

## Cyclic-CPRS : A Diagnosis Technique for BISTed Circuits for Nano-meter Technologies

Chun-Yi Lee, Hung-Mao Lin, Fang-Min Wang, and James Chien-Mo Li

Laboratory of Dependable Systems (LaDS)  
Graduate Institute of Electronics Engineering  
National Taiwan University  
Taipei City, 10617 Taiwan, R.O.C.  
Email: cmli@cc.ee.ntu.edu.tw

**Abstract** - A Cyclic-CPRS (Column Parity Row Selection) technique is presented to diagnose built-in self tested (BISTed) circuits, even in the presence of many unknowns and transient errors. The novel cyclic scan chains retain the transient errors and unknowns in the CUT until they are fully diagnosed. Instead of masking the unknowns, Cyclic-CPRS directly diagnoses the unknowns as if they were errors. Direct diagnosis of unknowns not only eliminates the masking circuitry but also enhances the diagnosis resolution. Experimental results show that Cyclic-CPRS is very successful even in the presence of 10% errors and unknowns. The proposed technique is especially suitable for nano-meter technologies, in which transient errors and systematic defects are becoming serious problems.

### I. Introduction

In the System-on-Chip (SOC) era, one popular solution to test embedded cores is Built-In Self Test (BIST). If an embedded core fails a test, diagnosis requires two pieces of information: the failing patterns and the failing locations. The failing patterns are the patterns for which errors occur. An error is a mismatch between the expected circuit outputs and the actual circuit outputs. The failing locations are two dimensional indexes of failing scan chains and failing scan cells where errors are observed. Once the failing patterns and failing locations are available, scan-based diagnosis techniques (such as [1]) can be applied to identify the faults. Diagnosing embedded or BISTed cores faces two major obstacles: limited I/O pins and unknown circuit output values. Due to the fast test speed and the large number of patterns applied, it is very difficult to dump all the circuit outputs to identify the failing patterns in BIST mode.

Traditionally, limited accessibility to CUT outputs is the major obstacle to diagnose BISTed circuits. In nano-meter technologies, we face even greater challenges because systematic defects become a more serious problem than random defects [2]. Unlike random defects, systematic defects oftentimes induce a relatively large number of errors. The errors can be scattered in different locations of the CUT. In addition to large error multiplicity, two issues make diagnosis even more difficult: transient errors and the unknowns. *Transient errors* are incorrect CUT signals that vary from time to time even when the same test pattern is applied [3][4]. This is because certain defects in nano-meter technologies, such as IR drop and crosstalk, are not deterministic and can only be provoked in certain operation conditions [5][6][7]. Transient errors can lead to confusing and sometimes wrong diagnosis results. *Unknowns* (*X's*),

such as floating tri-state nets or bus contention, are caused by random patterns applied in BIST mode [8]. Unknowns corrupt the MISR signature and spoil the diagnosis information. A simple solution to the X problem is to add X-masking circuitry, which prevents X's from getting into MISR. Unfortunately, the masking circuitry also degrades the diagnosis resolution.

Past BIST diagnosis solutions have limitation in meeting the challenges of embedded core diagnosis. In the software category, existing BIST diagnosis solutions include reciprocal polynomials [9], diagonal matrix for MISR [10], and MISR quotients [11][12]. Pure software techniques have difficulty in diagnosing multiple errors due to aliasing and masking of signature analyzers [13]. In the hardware category, a simple BIST diagnosis solution is to bypass the compressor and dump scan chain contents [14][15][16]. However, the hardware overhead increases rapidly with the number of scan chains, which is oftentimes very large for BISTed CUT. Also, it is difficult to identify the failing patterns due to the large number of patterns applied. Other hardware diagnosis solutions include error control codes [17][18], cycling registers [19][20] serial LFSR of different polynomials [21], programmable MISR [22][23], random selection LFSR [24][25][26][27], and counter-based selection [28]. X-tolerant test and diagnosis solutions include the X-compact [29], row/column parity [30], and convolutional compactors [31][32]. However, the diagnosis resolution degrades rapidly as the number of unknowns increase. Specifically, if one hundred scan chains are compressed into one single output, fewer than 15% errors are correctly diagnosed in the presence of 1% unknowns [31].

