LFSR Design using Low Transition for BIST

R.Sree santhi
VLSI Design, India

Abstract: **The basic idea of this project is to implement a low transition LFSR that generates test patterns with improved correlation between the adjacent bits. The improved correlation between the adjacent bits of test patterns reduces the switching activity in the circuit. The reduced switching activity results in low power dissipation. In the past, the major concerns of the VLSI designer were area, performance, cost and reliability; power consideration was mostly of only secondary importance. In recent years, however, this has begun to change and, increasingly, power is being given comparable weight to area and speed considerations. Several factors have contributed to this trend. Perhaps the primary driving factor has been the remarkable success and growth of the class of personal computing devices (portable desktops, audio- and video-based multimedia products) and wireless communications systems (personal digital assistants and personal communicators) which demand high-speed computation and complex functionality with low power consumption.**

Keywords: **Built-In Self-Test, LFSR technique, low-power pattern generation, Low power LFSR.**

INTRODUCTION

The main challenging areas in VLSI are performance, cost, testing, area, reliability and power. The demand for portable computing devices and communications system are increasing rapidly. These applications require low power dissipation for VLSI circuits. The power dissipation during the test mode is 200% more than in normal mode. Hence it is important aspect to optimize power during testing. Power optimization is one of the main challenges. There are various factors that affect the cost of chip like packaging, application, testing etc. In VLSI, according to thumb rule 50% of the total integrated circuits cost is due to testing.

BUILT-IN SELF-TEST

A built-in self-test (BIST) or built-in test (BIT) is a mechanism that permits a machine to test itself. Engineers design BISTs to meet requirements such as:

- high reliability
- lower repair cycle times

BIST is commonplace in weapons, avionics, medical devices, automotive electronics, complex machinery of all types, unattended machinery of all types, and integrated circuits.

BIST ARCHITECTURE

It is very important to choose the proper LFSR architecture for achieving the appropriate fault coverage. Every architecture consumes different power even for same polynomial. Another problem associated with choosing LFSR is LFSR design issue, which includes LFSR partitioning, in this the LFSR are differentiated on the basis of hardware cost and testing time cost.

A typical BIST architecture consists of a test pattern generator (TPG), usually implemented as a linear feedback shift register (LFSR), a test response analyzer (TRA), implemented as a multiple input shift register (MISR), and a BIST control unit (BCU), all implemented on the chip (Figure1). This approach allows applying at-speed tests and eliminates the need for an external tester. The BIST architecture components are given below.

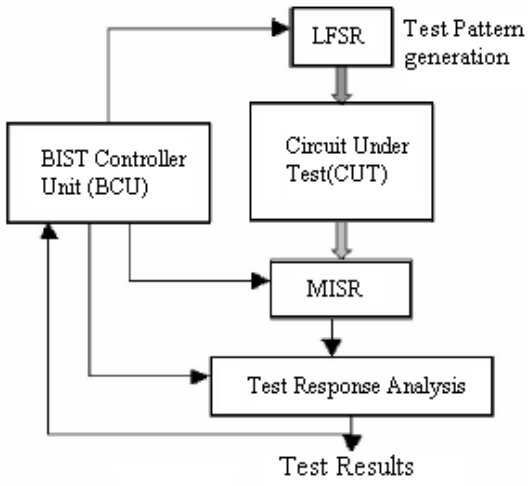

Figure 1.1 BIST Architecture

BIST PATTERN GENERATION

There are various methods and approaches have been used to generate test patterns during BIST. This can be described in brief below:

- Linear Feedback Shift Register is used to generate pseudo-random test patterns. This normally requires a sequence of one million or more tests pattern in order to achieve high fault coverage. One of the advantages of LFSR is it uses very little hardware and thus is currently the preferred BIST pattern generation method. In this project, LFSR is being chosen as the test pattern generation method.
- A binary counter can generate an exhaustive but not randomized test sequences. The Drawback of binary counters as the pattern generator is, it requires more hardware than typical LFSR pattern generator.
- Modified counters also have been successfully as test-pattern generators. However, they also require long test sequences.
- This method stores a good test pattern set from an ATPG program in a ROM on the chip. However, drawback of this approach is relatively expensive in chip area.
- In this method, each pattern generator cell has a few logic gates, a flip-flop, and connections only to neighboring gates. The cell is replicated to produce the cellular automaton.

