Automated Synthesis of Configurable Two-dimensional Linear Feedback Shifter Registers for Random/Embedded Test Patterns

Chien-In Henry Chen and *Kiran George*
Department of Electrical Engineering
Wright State University
Dayton, Ohio 45435, USA

Abstract

A new approach to optimize a configurable two-dimensional (2-D) linear feedback shift registers (LFSR) for both embedded and random test pattern generation in built-in self-test (BIST) is proposed. This configurable 2-D LFSR based test pattern generator generates: 1) a deterministic sequence of test patterns for *random-pattern-resistant* faults, and then 2) random patterns for *random-pattern-detectable* faults. The configurable 2-D LFSR test generator can be adopted in two basic BIST execution options: *test-per-clock* (parallel BIST) and *test-per-scan* (serial BIST). Experimental results of test-per-clock BIST for benchmark circuits show with the configurable scheme the number of flip-flops of 2-D LFSR is reduced by 79%. The average number of faults detected by configurable 2-D LFSR is 9.27% higher than the conventional LFSR. Experimental results of test-per-scan BIST for benchmark circuits demonstrate the effectiveness of the proposed technique in which high fault coverage can be achieved.

1. Introduction

BIST has been proven to be an effective design-for-testability technique to achieve sufficiently high fault coverage in which testing is accomplished through built-in hardware for test generation and response analysis. There are typically two basic BIST execution options: 1) *test-per-clock* (parallel BIST) and 2) *test-per-scan* (serial BIST). In test-per-clock BIST, test patterns are applied to the circuit under test (CUT) by the test generator and the responses are captured by response analyzer every clock cycle. In *test-per-scan* BIST, test patterns are shifted into a serial scan path or multiple scan paths to test the CUT. The test responses are subsequently captured by the scan flip-flops and shifted out to the response analyzer while new patterns are being shifted in.

There have been various test generation techniques based on pseudo-random testing [1,9], pseudo-exhaustive testing [2,10,11], weighted random testing [3,12,13], and reseeding of LFSR [5-7]. Justification of use of any of these techniques includes test application time, test length, test storage, hardware overhead, and fault coverage. The major limitations with these techniques are the test length, the fault coverage, and the complexity of implementation. Several approaches have been proposed to overcome these disadvantages, one of which is to embed a set of pre-computed test vectors, called deterministic patterns, to reach high fault coverage within a shorter time. Such deterministic patterns can be obtained by an automatic test pattern generation tool and implemented with a ROM to store the patterns, or the traditional LFSR with pre-designed seeds to generate the patterns. However, a number of weakness and problems still remain unsolved, which has a great impact on testing cost and performance. It is expected that easy and regular structure for implementation will result in a reduction of the product development cycle and cost, and the cost of system maintenance can be significantly reduced.

A configurable 2-D LSFR based test generator is proposed in this paper, which first generates a sequence of pre-computed test patterns that detect random-pattern-resistant (or hard-to-detect) faults, and then generates random patterns for detecting random-pattern-detectable faults. The configurable 2-D LSFR can be easily adopted in *test-per-clock* BIST (see Fig. 1(a)) and *test-per-scan* (see Fig. 1(b)). With the proposed configuration scheme the hardware overhead of 2D LFSR [8] can be considerably reduced.

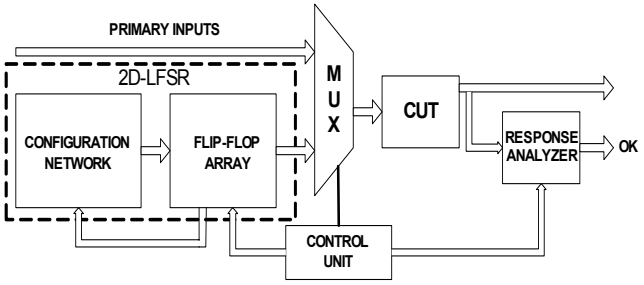

Fig. 1(a). Test-per-clock BIST.