Cyclic-CPRS is presented to meet the diagnosis requirements for embedded cores in SOC. Cyclic-CPRS is a revised version of the original CPRS [33] with two major features added: *cyclic scan chains* and *direct diagnosis of unknowns*. Cyclic scan chains retain the contents of scan chains by feeding back the output of every scan chain to its own input. Cyclic scan chains hence "lock" unknowns in feedback loops until all errors are fully diagnosed. The cyclic scan chains enable multiple-session diagnosis without the interference of unknowns. The second innovation is direct diagnosis of unknowns. Instead of masking the unknowns, Cyclic-CPRS "faces" the unknowns and diagnoses them in the same way as errors. Compared to the traditional masking solution, the direct diagnosis approach not only eliminates the masking circuitry but also improves the diagnosis resolution. The direct diagnosis approach is especially effective for errors located in the scan chains that

generate unknowns. Experimental results show that Cyclic-CPRS diagnosis is 100% correct even when up to 10% scan cells are erroneous or unknown. The penalty of Cyclic-CPRS includes area overhead and the observation of the column parity after every scan clock.

The organization of this paper is as follows. The second section presents the Cyclic-CPRS diagnosis hardware and software. The third section shows our experimental results. Finally, section four concludes this paper.

## II. Cyclic-CPRS

### A. Hardware Architecture

The Cyclic-CPRS hardware architecture is shown in Fig. 1. Cyclic scan chains have two modes of operation. In BIST mode, cyclic scan chains function as regular scan chains. In diagnosis mode, the scan outputs are fed back to their own scan inputs. At the same time, the scan outputs from  $m$  scan chains are randomly selected by the row selection LFSR (RS-LFSR) every scan cycle. The row selection hardware is made up of  $m$  AND gates. After the row selection hardware, the scan outputs are XORed together to produce one bit of column parity (CP), which is observed every scan cycle in the diagnosis mode. The selected scan outputs are also XORed with their previous row parities (RP), which are accumulated in the row parity registers (RP-registers). The row parity registers are connected into a scan chain and are shifted out after all  $l$  scan cells are unloaded. (This chain is not drawn in the figure for clarity.) All the scan flip-flops (FF), RS-LFSR, and RP-registers are triggered by the same clock. Please note that the cyclic scan chains are of the same length to ensure correct diagnosis. If the scan chains in the original design are not of the same length, dummy scan cells have to be padded to equalize the scan chain length.

### B. Diagnosis Data Collection Flow

A complete Cyclic-CPRS diagnosis consists of multiple diagnosis snapshots, which in turn consists of several diagnosis sessions. A diagnosis snapshot has only one system clock, which captures the circuit responses in cyclic scan chains. The contents of cyclic scan chains within a diagnosis snapshot remain unchanged so the unknowns are correctly diagnosed before the next diagnosis snapshot. A diagnosis session unloads the cyclic scan chains with an RS-LFSR seed. The RS-LFSR seeds of every diagnosis session in the same diagnosis snapshot must be unique so that the row and column parities do not repeat.

Figure 2 shows the Cyclic-CPRS diagnosis data collection flow with two nested loops. The outer loop (step 2 to step 7) is the diagnosis snapshot and the inner loop (step 3 to step 6) is the diagnosis session. Every step is explained as follows.

1. Run a regular BIST and observe the row and column parities to identify failing patterns.
2. Start BIST all over again and pause after loading a particular failing pattern.
3. Load an RS-LFSR seed and reset the row parity register.
4. Unload the scan chains. The RS-LFSR and row parity registers are also clocked while unloading the scan