LINEAR FEEDBACK SHIFT REGISTER

A linear feedback shift register (LFSR) is a shift register whose input bit is a linear function of its previous state. The only linear function of single bits is XOR, thus it is a shift register whose input bit is driven by the exclusive-or (XOR) of some bits of the overall shift register value .The initial value of the LFSR is called the seed, and because the operation of the register is deterministic, the stream of values produced by the register is completely determined by its current (or previous) state. Likewise, because the register has a finite number of possible states, it must eventually enter a repeating cycle. However, an LFSR with a well-chosen feedback function can produce a sequence of bits which appears random and which has a very long cycle.

Applications of LFSRs include generating pseudo-random numbers, pseudo-noise sequences, fast digital counters, and whitening sequences. Both hardware and software implementations of LFSRs are common.

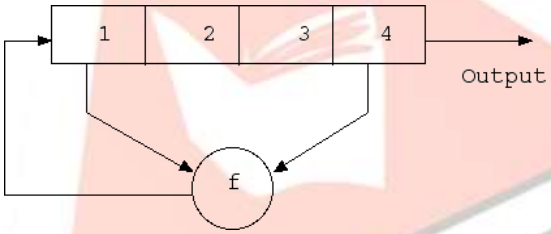

Figure 1.2 4-Bit LFSR, Tap Sequence;

MAXIMAL LENGTH TAP SEQUENCES

LFSR's can have multiple maximal length tap sequences. A maximal length tap sequence also describes the exponents in what is known as a primitive polynomial mod 2. For example, a tap sequence of 4, 1 describes the primitive polynomial x^4 + x^1 + 1. Finding a primitive polynomial mod 2 of degree n (the largest exponent in the polynomial) will yield a maximal length tap sequence for an LFSR that is n bits long.

There is no quick way to determine if a tap sequence is maximal length. However, there are some ways to tell if one is not maximal length:

- Maximal length tap sequences always have an even number of taps.
- The tap values in a maximal length tap sequence are all relatively prime. A tap sequence like 12, 9, 6, 3 will not be maximal length because the tap values are all divisible by 3.

Discovering one maximal length tap sequence leads automatically to another. If a maximal length tap sequence is described by [n, A, B, C], another maximal length tap sequence will be described by [n, n-C, n-B, n-A]. Thus, if [32, 3, 2, 1] is a maximal length tap sequence, [32, 31, 30, 29] will also be a maximal length tap sequence. An interesting behaviour of two such tap sequences is that the output bit streams are mirror images in time.

LOW TRANSITION LFSR

This paper presents a new test pattern generator for low power BIST and scan-based BIST architectures. The proposed technique increases the correlation in two dimensions, i.e. vertical dimension between test patterns (Hamming Distance) and horizontal dimension within each pattern (adjacent bits in one pattern) as well.

Here, we propose a random pattern generator that combines two methods of test pattern generation called R-Injection (RI) and Bipartite LFSR. Briefly, the RI method inserts a new intermediate pattern between two consecutive test patterns by positioning a random-bit (R) in the corresponding bit of the intermediate pattern when there is a transition between corresponding bits of pattern pairs. The Bipartite LFSR generates an intermediate pattern using one half of each of the two consecutive random patterns. The main advantage of our proposed technique is that it can be used for both combinational and sequential circuits and the randomness quality of patterns does not deteriorate.

RANDOMNESS IN BIPARTITE LFSR

The implementation of a LFSR can be changed to improve some design features such as power during test. However, such modification may change the order of patterns or insert new patterns that affect the overall randomness. For example, suppose that T^iandT^{i+1} are two consecutive patterns generated by an n-bit LFSR. The maximum number of transitions will be n when T^i and T^{i+1} are complement of each other. One strategy used to reduce number of transitions to maximum of n/2 is to insert a pattern T^{i1} half of which is identical $T^i$to and T^{i+1} as show in figures 1.3 and 1.4