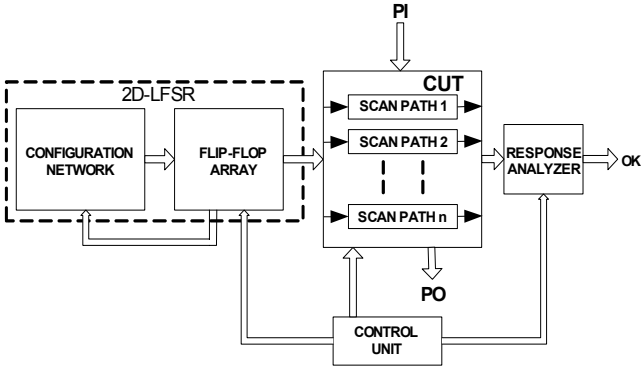

Fig. 1(b). Test-per-scan BIST.

2. 2-D LFSR

A conventional LFSR can be used to generate a large number of pseudo-random test patterns with very small area overhead that is composed of exclusive-or gates (XORs) and flip-flops (FFs). The general structure is shown in Fig. 2, which can be expressed with the polynomial in Eq. (1). The coefficients *Ci (i = 1~M-1)* are zero or one.

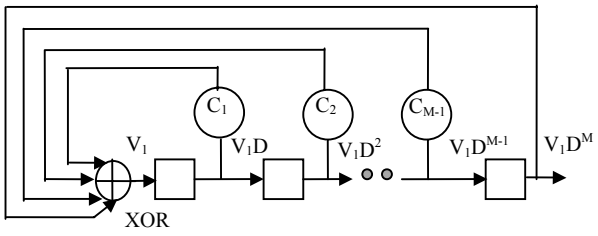

Fig. 2. A conventional LFSR.

$$P(V) = 1 + C_1V + C_2V^2 + ... + C_{M-1}V^{M-1} \qquad (1)$$

An LFSR goes through a cyclic sequence of states and the outputs produced are also periodic. The maximum length of this period is $2^M - 1$, where M is the number of stages. So the patterns generated by an LFSR are pseudo-random and the random properties depend on its initial state and its generating function $P(V)$. Once the generating function and the initial state are fixed, the patterns produced are also determined. It cannot produce a sequence of deterministic ordered patterns that are required to detect the random-pattern-resistant faults. A number of the LFSR based techniques [5-7] have been developed to generate deterministic ordered test patterns but the encoding and decoding procedures and the implementation of control logic are relatively complex. This 2-D LFSR can generate a set of pre-computed test vectors for detecting random-pattern-resistant faults. In addition, it can generate better random patterns than a conventional LFSR. Fig. 3 shows the general representation of a 2-D LFSR based on N primitive polynomials:

$$V_i = \sum_{j=1}^{N} \sum_{k=1}^{M} a_{ijk} V_j D^k + C_i \ , \quad i = 1 \sim N \qquad (2)$$

This circuit consists of N shift registers, each of which has M stages. V_i *(i=1~N)* represents an N-bit vector, and V_iD^k *(k=1~M)* represents the k^{th} delay of a vector V_i. V_i is generated by the feedback network given by Eq. (2). If a_{ijk}=1, the signal V_jD^k is connected to the XOR gate to generate V_i. Otherwise, there is no connection. If C_i=0, the output of the XOR gate is connected to the shift register directly. If C_i=1, then an inverter is added to the input of the shift register for generating a complementary feedback signal. Given L vectors with a length of N-bit,

V_1: $b_{11}, b_{12}, b_{13}, ..., b_{1L}$
V_2: $b_{21}, b_{22}, b_{23}, ..., b_{2L}$
...
...
V_N: $b_{N1}, b_{N2}, b_{N3}, ..., b_{NL}$.

To generate these vectors in a 2-D LFSR with minimum stage M, we construct Eq. (3) and obtain the solutions by determining the minimum value of M.

$$V_i(n) = \sum_{j=1}^{N} \sum_{k=1}^{M} a_{ijk} V_j(n-k) + C_i \qquad (3)$$

where $V_i(n) = b_{i,n}$, $V_j(n-k) = V_j(n)D^k = b_{j,(n-k)}$, i = 1~N and n = 1~L. When solving the first vector $V_i(1)$, $V_j(1-k)$ represents the initial states of the 2-D LFSR.

