Test Data Compression Technique Using Selective Don't-Care Identification

Terumine Hayashi, Haruna Yoshioka, Tsuyoshi Shinogi, Hidehiko Kita, and Haruhiko Takase

Department of Electrical and Electronic Engineering
Mie University, Japan

Abstract - *In this paper, we propose an effective method for reducing test data volume under multiple scan chain designs. The proposed method is based on (1) reduction of distinct scan vectors (words) using selective don't-care identification, and (2) reduction of total test data volume using single/double length coding. In (1), don't-care identification is repeatedly applied under conditions that each bit in specified scan vectors is fixed to binary value (0 or 1). In (2), the code length for frequent scan vectors is shortened in the manner that the code length for rare scan vectors is designed as double of that for frequent ones. The proposed method achieves not only high compression efficiency, but also has a feature that the decompressor circuits are rather simple like combinational ones. The effectiveness of the proposed method is shown through experiments for ISCAS'89 and ITC'99 benchmark circuits*

I. Introduction

As LSI device and design technologies progress, the importance of testing and reliability has significantly increased. Especially, the problems on test cost have become one of the most important issues. Test cost reduction includes test data volume reduction, test application time reduction, reduction of power consumptions, and so on. It is well known that test compaction techniques [1,2] are fundamental to reduce test cost for circuits with scan structure. Besides, several approaches (for example, [3-10]) have been already proposed for test cost reduction. Test data for a circuit-under-test (CUT) are compressed by these techniques, and they are decompressed through decompressor circuits. In this paper, we concentrate the reduction of test data volume for circuits with multiple scan structure.

It is known that several statistical compression methods such as Huffman coding, run length coding, dictionary based method and so on can be applied to test data compression for multiple scan designs. Also, it has been shown that don't-care identification technique [11] can be effectively used for making compression ability higher [9]. Decompressor circuits are important for these methods. As the decompressor in [9] is a combinational circuit, that is, a decoder, the decompressing is relatively simple. In this method, the decompressor transforms one external scan vector into one internal scan vector. Thus, the smaller the number of distinct internal scan vectors is, the smaller the bit width of external scan vectors can be. Therefore, the number of distinct internal scan vectors should be reduced. This fact also holds for dictionary-based methods such as [10].

Therefore, we first propose a method for reducing the number of distinct internal scan vectors using selective don't-care identification techniques. In this paper, distinct internal scan vectors are referred to as simply "words" like those in dictionary-based compression In our method, don't-care identification is applied under conditions that each bit in frequent words is fixed to binary value (0 or 1), and cannot be replaced into don't-care value X. It is similar to don't-care identification for specific bits in [13]. This process is repeated through gradually increasing the number of binary-fixed words. Moreover, the code length for frequent words is shortened in the manner that the code length for rare words is designed as double of that for frequent ones. It is a feature that this coding method does not need any additional prefix bit as used in dictionary-based compression methods. The proposed techniques are useful for making not only compression rate higher, but also the decompressor for a CUT is simple like combinational ones in [9].

The rest of the paper is organized as follows. In Section II, we briefly give some preliminaries. Section III describes the proposed method for word reduction using selective don't-care identification. In Section IV, we describe a coding method that composes of single and double length codes. Then, experimental results are presented and compared with previous work in Section V. Finally, we conclude the paper in Section VI.

II. Preliminaries

This section describes some terms and notations about test data compression for circuits with multiple scan structure. Figure 1 shows the general multiple scan structure and its decompressor. Now, let *NSC* and *MSCL* be the number of scan chains and the length of each scan chain in a CUT. Then, the number of flip-flops (FFs) is nearly equal to *NSC*MSCL*, where we assume that all primary inputs in the CUT are treated as scan FFs. Each bit in an internal scan vector is received by the corresponding each scan chain. Let *NT* be the number of test patterns in a test set for the CUT. Then, the number of internal scan vectors is *NT*MSCL*. Now suppose that the decompressor in Figure 1 has *NSI* inputs and *NSC* outputs. An input vector and an output vector for the decompressor is referred to as an external scan vector and an internal scan vector, respectively. If the decompressor is a combinational circuit as shown in [9], an external scan vector is converted into an internal one. In Section IV, we enhance the decompressor as an internal scan vector is made from one or two external scan vectors (called single or double length indices). Let *NDO* be the number of distinct internal scan vectors in a test set, which are referred to as "words," where every internal scan vector in the test set must be represented by any of words. It is desirable that the number *NDO* should be reduced for making the compression efficiency higher. In the next section, we introduce the selective don't-care identification technique for this purpose.

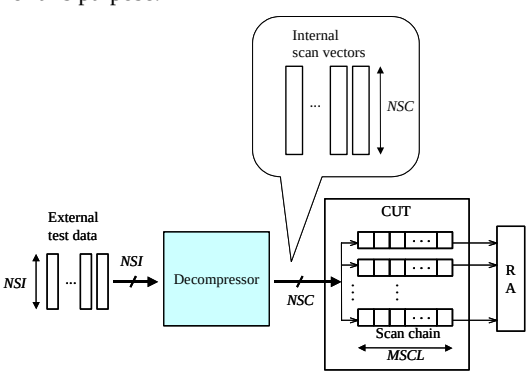


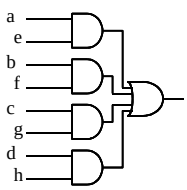
Figure 1: Multiple scan structure and decompressor

III. Word reduction using selective don't-care identification

This section presents our method for reducing the number of words. Now suppose that a test set *T* with X values for a CUT is already obtained by methods as shown in [11,12]. We can easily obtain a set of words with as small *NDO* as possible by greedy

methods such as used in [9,10]. As a result, we can obtain a word set $\{w_1, w_2, \cdots, w_{NDO}\}$ of *NDO* words and the occurrence frequency n_i $(1 \le i \le NDO)$ of each word in the test set.