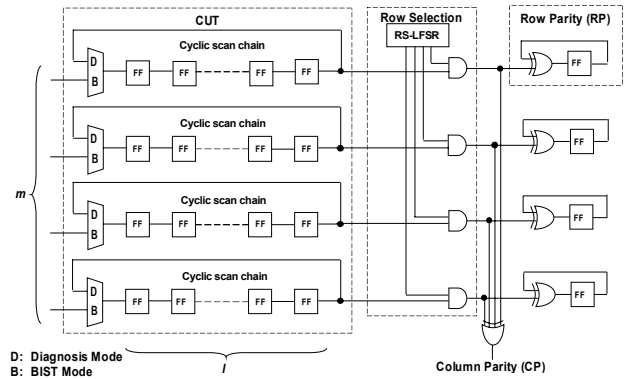


Fig. 1. Cyclic-CPRS Architecture

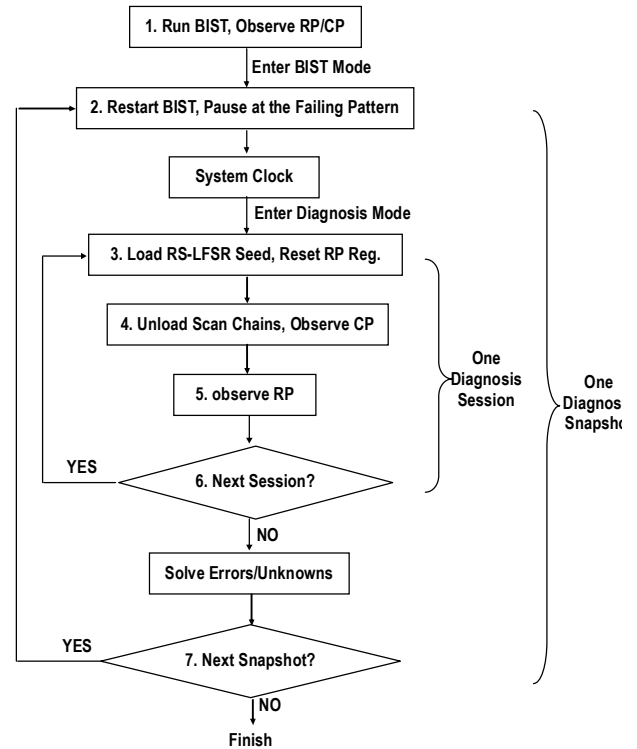


Fig. 2. Cyclic-CPRS Diagnosis Data Collection Flow

chains. One bit of CP is observed every scan clock. The scan outputs are fed back to their scan inputs.

5. After unloading all scan cells, the row parities are shifted out and observed.
6. If there are more diagnosis sessions to do, the procedure returns step 3 to load a new RS-LFSR seed. If there is no more diagnosis session, this diagnosis snapshot is finished. The errors and unknowns in this diagnosis snapshot can now be solved.
7. If more diagnosis snapshots are required, the procedure goes back to step 2. If not, the whole diagnosis is finished.

### C. Solving Errors and Unknowns

Within a diagnosis snapshot, the cyclic scan chain contents are represented by an *error matrix* (E). Every row in the error matrix corresponds to a scan chain and every element corresponds to a scan cell. Figure 3 shows an error matrix in which  $m$  equals to four and  $l$  equals to five. Each element  $E_{i,j}$

in the error matrix is called an *error variable*. A failing scan cell in the  $j_{th}$  scan cell of the  $i_{th}$  scan chain is represented by a one in the corresponding error variable ( $E_{i,j} = 1$ ). A passing scan cell is represented by a zero in the corresponding error variable ( $E_{i,j}=0$ ).

Cyclic-CPRS directly diagnoses the unknowns rather than masking or skipping them. On the silicon, every captured X can be either one or zero. No matter which value is captured, the cyclic scan chains keep every X unchanged within a diagnosis snapshot. Initially, all X's are assumed to be zeros. If the captured X is zero, then this scan cell is regarded as error-free ( $E_{i,j}=0$ ). If the captured X is one, the scan cell is regarded as an error ( $E_{i,j}=1$ ).

The *selection matrix* is of the same size as the error matrix ( $m \times l$ ). A one in the selection matrix means the corresponding scan cell is selected by the row selection hardware; a zero in the selection matrix means the corresponding scan cell is masked. Figure 3 shows an example of the selection matrix of size 4 x 5. The *selected error matrix*, SEM, is obtained by multiplying the error matrix with the selection matrix. Note that the multiplication is performed in a bit-by-bit way so the SEM is also of the size  $m \times l$ .

The *error row parity* (ERP) is a column vector of size  $m$ . The ERP represents the difference of the gold row parity and the observed row parity. The *gold row parity* is the expected row parity obtained from simulation, assuming all unknowns are zeros. A one in the  $i_{th}$  row of ERP means that the row parity of the  $i_{th}$  scan chain is different from its expected value. Similarly, the *error column parity* (ECP) is a row vector of size  $l$ , which represents the difference between the gold column parity and the observed column parity. The number  $r$  is the weight of the ERP and  $c$  is the weight of the ECP. The number  $r$  represents the total number of mismatches observed in the row parity while  $c$  represents the total number of mismatches observed in the column parity. Suppose that  $E_{1,3}$  in Fig. 3 is an error and  $E_{3,4}$  is an unknown. The ECP and the ERP are therefore  $[0 \ 0 \ 1 \ 1 \ 0]$  and  $[1 \ 0 \ 1 \ 0]^T$ , respectively. The weight of ECP ( $c$ ) and the weight of ERP ( $r$ ) are both 2.