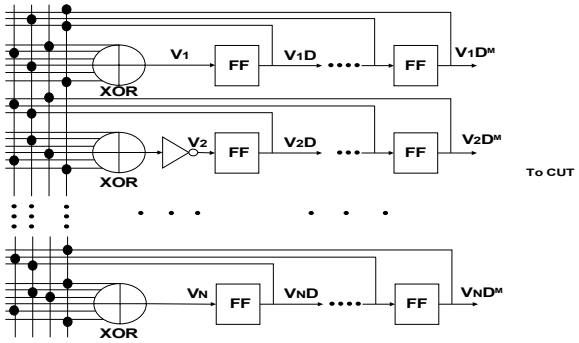

Fig. 3: 2-D LFSR.

An optimization to Eq. (3) is described below:

```
{ M = 1;  /* M is the no. of  flip-flop stages */
      Set initial states of 1st stage FF's; /* <1010…> */
      Given all Vi's (i=1 to N); /* each Vi is an N-bit vector*/
      While ( a_ijk for all Vi's are not solved ) {
        if ( a_ijk  for  V_i  is solved) return; /* j=1 to N,
                                                    k=1 to M */
        for (each unsolved  V_i ) {
          Set  C_i = 0 ;  /* no inverter added */
          Solve Eq. (3);
           if ( a_ijk  values are found) {
                  mark signal  V_i  solveds;
           }
          else {
            Set  C_i = 1 ;  /* inverter added */
            Solve Eq. (3);
            if ( a_ijk  values are found) {
                  mark signal  V_i  solved;
            }
        } /* unsolved signal  V_i */
        for (each  V_i )   {
          if  (signal  V_i  marked unsolved) {
            M = M+1;
            Set initial states of M-th stage FF's; /* alternate 1
                                and 0 in each column of the FFA
                */
          }
      } /* time frame M = M+1 */
} /* end */
```

Example 1: A sequence of 16 patterns is embedded in 2-D LFSR.

Seq:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	0	1	0	0	0	0	1	1	1	0	0	0	0	1	0
2	0	1	1	0	1	0	0	1	1	0	0	0	0	0	1	0

3	1	1	1	1	0	0	0	1	0	0	1	1	0	0	1	1
4	0	1	1	1	1	0	1	0	1	0	0	1	1	0	1	0
5	0	0	1	1	1	0	1	0	0	1	0	1	0	1	0	1
6	1	0	1	1	1	1	0	0	0	1	1	1	0	1	0	1

The synthesized results are shown below and the synthesized 2-D LFSR is as shown in Fig. 4.

$$V_1 = V_1D^2 + V_3D^2 + V_4D^2 + V_2D + V_4D + V_5D$$
$$V_2 = V_1D^2 + V_2D^2 + V_5D^2 + V_6D^2 + V_1D + V_2D + V_6D$$
$$V_3 = V_1D^2 + V_2D^2 + V_6D^2 + V_2D + V_3D + V_4D + V_5D$$
$$V_4 = V_2D^2 + V_4D^2 + V_4D + V_5D + V_6D$$
$$V_5 = V_5D^2 + V_6D^2 + V_1D + V_2D$$
$$V_6 = V_1D^2 + V_3D^2 + V_2D + V_4D + V_6D$$

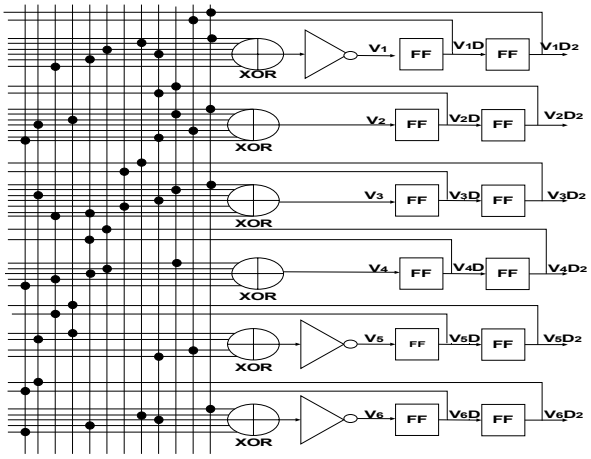

Fig. 4. 2-D LFSR for example 1.

