test pattern generator (TPG) and the responses of the circuit under test (CUT) is compressed and analyzed by an on-chip signature analyzer.

There are generally three strategies of testing:

- Exhaustive
- Deterministic
- Pseudorandom

In exhaustive testing, all 2^n input combinations for an n-input CUT are applied during the test process. Exhaustive testing does not face the problem of random pattern resistant faults, and it achieves 100% stuck-at fault coverage [2]. However, test time becomes too long for CUTs with large number of inputs. Test time can be reduced by partitioning the CUT into subcircuits and then testing each subcircuit exhaustively (pseudoexhaustive method). However, the fault coverage for pseudoexhaustive testing is lower.

Another category of testing techniques is based on deterministic test set embedding. In deterministic test, the circuit under test (CUT) is analyzed prior to testing to determine the appropriate test set to be applied. After the test set has been obtained, the TPG is designed to generate the predetermined test set. Various techniques have been proposed to obtain the best seeds and LFSR characteristic polynomials to cover a deterministic test set. However, these approaches can only be applied to circuits with small size or regular structures due to high computational complexity of the search procedures [1].

Pseudorandom testing is popular due to the simplicity of the linear feedback shift registers (LFSRs) used as TPGs to generate a subset of the 2^n test patterns. Fault coverage is estimated by probabilistic methods. The number of random patterns to be applied is decided by the detection probability of the faults [3].

In terms of TPG hardware realization, two opposite architectures with respect to area overhead and testing time are the ROM based architectures and the counter-based architectures. The later architecture uses a ROM to store the vectors generated by an external automatic test

*Chek T Lim, Ng, Kok Hong, Ng, Sheng Kwang are with Intel Sdn. Bhd. Penang.

pattern generator (ATPG). High fault coverage and short testing times can be achieved. But the area overhead introduced by this method is in general prohibitive for practical applications.

In counter-based architectures, the test patterns are generated by a counter, which introduces a small area penalty. The main disadvantage of this method is that long test sequences may be required to achieve an acceptable fault coverage, which results in longer testing time.

The most used BIST architectures are the LFSR based architectures [4]. For these architectures, good fault coverage can be achieved in most cases and several techniques have been proposed to reduce the test time or to increase the fault coverage of LFSR based architectures.

This paper is organized in the following way. A brief review on previous architectures of LFSR based TPG is presented in part II. The proposed architecture of an LFSR-based TPG that is used to generate pseodorandom patterns is presented in part III. Initial experimental results are explained in part IV. Future work is highlighted in part V and conclusion in part VI.

II. PREVIOUS WORK

Wang and McCluskey discussed the theory of both standard and modular LFSRs. A hybrid TPG design of standard and modular LFSRs is also proposed. This design is used to generate maximum length sequences and can be reconfigured to include the all zeros state for exhaustive testing. Compared to the standard or modular LFSR that uses m 2-input XOR gates, the hybrid design uses only (m+1)/2 XOR gates [5].

Chen and Gupta proposed a TPG design called "input reduction". The proposed design partitions the inputs of a CUT into groups, similar to pseudo-exhaustive testing, where each group corresponds to a test signal. However, unlike pseudo-exhaustive testing, which only combines unrelated inputs (inputs that are not in the same cone) into a test signal, the proposed technique analyzes the CUT to determine compatible inputs to be combined into a test signal, even if the inputs belong to the same cone [1].

Dufaza and Chevalier presents a TPG design composed of a simplified LFSR associated with an OR network and a set of multiplexers, called an LFSROM. The design is able to generate a precomputed set of deterministic test vectors obtained with an ATPG tool [6]. It provides a relatively simple way of generating deterministic test vectors without having to use complex procedures. It is also able to produce the all-zero's test vector without the need for additional circuitry (unlike conventional LFSR). However, the use of LFSROM requires extra circuitry, especially for the MUX and its related selector circuitry. Once implemented, the set of test patterns produced is fixed and unchangeable except by rewiring the OR gates network. This means that once embedded in the chip, the test patterns cannot be changed.