$$r = \text{weight}(\text{ERP}) \quad (1)$$

$$c = \text{weight}(\text{ECP}) \quad (2)$$

The reduced SEM (RSEM) is obtained from SEM by deleting the error-free rows and columns. The reason is to reduce the number of error variables involved in the equations. The RSEM is therefore of size  $r \times c$ . The  $i_{th}$  row equation is obtained by summing all elements in the  $i_{th}$  row of the RSEM, as in (3). The  $j_{th}$  column equation is obtained by summing every element in the  $j_{th}$  column of the RSEM, as in (4). A total of  $r$  row equations and  $c$  column equations are derived from a RSEM. The row equations and column equations form a system of linear equations  $Ae=B$ , where A is the *coefficient matrix*.

$$\text{row\_equations}(i) = \sum_{j=1}^c \text{RSEM}_{i,j} = \text{ERP}_i \quad (3)$$

$$\text{column\_equations}(j) = \sum_{i=1}^r \text{RSEM}_{i,j} = \text{ECP}_j \quad (4)$$

Figure 4 shows an example to find row and column

Error Matrix					Selection Matrix					Selected Error Matrix				
$E_{1,1}$	$E_{1,2}$	$E_{1,3}$	$E_{1,4}$	$E_{1,5}$	0	0	1	0	0	0	0	$E_{1,3}$	0	0
$E_{2,1}$	$E_{2,2}$	$E_{2,3}$	$E_{2,4}$	$E_{2,5}$	0	1	1	1	1	0	$E_{2,2}$	$E_{2,3}$	$E_{2,4}$	$E_{2,5}$
$E_{3,1}$	$E_{3,2}$	$E_{3,3}$	$E_{3,4}$	$E_{3,5}$	1	0	1	1	1	$E_{3,1}$	0	$E_{3,3}$	$E_{3,4}$	$E_{3,5}$
$E_{4,1}$	$E_{4,2}$	$E_{4,3}$	$E_{4,4}$	$E_{4,5}$	0	1	1	1	0	0	$E_{4,2}$	$E_{4,3}$	$E_{4,4}$	0

Fig. 3. Error Matrix, Selection Matrix, and Selected Error Matrix

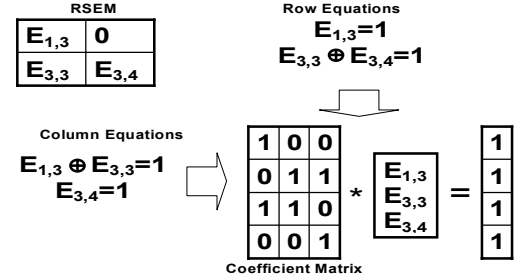


Fig. 4. Derivation of Row and Column Equations

equations. The first, second, and fifth columns are deleted from the SEM because the ECP is  $[0 \ 0 \ 1 \ 1 \ 0]$ . The second and fourth rows are also deleted because the ERP is  $[1 \ 0 \ 1 \ 0]^T$ . The 2 x 2 RSEM produces two row equations and two column equations. The coefficient matrix is of the size 4 x 3. Solving the system of linear equations produces a unique solution —  $E_{1,3} = 1$ ,  $E_{3,3} = 0$ , and  $E_{3,4} = 1$ .  $E_{1,3}$  is a correctly solved error.  $E_{3,4}$  is not an error but an unknown of which the initial guess value “0” differs from the actual captured value “1”.

The preceding example demonstrates how the error variables are solved for a single diagnosis session. If there are multiple diagnosis sessions within a diagnosis snapshot, the above process is repeated with a distinct SM for every diagnosis session. The row/column equations for those rows/columns that fail at least one session are put together in the system of linear equations. Figure 5 summarizes the flow to solve the error variables for multiple diagnosis sessions of one diagnosis snapshot. Please note that, for multiple diagnosis snapshots, every snapshot is represented by an independent error matrix. Every snapshot has to be handled independently because the error matrices can be different in the presence of transient errors.

It is not always the case that all error variables are uniquely solved in the above manner. If an error variable is not uniquely solved, its diagnosis result is ambiguous. Of course all errors have to be uniquely solved but the unknowns do not. To diagnose ambiguous errors, at least one deterministic diagnosis is needed. The *deterministic diagnosis* finds a deterministic RS-LFSR seed such that new linear independent equations can be added to the existing system of linear equations. More details about the deterministic diagnosis can be found in [33].