The longer the deterministic sequence is, the more stages of a 2-D LFSR are required. For some practical circuits, a large overhead is unacceptable. In the next section, a feasible configuration scheme based on a 2-D LFSR to reduce the overhead is proposed.

3. Configurable 2-D LFSR

A configurable 2-D LFSR based test pattern generator is proposed to generate an embedded deterministic sequence of test patterns followed by pseudo-random patterns. The generator mainly consists of four types of function blocks – the flip-flop array (FFA), the configuration networks (CN), the multiplexers (MUXs), and the control unit (CU) as shown in Fig. 5. The FFA is an $N \times M$ flip-flop array, where N is the number of inputs of a circuit under test (CUT) and M is the number of stages of the 2-D LFSR. To reduce the hardware, M is usually a small number. If Eq. (3) cannot be solved with a small value of M, the embedded test sequence is then partitioned into multiple subsequences embedded by a configurable scheme. Each CN consists of XOR gates and an inverter if necessary. The MUX selects one of the p configuration networks to feed the feedback signals to the FFA. The MUX is controlled by the CU. When resetting the generator, the initial states of FFA are set to <1010…>, alternating 1 and 0, in each column of the FFA.

The synthesis procedure for a configurable 2-D LFSR has two main tasks: (1) find a set of deterministic ordered patterns and (2) embed these patterns to the configurable 2-D LFSR. With the deterministic sequence of test patterns, all coefficients a_{ijk}'s of Eq. (3) are to be solved by the optimization procedures. The number of stages, M, in the 2-D LFSR reflects the hardware complexity. Given a small value of M, the solution space may not be large enough to obtain all a_{ijk}'s of Eq. (3); therefore, partitioning the original sequence of test patterns into consecutive multiple subsequences are required. As the length of each subsequence decreases, Eq. (3) can be solved with a small M.

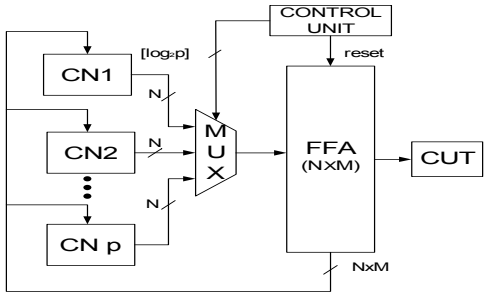

Fig. 5. Configurable 2-D LFSR.

Example 2: Two consecutive subsequences of 18 patterns are embedded in 2-D configurable LFSR.

Seq:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
1:	1	0	1	0	0	0	0	1	1	1	0	0	1	0	1	1	1	0
2:	0	1	1	0	1	0	1	1	1	0	0	0	1	0	0	0	0	1
3:	1	1	1	1	0	0	0	1	0	0	1	1	0	0	0	1	0	0
4:	0	1	1	1	1	0	0	0	1	0	0	1	1	0	0	0	1	0
5:	0	0	1	1	1	0	1	1	0	1	0	1	1	1	0	0	1	0
6:	1	0	1	1	1	1	0	0	0	1	1	1	1	1	0	1	0	1

Subsequence 1 Subsequence 2

The synthesized results are shown below and the synthesized 2-D LFSR is as shown in Fig. 6.

$$V_1 = V_4D + V_5D$$
$$V_2 = V_2D + V_6D$$
$$V_3 = V_2D + V_3D + V_4D \quad \text{(Subsequence 1)}$$
$$V_4 = V_1D + V_2D + V_5D$$
$$V_5 = V_3D + V_5D + V_6D$$
$$V_6 = V_1D + V_2D + V_3D + V_4D + V_5D$$

$$V_1 = V_1D + V_5D$$
$$V_2 = V_2D + V_4D$$
$$V_3 = V_1D + V_2D + V_5D + V_6D \quad \text{(Subsequence 2)}$$
$$V_4 = V_3D$$
$$V_5 = V_2D + V_3D$$
$$V_6 = V_1D + V_3D$$