Shi and Zhang presents a reseeding technique for LFSR-based test pattern generation, which can generate pseudorandom vectors in normal mode while also being able to produce the seed of a group of test vectors in jumping mode [7].

Wang and Lee present an LFSR-based TPG that can accelerate the application of deterministic patterns from the LFSR to a scan chain. The target scan chain is divided into multiple sub-chains and an LFSR-based multiple sequence generator is used to generate all the subsequences required by the sub-chains. A generalized relationship between the bits in the original scan chain and the state of the LFSR is derived such that the bits generated by an LFSR in any future clock cycle can be pre-generated by the proposed TPG [8].

III. PROPOSED ARCHITECTURE

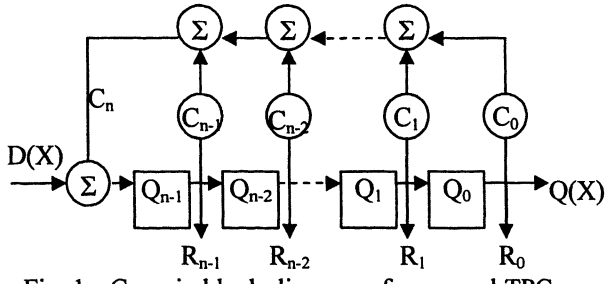

Fig. 1 Generic block diagram of proposed TPG structure.

The proposed TPG structure is presented in Figure 1. The structure is based on that of a conventional LFSR signature analyzer. The TPG operates on two clock signals; one is the normal clock for the LFSR registers and another clock signal, which is connected to the input D(X). The

D(X) clock is set at a lower frequency from the register clock.

One characteristic of this TPG structure is that it can generate longer sequences of the same set of pseudorandom test patterns. A TPG with an n stage LFSR can generate a maximum set of $2^n - 1$ number of test vectors. In conventional LFSR TPGs, this set of test vectors will then produces a maximum sequence length of $2^n - 1$ patterns, before the sequence repeats itself. In the proposed TPG, this maximum length sequence can be extended to beyond $2^n - 1$ patterns with the manipulation of the D(X) clock. The total number of test vectors is still a maximum of $2^n - 1$, but there will be a duplicate of certain vectors within the same sequence. Table 1 shows an example of the simulated sequence generated by a conventional LFSR as compared to the proposed TPG.

LFSR clock	Conventional LFSR sequence	Proposed TPG sequence, t=2
	$Q_2Q_1Q_0$	$Q_2Q_1Q_0$
0	1 0 0	1 0 0
1	1 1 0	0 1 0
2	1 1 1	0 1 1
3	0 1 1	1 0 1
4	1 0 1	1 1 0
5	0 1 0	1 1 1
6	0 0 1	1 1 1
7	1 0 0	0 1 1
8	1 1 0	0 0 1
9	1 1 1	1 0 0
10	0 1 1	0 1 0
11	1 0 1	0 0 1
12	0 1 0	0 0 0
13	0 0 1	0 0 0
14	1 0 0	1 0 0
15	1 1 0	0 1 0
16	1 1 1	0 1 1
17	0 1 1	1 0 1
18	1 0 1	1 1 0
19	0 1 0	1 1 1
20	0 0 1	1 1 1
21	1 0 0	0 1 1
22	1 1 0	0 0 1
23	1 1 1	1 0 0
24	0 1 1	0 1 0
25	1 0 1	0 0 1
26	0 1 0	0 0 0
27	0 0 1	0 0 0

Table 1 Maximum length sequences generated by TPG

The test patterns in Table 1 are generated by a TPG using a standard LFSR with a characteristic polynomial of X^3+X^2+1 and a seed value of $Q_2Q_1Q_0 = 100$. The frequency of the D(X) clock is half of the register clock ($f = ½$ or $t = 2$).