### III. Experimental Results

Experimental results are shown to demonstrate the effectiveness of the Cyclic-CPRS technique. Each of the following experiment is performed on 10,000 randomly generated error matrices with various *error multiplicity* and *unknown multiplicity* (UM). Error multiplicity is the total number of errors injected into the error matrices. Unknown

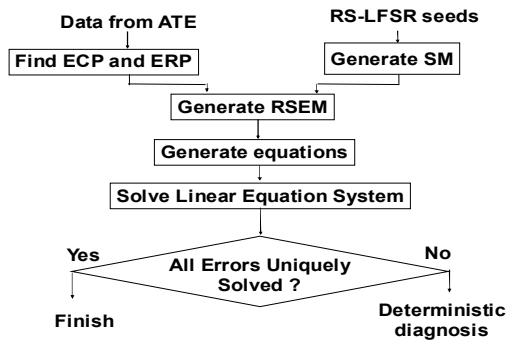


Fig. 5. Flow to Solve Errors (one diagnosis snapshot)

multiplicity is the total number of X's injected into the error matrices. The errors are assumed to be uniformly distributed.

Two experiments are performed on 10 x 100 error matrices. In the first experiment, all of unknowns (X's) are randomly distributed in a uniform way. The error multiplicity is fifteen. Table I shows the average number of correctly diagnosed bits, wrongly diagnosed bits, and ambiguous bits. The left three columns are from the original CPRS, in which equations with unknowns are skipped. More than 256 diagnosis sessions are needed to completely diagnose all 15 errors in the presence of 100 unknowns. The right three columns are from Cyclic-CPRS, in which the unknowns are directly diagnosed. Within sixteen random diagnosis sessions, Cyclic-CPRS correctly diagnoses every error even in the presence of 100 unknowns. Compared with the original CPRS, Cyclic-CPRS is much better in terms of diagnosis time and diagnosis resolution. Although 10 x 100 does not represent a big circuit, the effectiveness of Cycle-CPRS is not affected by the size of CUT. (Please see Table III for experiments on ten times larger CUTs.)

The amount of data required for Cyclic-CPRS is equal to the number of sessions times the sum of rows and columns; that is,  $s \times (m + l)$ . If we do a direct dumping, the amount of data would be  $m \times l$ . For a small CUT, such as this experiment, the amount of data for Cyclic-CPRS might seem larger than a direct dumping. However, as the length of scan chain chains gets large, the amount of data of Cyclic-CPRS may actually be smaller than that of a direct dumping.

The second experiment is similar to the first one except that the unknowns are now clustered in two scan chains. The numbers of the second experiment are shown in Table II. It is again observed that in the presence of 100 unknowns, every error is correctly diagnosed by Cyclic-CPRS within sixteen diagnosis sessions.

It is noted that Cyclic-CPRS has worse diagnosis results than the original CPRS in the first few diagnosis sessions. Because Cyclic-CPRS treats unknowns as error variables, the error variables of Cyclic-CPRS increase faster than that of the original CPRS. As the number of sessions increases, Cyclic-CPRS outperforms the original CPRS. Finally, the results of Cyclic-CPRS are significantly better than those of the original CPRS after twelve diagnosis sessions. Experimental results suggest that sixteen diagnosis sessions is enough for CPRS to diagnose 15 errors plus unknowns.

Table III compares the percentage of correctly diagnosed

TABLE I  
DIAGNOSIS RESULTS OF 10X100 (15 ERRORS, SCATTERED X)