Example 2 shows the effectiveness of the configurable 2-D LFSR based generator. Given a deterministic sequence of 18 6-bit patterns with the initial seed <101010>, both a configurable and a non-configurable 2-D LFSRs are generated. Fig. 6 shows the configurable 2-D LFSR with one

stage of FFA, $M = 1$ and $p = 2$. Feeding the initial seed <101010> back to FFA through the configuration network CN1 generates the first test pattern <101001>. Other patterns are generated subsequently till the ninth pattern <110100> is generated. In this configurable test generator, the CN1 generates the first 9 patterns of subsequence 1, <101001>, <011100>, <111111>, <001111>, <010111>, <000001>, <010010>, <111010> and <110100>. The CN2 generates the next 9 patterns of subsequence 2, <100011>, <001001>, <001111>, <110111>, <000011>, <100000>, <101001>, <100110> and <010001>. The tenth pattern is generated with the ninth pattern <110100> of subsequence 1 feeding through the CN2. The CU is a modulo-9 counter controlling this switch. Optimizing the feedback network is equivalent to assigning minimal number of 1's to a_{ijk}'s of Eq. (3). An optimized configurable 2-D LFSR is shown in Fig. 7. Without configuration, 6 stages of FFA are required.

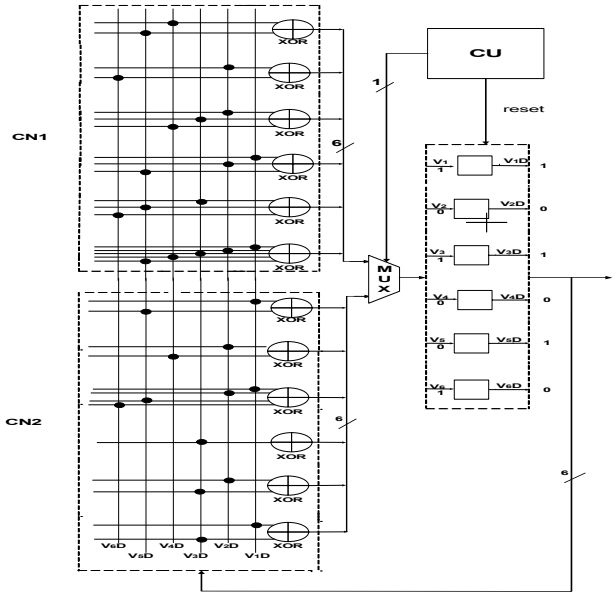

Fig. 6: A configurable 2-D LFSR for example 2.

Applying the configuration scheme to example 1 results in the synthesized 2-D LFSR of only one stage of flip-flops and three configuration networks, which is shown in Fig. 8.

Seq:	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1	0	1	0	0	0	0	1	1	1	0	0	0	0	1	0
2	0	1	1	0	1	0	0	1	1	0	0	0	0	0	1	0
3	1	1	1	1	0	0	0	1	0	0	1	1	0	0	1	1
4	0	1	1	1	1	0	1	0	1	0	0	1	1	0	1	0
5	0	0	1	1	1	0	1	0	0	1	0	1	0	1	0	1
6	1	0	1	1	1	1	0	0	0	1	1	1	0	1	0	1
	Subsequence 1						Subsequence 2						Subsequence 3			

$$
\begin{aligned}
V_1 &= V_4D + V_5D \\
V_2 &= V_2D + V_6D \\
V_3 &= V_2D + V_3D + V_4D \qquad \text{(Subsequence 1)} \\
V_4 &= V_1D + V_2D + V_5D \\
V_5 &= V_3D + V_5D + V_6D \\
V_6 &= V_1D + V_2D + V_3D + V_4D + V_5D
\end{aligned}
$$