From the table, we see that the maximum length sequence of the conventional LFSR is 7 vectors. For the proposed TPG, the sequence extends to a total of 14 vectors before the sequence repeats itself.

Another thing to note is that the proposed TPG is able to generate the all zeros state autonomously and is not stuck in the all zeros state.

IV. PRELIMINARY RESULTS

Preliminary testing has revealed a relationship between the register/D(X) clock ratio and the maximum length of the sequence generated by the proposed TPG.

Taking the frequency of the register clock as f, and the frequency of the D(X) clock as $f_{D(X)}$, the ratio of the two clocks is $r = f_{reg}/f_{D(X)}$.

Table 2 shows some different values of r and their corresponding sequence lengths, which were obtained through simulation. In this example, the LFSR used has the characteristic polynomial of X^3+X^2+1 with a maximum length sequence of 7.

R	Sequence length
Conventional LFSR	7
2	14
3	21
4	28
5	35
6	42
7	14
8	56
10	70
14	14
21	21
28	28

Table 2: Different values of r and their respective sequence lengths

Preliminary results suggest that for values of r that is not multiples of 7 (the maximum length of a conventional sequence), the corresponding sequence length is a multiple of r and 7, which is the length of the conventional sequence. For $r = 7$, the sequence

length is 14, while for r that is multiples of 7, the sequence length is equals to r. Though the number of test sequence is higher, it comes at the cost of testing speed which is now slower.

While the proposed TPG is capable of cycling through the all zeros state, it does have its own trivial sequences which will cause it to be "stuck" in a similar manner that a conventional LFSR is stuck at the all zero state.

The occurrence of these trivial sequences is on condition of certain initial states at different values of r. Figure 2 shows the state diagram of an LFSR to illustrate how this can happen.

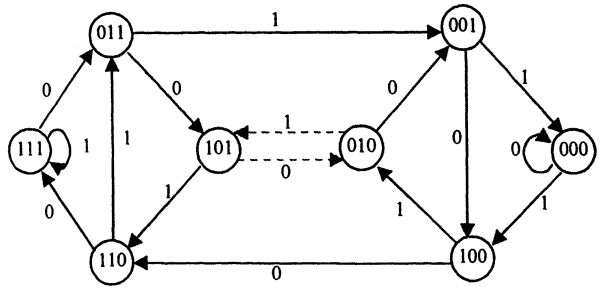

Fig 2 State diagram of LFSR with characteristic polynomial $X^3 + X^2 + 1$

If the initial state of the LFSR is $Q_2Q_1Q_0$ = 101, and at the first register clock, the input at D(X) is 0, then the next LFSR state will be $Q_2Q_1Q_0$ = 010. If at the next register clock the input at D(X) is 1, then the LFSR will switch back to 101. Subsequently, an alternating stream of 0's and 1's will cause the LFSR to be stuck within these two states.

For an r of 2 (D(X) frequency is half of register clock frequency), the input at D(X) is effectively an alternating stream of 0's and 1's, as shown in Figure 3 below.

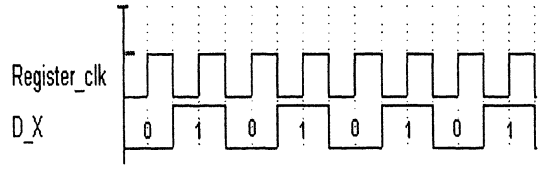

Fig 3 Positive edge register clock with D(X) clock at half frequency

The TPG operating at different values of r will have different sets of trivial sequences, some of which is shown in Figure 4.