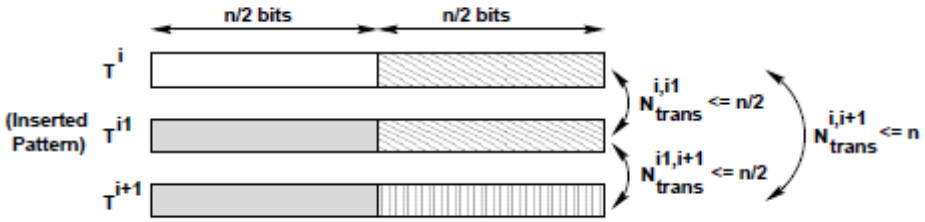

Figure 1.3 Pattern insertion based on bipartite strategy

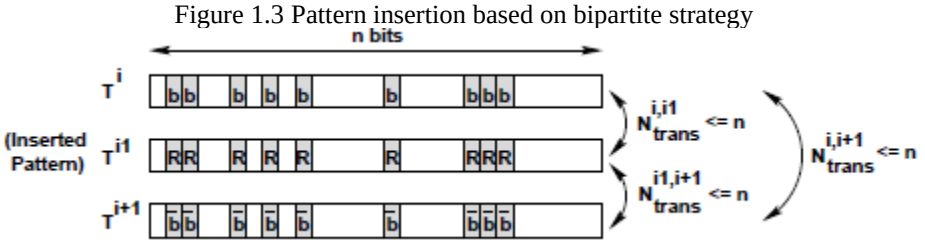

Figure 1.4 Pattern insertion based on Random Injection strategy

The Bipartite strategy guarantees the transition change to be limited to n/2 between two consecutive patterns.

RANDOMNESS IN RI-LFSR

To preserve the randomness of patterns, instead of Bipartite strategy we randomly inject a value in bit positions where $T_j^i \neq T^{i+1}{}_j$. Briefly

$$T^{i1}{}_j = T^i{}_j \text{ if } T_j^i = T^{i+1}{}_j$$
$$T^{i1}{}_j = R \text{ if } T_j^i \neq T^{i+1}{}_j$$

Figure 3 shows this symbolically. The shaded cells show those bit positions where $T_j^i \neq T^{i+1}{}_j$.We insert a random bit (shown as R in T^{i1}) if the corresponding bits in T^iandT^{i+1} are different. Note that since such bits are uniformly distributed and also we replace them with another random value the overall randomness remains unchanged, i.e. H_{max}=n. The circuitry for RI is shown in figures 1.5 and 1.6.

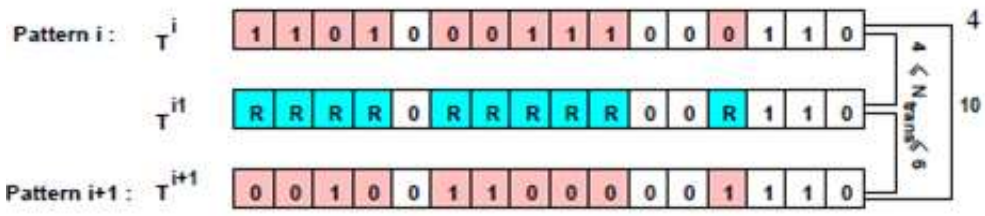

Figure 1.5 Example of RI (Random Injection)

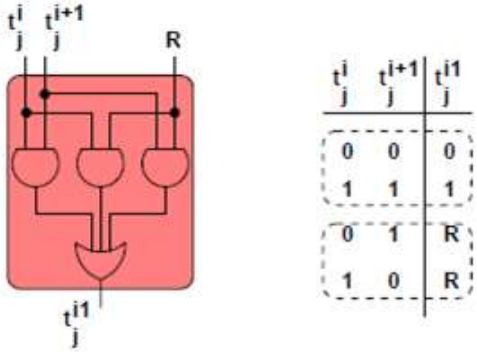

Figure 1.6 RI circuit

IMPLEMENTING LT-LFSR ARCHITECTURE

We combine our two proposed techniques of pattern generation (RI and Bipartite LFSR) for low-power BIST. The new low transition LFSR (LT-LFSR) generates three intermediate patterns (T^{i1}, T^{i2} and T^{i3}) between T^{i} and T^{i+1}. We embed these two techniques into a bit-sliced LFSR architecture to create LTLFSR which provides more power reduction compared to having only one of the R-Injection and Bipartite LFSR techniques in a LFSR. This may seem to prolong test session by a factor of 4. However, due to high randomness of the inserted patterns many of the intermediate patterns can do as good as patterns generated by a LFSR in terms of fault detection.