$$
\begin{aligned}
V_1 &= V_1D + V_5D \\
V_2 &= V_2D + V_4D \\
V_3 &= V_1D + V_3D + V_4D \qquad \text{(Subsequence 2)} \\
V_4 &= V_2D + V_4D + V_5D + V_6D \\
V_5 &= V_4D + V_5D + V_6D \\
V_6 &= V_1D + V_3D
\end{aligned}
$$

$$
\begin{aligned}
V_1 &= V_1D + V_3D + V_6D \\
V_2 &= V_1D + V_3D + V_6D \\
V_3 &= V_3D + V_6D \qquad \text{(Subsequence 3)} \\
V_4 &= V_6D \\
V_5 &= V_4D + V_6D \\
V_6 &= V_1D + V_3D + V_4D
\end{aligned}
$$

4. Experimental Results

Five synthesized benchmark circuits are used to evaluate the configuration structure of the 2-D LFSR scheme. The characteristics of the five circuits are summarized in Table I. First, we applied the configurable 2-D LFSR for test-per-clock BIST of the benchmark circuits. In Table II, the configuration and the non-configuration results of 2-D LFSR for benchmark circuits are compared. The second column is the number of embedded deterministic patterns detecting the random-pattern-resistant faults for each circuit. The third and fourth columns give the stage number of shift registers and the total number of flip-flops for the non-configurable test generator. The number of configurations, the stage number of shift registers, and the number of flip-flops for the configurable 2-D LFSR are given in the fifth, sixth and seventh columns. The synthesis results of *am2910* and *mul16* in Table II show the number of flip-flops is significantly reduced from 260 to 40 (84.62%) and from 252 to 36 (85.71%), respectively. Because the number of embedded patterns for *div16*, *pcont2* and *piir8* is small, the synthesis procedure results in no reconfiguration. Among the benchmark circuits, *div16* requires the largest number of flip-flops (330) due to its large number of primary inputs (33). The number of flip-flop's stage (M) for *div16* can be reduced from 10 to 2 by increasing the number of iterations while solving Eq. (3). The number of flip-flop's is also reduced by 81.82%, from 330 to 60. Table III shows the comparison of the test generation and fault simulation. The third and fourth columns show the number of single stuck-at faults and ATPG results of HITEC [4] where "Det" denotes the number of detected faults and "Vec" denotes the number of test patterns. The fault simulation of the conventional LFSR is summarized in the fifth and sixth columns. Columns seven to ten summarizes the fault simulation of 2-D LFSR and configurable 2-D LFSR. As shown in Table III the average number of equivalent faults is 9,441 where 3,955 faults are detected by "HITEC". Comparing "LFSR" and "2-D LFSR", the average number of detected faults by "2-D LFSR" is 5,952 which is 8.65 % higher than 5,478 of "LFSR" and 50.49% higher than 3,955 of "HITEC". The average number of faults detected by “Configurable 2-D LFSR” is 5,986, which is 0.57% higher than 5,952 of the 2-D LFSR. The average number of flip-flop's required by "Configurable 2-D LFSR” is 187 which is reduced by 78.92% in comparison with 887 flip-flop's of "2-D LFSR".

Next, we applied the configurable 2-D LFSR for test-per-scan BIST to the benchmark circuits that were optimized by Synopsys Design Compiler. In Table IV, the number of configurations, the stage number of shift registers, and the number of flip-flops for the configurable 2-D LFSR are given in the second, third and fourth columns. Columns 5 to 9 list the number of scan paths, scan chain patterns embedded in 2-D LFSR, total faults, detected faults, and redundant faults.