# of Diagnosis Sessions	UM	Original CPRS			Cyclic-CPRS		
		Correct	Wrong	Amb.	Correct	Wrong	Amb.
2	10x	954.8	13.7	31.5	874.1	12.2	113.7
	50x	939.3	11.3	49.4	702.2	31.5	266.3
	100x	924.6	7.9	67.5	551.8	53.7	394.4
4	10x	941.1	16.3	42.7	817.8	4.7	177.5
	50x	947.0	14.6	38.4	569.2	10.9	419.9
	100x	902.6	5.6	91.9	375.8	19.2	605.0
8	10x	977.7	21.2	1.0	987.8	11.5	0.8
	50x	964.9	19.7	15.4	615.8	11.8	372.4
	100x	901.2	6.7	92.1	434.6	17.8	547.6
12	10x	985.9	13.4	0.8	998.6	1.3	0.1
	50x	973.8	22.5	3.7	999.8	0.2	0.0
	100x	929.9	13.2	56.8	999.4	0.6	0.0
16	10x	987.5	11.7	0.8	1000.0	0.0	0.0
	50x	974.5	23.6	1.9	1000.0	0.0	0.0
	100x	942.5	16.0	41.4	1000.0	0.0	0.0
32	10x	988.8	10.2	0.9	1000.0	0.0	0.0
	50x	973.2	25.1	1.7	1000.0	0.0	0.0
	100x	965.2	20.5	14.2	1000.0	0.0	0.0
64	10x	989.1	9.9	1.0	1000.0	0.0	0.0
	50x	972.0	25.6	2.4	1000.0	0.0	0.0
	100x	972.2	21.5	6.3	1000.0	0.0	0.0
128	10x	989.2	9.8	1.0	1000.0	0.0	0.0
	50x	972.0	24.9	3.1	1000.0	0.0	0.0
	100x	974.5	21.9	3.5	1000.0	0.0	0.0
256	10x	989.2	9.8	1.0	1000.0	0.0	0.0
	50x	972.6	23.6	3.8	1000.0	0.0	0.0
	100x	974.3	23.1	2.6	1000.0	0.0	0.0

TABLE II  
DIAGNOSIS RESULTS OF 10X100 (15 ERRORS, CLUSTERED X)

# of Diagnosis Sessions	UM	Original CPRS			Cyclic-CPRS		
		Correct	Wrong	Amb.	Correct	Wrong	Amb.
2	10x	945.1	10.8	44.1	885.8	12.8	101.4
	50x	952.1	12.2	35.8	725.4	32.5	242.1
	100x	960.1	13.8	26.1	581.9	57.7	360.4
4	10x	933.1	14.4	52.5	839.0	5.9	155.1
	50x	937.7	14.8	47.5	629.7	15.8	354.5
	100x	945.4	15.8	38.7	428.5	24.8	546.7
8	10x	977.7	21.7	0.6	991.2	8.2	0.6
	50x	973.7	26.2	0.1	847.8	15.4	136.9
	100x	967.2	32.3	0.5	675.9	28.1	296.0
12	10x	986.9	12.9	0.2	1000.0	0.0	0.0
	50x	980.2	19.8	0.0	999.9	0.1	0.0
	100x	973.2	26.6	0.1	990.4	9.0	0.6
16	10x	987.9	11.9	0.2	1000.0	0.0	0.0
	50x	983.4	16.6	0.0	1000.0	0.0	0.0
	100x	978.2	21.8	0.0	1000.0	0.0	0.0
32	10x	988.9	10.7	0.4	1000.0	0.0	0.0
	50x	987.1	12.9	0.0	1000.0	0.0	0.0
	100x	985.7	14.3	0.0	1000.0	0.0	0.0
64	10x	989.3	10.2	0.5	1000.0	0.0	0.0
	50x	987.6	12.4	0.0	1000.0	0.0	0.0
	100x	987.7	12.3	0.0	1000.0	0.0	0.0
128	10x	989.7	9.5	0.7	1000.0	0.0	0.0
	50x	987.6	12.4	0.0	1000.0	0.0	0.0
	100x	987.8	12.2	0.0	1000.0	0.0	0.0
256	10x	990.0	9.2	0.9	1000.0	0.0	0.0
	50x	987.6	12.4	0.0	1000.0	0.0	0.0
	100x	987.8	12.2	0.0	1000.0	0.0	0.0

TABLE III

COMPARISON OF CORRECTLY DIAGNOSED BITS

Technique	0.10% X	0.50% X	1.00% X	5.00% X	10.00% X
XC [29]	92.00%	15.70%	0.06%	NA	NA
CC3 [31]	97.10%	18.70%	0.08%	NA	NA
CC7 [31]	97.70%	52.40%	14.30%	NA	NA
Cyclic-CPRS	100.0%	100.0%	100.0%	100.0%	100.0%

TABLE IV

GATE COUNT COMPARISON

Methods	FF	XOR	NAND	Total Area (NAND)
[24]	$\log_2 b+m+k$	$k\log_2 b+m+k$	$m(\log_2 b+1)$	$32.3m+22.7^*$
CC3 [31]	32	3m	1	$182.3+8m$
CC7 [31]	32	7m	1	$182.3+18.7m$
[30]	m	2m	0	11m
Cyclic-CPRS	2m	3m	4m	23.3m

\*assume  $k=m$ ,  $b=16$ 

bits with existing techniques under various unknown multiplicities. The numbers of Cyclic-CPRS are obtained from our simulations and the other numbers are obtained from [31]. For a fair comparison, all techniques have one hundred times compression ratio (*i.e.* one hundred scan outputs compressed into one output). The Cyclic-CPRS simulations are performed on 10,000 scan cells, which are partitioned into 100 scan chains. In the presence of 10% scattered unknowns, Cyclic-CPRS correctly diagnoses all cases of single errors and multiple errors. The performance of Cyclic-CPRS is better than previous techniques.