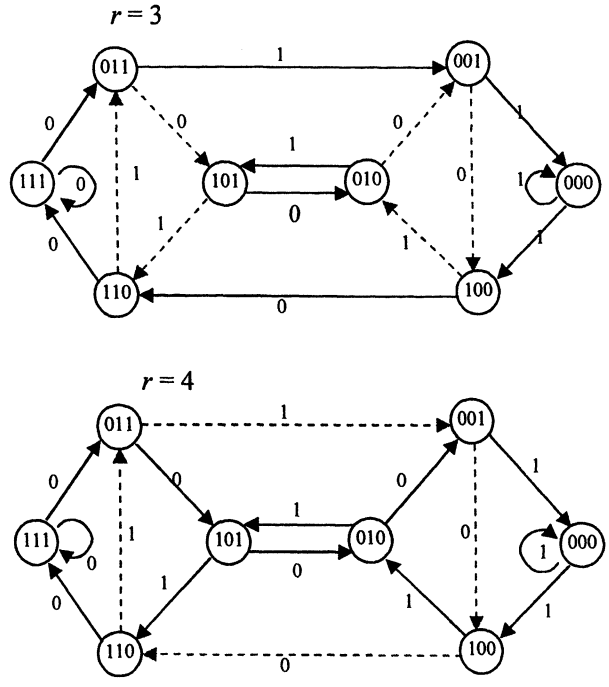

Fig 4 Trivial sequences at different values of r

V. Further Work

Further work on this proposed TPG will involve further testing using different lengths of LFSRs. LFSRs with non-primitive polynomials will also be tried out.

At present, the D(X) uses symmetrical clock signals as input. In future, the input at D(X) will be tried out with asymmetrical input signals to observe their effects on the LFSR output.

The occurrence of trivial sequences should also be looked into. Seed values that will cause trivial sequences should be identified so that they will be avoided.

Towards the end, the performance of the TPG will be evaluated by testing it on benchmark circuits.

VI. Conclusion

TPG methods, which can generate longer sequences of test patterns, have been proposed. This TPG architecture is derived from an LFSR based signature analyzer, by manipulating the input of the analyzer. Preliminary results have shown a relationship between frequencies of the

LFSR register clock and the input signal to the analyzer.

ACKNOWLEDGEMENT

Thank you to Intel Penang for providing research fund and professional guidance.

REFERENCES

[1] C.-A. Chen and S.K. Gupta, "A methodology to design efficient test pattern generators", *Proceedings of International Test Conference*, pp. 814 – 823 (1995)

[2] I. Hamzaoglu, J.H. Patel, "Reducing test application time for Built-in-Self-Test test pattern generators", *VLSI Test Symposium, 2000. Proceedings. 18th IEEE*, pp. 369–375 (2000)

[3] B. Vasudevan, D.E. Ross, M. Gala and K.L. Watson, "LFSR based deterministic hardware for at-speed BIST", *Digest of Papers, Eleventh Annual 1993 IEEE VLSI Test Symposium,* pp. 201 – 207 (1993)

[4] P. Flores, H. Neto, K. Chakrabarty and J. Marques-Silva, "Test Pattern Generation For Width Compression in BIST", *Proceedings of the 1999 IEEE International Symposium on Circuits and Systems, 1999. ISCAS '99*, Volume: 1, pp. 114 -118 vol.1 (1999)

[5] L.-T. Wang and E.J. McCluskey, "Hybrid designs generating maximum-length sequences", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Volume: 7, Issue: 1 pp. 91 – 99 (1988)

[6] C. Dufaza and C. Chevalier, "LFSROM Basic Principle and BIST Application", *Design Automation, 1993, with the European Event in ASIC Design. Proceedings. [4th] European Conference on*; pp 211 –216 (1993)

[7] Y. Shi and Z.Zhang, "Multiple Test Set Generation Method for LFSR-Based BIST", *Proceedings of the ASP-DAC 2003. Asia and South Pacific Design Automation Conference*, 2003, pp. 863 –868 (2003)

[8] W.L. Wang and K.J Lee, "Accelerated Test Pattern Generators for Mixed-Mode BIST Environments", *Proceedings of the Ninth Asian Test Symposium 2000 (ATS 2000)*, pp. 368 - 373 (2000)