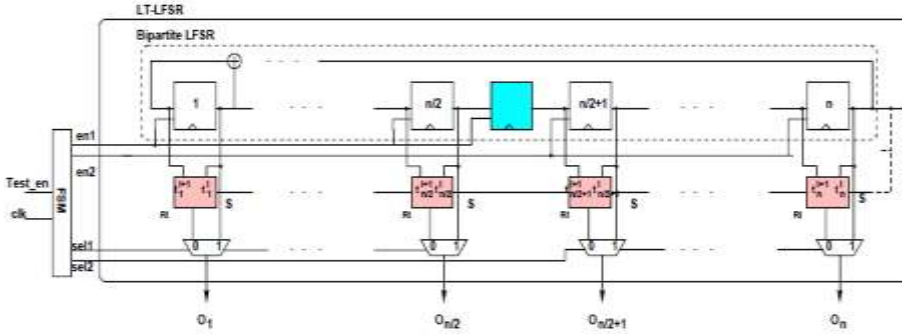

Figure 1.7 Low transition LFSR

Figure 1.7 shows LT-LFSR with RI and Bipartite LFSR included. The LFSR used in LT-LFSR is an external-XOR LFSR. As shown, injector circuit taps the present state (T^{i} pattern) and the next state (T^{i+1} pattern) of LFSR.Signalsen_1 and en_2select half of the LFSR to generate random patterns .Multiplexers select either the injection bit or the exact bit in LFSR. One very small finite state machine (FSM) controls the pattern generation process as follows:

Step 1: en_1en_2=10, sel_1sel_2=11. The first half of LFSR is active and the second half is in idle mode. Selectingsel_1sel_2=11, both halves of LFSR are sent to the outputs (O_1 to O_n). In this case, T^{i} is generated.

Step 2: en_1en_2=00, sel_1sel_2=10. Both halves of LFSR are in idle mode. The first half of LFSR is sent to the outputs (O_1 to $O_{n/2}$), but the RI injector circuit outputs are sent to the outputs ($O_{n/2+1}$ to O_n). T^{i1} is generated.

Step 3: en_1en_2=01, sel_1sel_2=11. The second half of LFSR works and the first half of LFSR are in idle mode. Both halves are transferred to the outputs (O_1 to O_n) and T^{i2} is generated.

Step 4: en_1en_2=00, sel_1sel_2=01. Both halves of LFSR are in idle mode. From the first half the injector outputs are sent to the outputs of LT-LFSR (O_1 to $O_{n/2}$) and the second half sends the exact bits in LFSR to the outputs($O_{n/2+1}$ to On) to generate T^{i3}.

Step 5: The process continues by going through Step 1 to generate T^{i+1}.

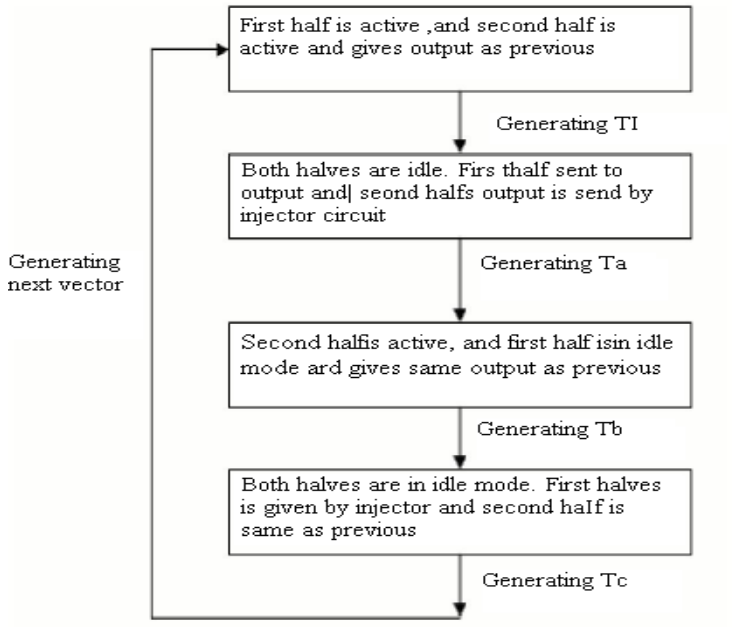

Figure 1.8 Low power LFSR Algorithm

LOW POWER TEST PATTERN GENERATION

One way to improve the correlation between the bits of the successive vectors is to avoid frequent transitioning of the logic levels of the primary inputs. The new approach entails inserting 3 intermediate vectors between every two successive