Table IV compares the diagnosis circuitry area overhead of several techniques. The number of flip-flops, XOR, and AND gates are derived from the original papers. The total area is expressed in the number of equivalent NAND gates as a function of  $m$  (number of scan chains). The conversion of cell area is based on the numbers in the TSMC 0.18  $\mu$ m standard cell library. Cyclic-CPRS costs about 23 NAND gates per scan chain, which is approximately in the same order as the other techniques. (For the convenience of comparison, some typical numbers are assumed for [24].)

#### IV. Summary

The Cyclic-CPRS technique proposes cyclic scan chain structure to retain the transient errors and unknowns until they are fully diagnosed. The direct diagnosis approach faces the unknowns as if they were errors. Experimental results show that Cyclic-CPRS is 100% correct even when up to 10% scan cells are errors or unknowns. Cyclic-CPRS is especially suitable for transient errors and systematic defects in nano-meter technologies.

#### Acknowledgements

This research is supported by the National Science Council of Taiwan under contract number NSC93-2220-E-002-012.

#### References

- [1] J. A. Waicukauski and E. Lindbloom, "Failure Diagnosis of Structured VLSI," *IEEE Journal - Design & Test of Computers*, VOL. 6, NO. 4, August, pp.49-60, 1989.
- [2] Bernd Koenemann, "What You see is NOT What You Get," *IEEE Proc. - International Test Conference* keynote speech, pp. 12, 2004.
- [3] C. Metra, M. Favalli, and B. Ricco, "Self-checking detection and diagnosis of transient, delay, and crosstalk faults affecting bus lines," *IEEE Trans. on Computers*, VOL. 49, June, pp. 560-574, 2000.
- [4] Y. Zhao, S. Dey, and L. Chen, "Double Sampling Data Checking Technique: An Online Testing Solution for Multisource Noise-Induced Errors on On-Chip Interconnects and Buses," *IEEE Trans. on VLSI Systems*, VOL. 12, NO. 6, June, pp. 746-755, 2004.
- [5] W. Dally and J. Poulton, *Digital Systems Engineering*. Cambridge, U.K.: Cambridge Univ. Press, 1998.
- [6] M. Nicolaidis, "Time redundancy based soft-error tolerance to rescue nanometer technologies," *IEEE Proc. - VLSI Test Symposium*, pp. 89-94, 1999.
- [7] T.M. Mak, A. Krstic, K-T Cheng and L-C. Wang, "New Challenges in Delay Testing of Nanometer, Multigigahertz Designs," *IEEE Journal - Design & Test of Computers*, vol. 21, no.3, pp. 241-247, May/June 2004.
- [8] A. A. Al-Yamani, S. Mitra, and E. J. McCluskey, "Testing digital circuits with constraints," *IEEE Proc. - International Symposium on Defect and Fault Tolerance in VLSI Systems*, pp. 195-203, 2002
- [9] W.H. McAnney and J. Savir, "There is Information in Faulty Signature," *IEEE Proc. - International Test Conference*, pp. 630-636, 1987.
- [10] J. C. Chan and J. A. Abraham, "A study of faulty signatures using a matrix formulation," *IEEE Proc. - International Test Conference*, pp. 553-561, 1990.
- [11] R. C. Aitken and V. K. Agarwal, "A diagnosis method using pseudo-random vectors without intermediate signatures," *IEEE Proc. - International Conference on Computer-Aided Design*, pp 574-577, 1989.
- [12] J. Savir, "Salvaging Test Windows in BIST Diagnostics," *IEEE Proc. - VLSI Test Symposium*, pp. 416-425, 1997.
- [13] J. Rajski and J. Tyszer, "On the diagnostic properties of linear feedback shift registers," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, VOL. 10, NO. 10, October, pp. 1316-1322, 1991.
- [14] P. Wohl, J. A. Waicukauski, S. Patel, and G. Maston, "Effective diagnostics through interval unloads in a BIST environment," *IEEE Proc. - Design Automation Conference*, pp. 249-254, 2002.
- [15] P. Wohl, J. A. Waicukauski, S. Patel, and M. B. Amin, "X-tolerant compression and application of scan-ATPG patterns in a BIST architecture," *IEEE Proc. - International Test Conference*, pp. 727-736, 2003.
- [16] R. C. Tekumalla, "On Reducing Aliasing Effects and Improving Diagnosis of Logic BIST Failures," *IEEE Proc. - International Test Conference*, pp. 737-744, 2003.
- [17] M. G. Karpovsky and S. M. Chaudhry, "Design of self-diagnostic boards by multiple signature analysis" *IEEE Trans. on Computers*, VOL. 42, NO. 9, September, pp. 1035-1044, 1993.
- [18] T. R. Damarla, C. E. Stroud, and A. Sathaye, "Multiple error detection and identification via signature analysis", *Journal of Electronic Testing: Theory and Applications (JETTA)*, VOL. 7, NO. 3, December, pp. 193-207, 1995.
- [19] J. Savir and W.H. McAnney "Identification of failing tests with cycling registers," *IEEE Proc. - International Test Conference*, pp. 322-328, 1988.