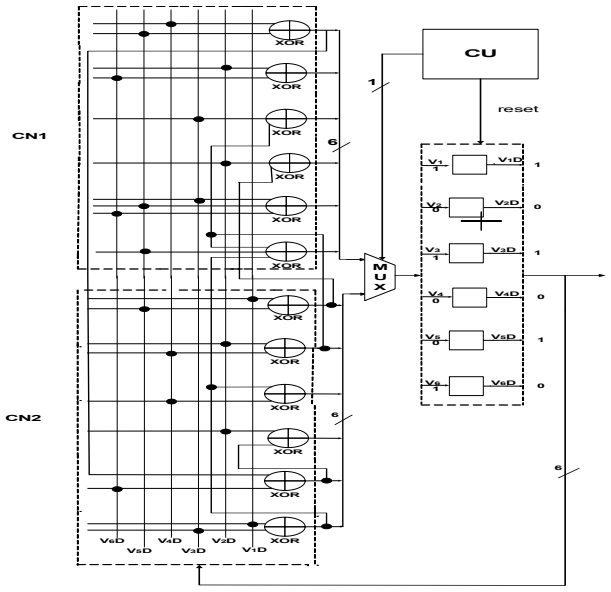

Fig. 7: An optimized configurable 2-D LFSR for example 2.

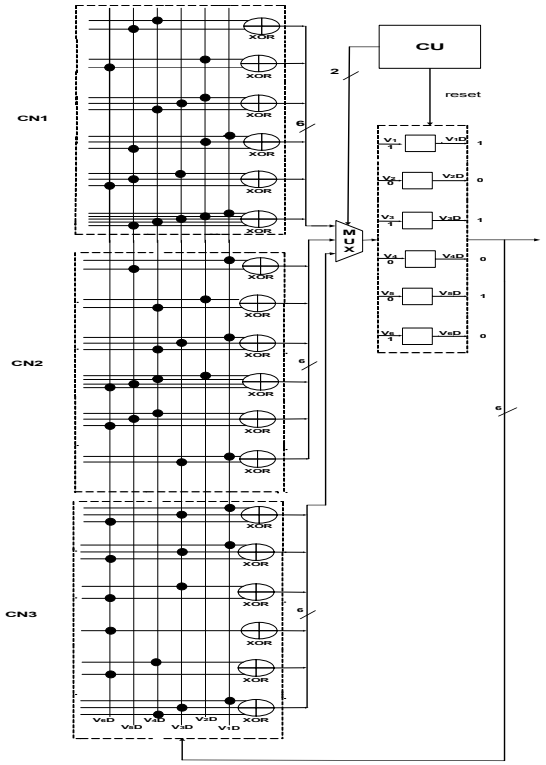

Fig. 8 A configurable 2-D LFSR for example 1.

5. Conclusion

A configurable 2-D LFSR based test generator and an automated synthesis procedure have been presented. Without storage of test patterns, a 2-D LFSR based test pattern generator can generate a sequence of pre-computed test patterns (detecting random-pattern-resistant faults) and followed by random patterns (detecting random-pattern-detectable). The hardware overhead is decreased considerably through configuration. To improve the fault coverage and reduce the hardware overhead, a minimal set of deterministic test patterns detecting random-pattern-resistant faults is needed. Choosing different partition of the embedded sequence of deterministic test patterns may result in different reconfiguration networks and generate different sequence of random patterns. Furthermore, logic optimization is necessary to reduce the hardware of the configuration networks.

6. References

[1] P. Bardell, W.H. McAnney and J. Savir, Built-in Test for VLSI: Pseudorandom Techniques, Wiley & Sons, New York, 1987.

[2] C.-I. H. Chen and J. Yuen, "Automated Synthesis of Pseudo-Exhaustive Test Generator in VLSI BIST Design", *IEEE Trans. on VLSI Systems*, Vol. 2, No. 3, pp. 273-291, Sept. 1994.

[3] F. Brglez, et al., "Hardware-Based Weighted Random Pattern Generation for Boundary-Scan", *Proc. IEEE Int. Test Conf.*, pp. 264-274, 1989.

[4] T. M. Niermann and J. H. Patel, "HITEC: A Test Generation Package for Sequential Circuits," *Proc. European Conf. on Design Automation*, pp. 213-218, 1991.

[5] S. Hellebrand, S. Tarnick, J. Rajski, and B. Courtois, "Generation of Vector Patterns through Reseeding of Multiple-Polynomial Linear Feedback Shift Registers", *Proc. IEEE Int. Test Conf.*, pp. 120-129, 1992.