vectors. The total number of signal transitions between these 5 vectors is equal to the total number of signal transitions between the 2 successive vectors generated using the conventional approach. This reduction of signal transition activity in the primary inputs reduces the switching activity inside the design under test and therefore results in reduced power Consumption by the device under test. The additional circuitry used to accomplish the generation of the 3 intermediate vectors is minimal at best consisting of few logic gates.

The number of LFSR outputs required is driven by the number of test inputs required for circuit under test. The technique of inserting 3 intermediate vectors is achieved by modifying the conventional LFSR circuit with two additional levels of logic between the conventional flip-flop outputs and the low power outputs as shown in Figure 6.1. The first level of hierarchy from the top down includes logic circuit design for propagating either the present or the next state of the flip-flops to the second level of hierarchy. The second level of hierarchy is a multiplexer function that provides for selecting between the two states (present or next) to be propagated to the outputs as low power output.

The outputs of the flip-flops are loaded with the seed vector. The feedback taps are selected pertinent to the characteristic polynomial $x^n + x + 1$. Only 2 inputs pins, namely test enable and clock are required to activate the generation of the pattern as well as simulation of the design circuit. It is also noteworthy here that the intermediate vectors in addition to aiding in reducing the number of transitions can also empirically assist in detecting faults just as good as the conventional LFSR patterns.

RESULTS

The power consumption of LFSR and LT-LFSR themselves used in the benchmarks. The power consumption of the RI of LT-LFSR, including its FSM and LFSR. Depending on the size, the power consumption of LT-LFSR is less than the same size of LFSR.

CONCLUSION

It is observed that the total power consumed in low transition LFSR and the power consumed in conventional LFSR is 54% less than the power consumption by conventional LFSR. It is concluded that low power LFSR is very useful for BIST implementation in which the CUT may be Combinational, sequential and memory circuits. Using low power LFSR technique we can further decrease the power in BIST implementation.

References

[1] E. Atoofian, S. Hatami, Z. Navabi, M. Alisaface and A. Afzali-Kusha,"A New Low-Power Scan-Path Architecture," IEEE International Symposium, Vol.5, pp.5278 - 5281, 23-26 May 2005.
[2] The National Technology Roadmap for Semiconductors (ITRS), 1997 Edition. Semiconductor Industry Association.
[3] Michael L.Bushnell, VishwaniD.Agawal," Essentials of electronic testing for digital, memory and mixed-signal VLSI circuits," Kluwer Academic.
[4] Mohammad Tehranipoor, MehrdadNourani, Nisar Ahmed," Low-Transition LFSR for BIST-Based Applications," 14th Asian Test Symposium, pp. 138- 143, 18-21 Dec. 2005.
[5] FulvioCorno, Paolo Prinettom, MatteoSonzarEorda "Testability analysis and ATPG on behavioural RT-level VHDL," IEEE International Test Conference, pp 753-759, 1997.
[6] F. Corno, P. Prinetto, M. Rebaudengo, M. SonzaReorda," A Test Pattern Generation methodology for low power consumption," pp.1-5, 2008.
[7] ShileshMalliyoor, Chao You," Comparison of hardware implementation and power consumption of low-power multiple output linear feedback shift register," Journal of engineering, computing and architecture, 2007.
[8] Jinkyu Lee and Nur A. Touba," LFSR-Reseeding Scheme Achieving Low-Power dissipation during Test," IEEE transactions on computer aided design of integrated circuits and systems, 26(2), February 2007.
[9] YervantZorianl, SujitDey, Michael J. Rodgers "Test of Future System on-Chips "ICCAD 2000.
[10] P.Glard et al "Survey 0 F Low-Power Testing of VLSI Circuits "IEEE Design & Test Of Computers, vol. 19, no. 3. (2002), pp. 80-90.