- [20] J. Ghosh-Dastidar, D. Das, and N. A. Touba, "Fault diagnosis in scan-based BIST using both time and space information," *IEEE Proc. - International Test Conference*, pp. 95-102, 1999.
- [21] T. R. Damarla, W. Su, M. J. Chung, Charles E. Stroud, and Gerald T. Michael, "Built-in self test scheme for VLSI," *IEEE Proc. - Asia and South Pacific Design Automation Conference, ASP-DAC*, pp. 217-222, 1995.
- [22] Y. Wu and S. Adham, "BIST fault diagnosis in scan-based VLSI environments," *IEEE Proc. - International Test Conference*, p 48-57, 1996.
- [23] Y. Wu and S. M. I. Adham, "Scan-based BIST fault diagnosis," *IEEE Trans. on Computer-Aided Design Of Integrated Circuits and Systems*, VOL. 18, NO. 2, February, pp. 203-211, 1999.
- [24] J. Rajski and J. Tyszer, "Fault diagnosis in scan-based BIST," *IEEE Proc. - International Test Conference*, pp. 894-902, 1997.
- [25] J. Rajski and J. Tyszer, "Diagnosis of scan cells in BIST environment," *IEEE Trans. on Computers*, VOL. 48, NO. 7, July, pp. 724-731, 1999.
- [26] I. Bayraktaroglu and A. Orailoglu, "Deterministic partitioning techniques for fault diagnosis in scan-based BIST," *IEEE Proc. - International Test Conference*, pp. 273-282, 2000.
- [27] I. Bayraktaroglu and A. Orailoglu, "Cost-effective deterministic partitioning for rapid diagnosis in scan-based BIST," *IEEE Journal - Design & Test of Computers*, VOL. 19, NO. 1, January/February, pp. 42-53, 2002.
- [28] J. Ghosh-Dastidar, and N. A. Touba, "Rapid and scalable diagnosis scheme for BIST environments with a large number of scan chains," *IEEE Proc. - VLSI Test Symposium*, pp. 79-85, 2000.
- [29] S. Mitra and K. S. Kim, "X-Compact: an efficient Response Compaction Technique for Test Cost Reduction," *IEEE Proc. - International Test Conference*, pp. 311-320, 2002.
- [30] O. Sinanoglu and A. Orailoglu, "Compacting Test Responses for Deeply Embedded SoC Cores," *IEEE Design & Test of Computers*, VOL. 20, NO. 4, July/August, pp. 22-30, 2003.
- [31] J. Rajski, J. Tyszer, and S. M. Reddy, "Convolutional Compaction of Test Responses," *IEEE Proc. - International Test Conference*, pp. 745-754, 2003.
- [32] J. Rajski, J. Tyszer, S. M.Reddy, and C. Wang, "Finite memory test response compactors for embedded test applications," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, VOL. 24, NO. 4, April, pp. 622-634, 2005.
- [33] H. M. Lin and J. C. -M. Li, "Column Parity and Row Selection (CPRS): A BIST Diagnosis Technique for Multiple Errors in Multiple Scan Chains" *IEEE Proc. - International Test Conference*, paper #42.3, 2005.
- [34] Li, J. C.-M. and E. J. McCluskey, "Diagnosis for Sequence Dependent Chips," *IEEE Proc. - VLSI Test Symposium*, pp.187-192, 2002.