[6] S. Venkataraman, J. Rajski, S. Hellebrand, and S. Tarnick, "An Efficient BIST Scheme Based on Reseeding of Multiple Polynomial Linear Feedback Shift Registers," *Proc. Int. Conf. on Computer-Aided Design*, pp. 572-577, 1993.

[7] S. Hellebrand, et al., "Built-In Test for Circuits with Scan Based on Reseeding of Multiple-Polynomial Linear Feed-back Shift Registers", *IEEE Trans. on Comp.*, Vol. 44, No. 2, pp. 223-233, Feb. 1995.

[8] X. Yuan and C.-I. H. Chen, "Automated Synthesis of a Multiple-Sequence Test Generator Using 2-D LFSR", *Proc. IEEE International ASIC Conf.*, pp. 75-79, 1998.

[9] K. D. Wagner, C. K. Chin, and E. J. McCluskey, "Pseudorandom Testing," *IEEE Trans. on Comp.*, Vol. 36, No. 3, pp. 332-343, 1987.

[10] D. Kagaris, F. Makedon, and S. Tragoudas, "A Method for Pseudo-Exhaustive Test Pattern Generation," *IEEE Trans. on CAD*, Vol. 13, No. 9, pp. 1170-1178, 1994.

[11] H.-J. Wunderlich and S. Hellebrand, "The Pseudo-exhaustive Test for Sequential Circuits," *IEEE Trans. on CAD*, Vol. 11, No. 1, pp. 26-32, 1992.

[12] R. Kapur, S. Patil, T. J. Snethen, and T. W. Williams, "Design of an Efficient Weighted Random Pattern Generation System," *Proc. ITC*, pp. 491-500, 1994.

[13] F. Muradali, V. K. Agarwal, and B. Nadeau-Dostie, "A New Procedure for Weighted Random *TC*, pp. 660-669, 1990.

Table I Characteristics of five synthesized circuits

Circuits	*Seq. Depth*	*FFs*	*PIs*	*POs*
am2910	4	87	20	16
mult16	9	55	18	33
div16	19	50	33	34
pcont2	3	24	9	8
piir8	5	56	9	8

Table II Hardware comparison of 2-D LFSR and configurable 2-D LFSR

Circuits	*Pat-tern*	*2D LFSR*		*Configurable 2D LFSR*		
		Stgs (M)	Flip-flops	Cnfg. (p)	Stgs (M)	Flip-flops
am2910	30	13	260	2	2	40
mul16	72	14	252	4	2	36
div16	21	10	330	1	2	66
pcont2	7	2	18	1	2	18
piir8	19	3	27	1	3	27
Total	149	-	887	-	-	187

Table III Test-per-clock BIST results of five synthesized circuits

Ckts	*Total*	*HITEC*		*LFSR*		*2-D LFSR*		*Configurable 2-D LFSR*	
	Faults	*Det*	*Vec*	*Det*	*Vec*	*Det*	*Vec*	*Det*	*Vec*
Am2910	2391	2142	726	2126	2^{17}	2188	2^{17}	2179	2^{17}
mult16	1708	1568	89	665	2^{17}	1605	2^{17}	1605	2^{17}
div16	2147	1676	190	1679	2^{17}	1705	2^{17}	1719	2^{17}
pcont2	11272	3147	7	6727	2^{17}	6336	2^{17}	6518	2^{17}
piir8	29689	11234	19	16195	2^{17}	17925	2^{17}	17913	2^{17}
Avg.	9441	3955	209	5478	2^{17}	5952	2^{17}	5986	2^{17}

Table IV Test-per-scan BIST results of five synthesized circuits

Circuits	Configurable 2-D LFSR							
	Confg.	*Stages*	*Flip-flops*	*No. of scan paths*	*Scan chain patterns*	*Total faults*	*Detected faults*	*Redundant faults*
am2910	40	6	66	11	318	2206	2206	0
mult16	18	4	32	8	175	1689	1689	0
div16	36	4	32	8	287	1841	1841	0
pcont2	11	5	40	8	131	2671	2663	4
piir8	61	3	21	7	486	5607	5561	21