[Example 1]
Suppose that the circuit in Figure 2(i) has four scan chains, and a test set with Xs that detects all single stuck-at faults is obtained as shown in Figure 2(ii). In this case, we know *NT* = 6, *NSC* = 4, *MSCL* = 2, and the total internal scan vectors *NT***MSCL* = 12. Also, the word set is computed as {1001, 0110, 0100, 0100, 1xx0, 0xx1}, and the occurrence frequencies $n_1, n_2, \cdots, n_6$ are 4, 4, 1, 1, 1, 1, respectively. In this case, the value of *NDO* is 6.



	abcd efgh
t_1	0110 1001
t_2	1001 0110
t_3	1xx0 100x
t_4	0100 x1xx
t_5	0010 xx1x
t_6	x001 0xx1

words	#freq.
1 0 0 1	4
0 1 1 0	4
0 1 0 0	1
0 0 1 0	1
1 x x 0	1
0 x x 1	1

(*NDO* = 6)

(i) Circuit (ii) Test set with x (iii) Words

Figure 2: Test set and words

The proposed method tries to reduce the number of words using selective don't-care identification. Selective don't-care identification means that don't-care identification is applied under conditions that some specified bits in the test set must not be replaced into X. It can be easily implemented by extending a simulation-based don't-care identification procedure as shown in [12]. In the proposed method, each bit in all internal scan vectors with frequent words is fixed to binary value 0 or 1, and the replacement into X is inhibited for all fixed bits. This process is repeated by the manner that the number of fixed words is small at the beginning, but it is gradually increased. Through the repetition of the process, bit positions with X value are modified so that they are useful to make the value of *NDO* smaller.

	abcd efgh
t_1	0110 1001
t_2	1001 0110
t_3	1xx0 1001
t_4	0100 0110
t_5	0010 0110
t_6	1001 0xx1

	abcdefgh
t_1	0110 1001
t_2	1001 0110
t_3	1xx0 1001
t_4	x10x 0110
t_5	x01x 0110
t_6	1001 0xx1

words	#freq.
1 0 0 1	4
0 1 1 0	4
1 1 0 0	2
0 0 1 1	2

(*NDO* = 4)

(i) Fixing to binary values (ii) Selective don't-care identification (iii) Words

Figure 3: An example of word reduction

[Example 2]
Now suppose that each bit in all internal scan vectors with the frequent words 1001 and 0110 is fixed to binary value 0 or 1 in the test set as shown in Figure 2(ii). By the fixing, Figure 3(i) is obtained. Under the condition, the selective don't-care identification process is carried out. As a result, each bit of 4 positions in the test set are newly replaced into X as shown in Figure 3(ii), and also the number of words, NDO, becomes 4 as shown in Figure 3(iii).

The proposed method for word reduction using the selective identification technique is described as follows.

[Word reduction procedure using selective don't-care identification]
Input: A test set *T* that detects all detectable faults in the given fault set.
Output: A test set *T'* with as small *NDO* as possible that detects all detectable faults in the given fault set, and its word set $\{w_1, w_2, \cdots, w_{NDO}\}$ and the frequencies $n_1, n_2, \cdots, n_{NDO}$ of occurrence for each word.
(Step 1) Set k = 0 and Δ=1, and copy *T* to *T'*. Where, k represents the number of words to be binary-fixed, and Δ is the interval for increasing the value of k.
(Step 2) Compute a word set in which the number of elements is as small as possible using a greedy method. At the same time, compute the value of *NDO*, and the frequencies of occurrence for each word.
(Step 3) Put $k \leftarrow k + \Delta$. If necessary, Δ is modified into a larger value. If $k \ge NDO$, then the procedure terminates.
(Step 4) Each bit of all internal scan vectors with any of most frequent k words in *T'* is fixed to binary value 0 or 1. If these words include X values, then their Xs are assigned to 0 or 1 randomly.
(Step 5) Let *T"* be the test set to be obtained from *T'* through the assignment in Step 4. Carry out don't-care identification for all bits except those in internal scan vectors with most frequent k words in *T"*. After that, set *T'* ← *T"*. Return to Step 2.

IV. Coding with single/double length indices

If the decompressor for producing internal scan vectors from external ones is a combinational circuit as shown in [9], the bit width NSI_0 of each external scan vector is determined by $NSI_0 = \lceil \log_2 NDO \rceil$. (For convenience, we use the notation of the suffix 0 like NSI_0 in the case of the combinational decompressor.) It means that the value NSI_0 cannot always be reduced even though the value *NDO* is reduced. For example, even if *NDO* =200 have been reduced to *NDO* =130, NSI_0 still remain 8. We introduce a coding method with indices of single/double length for enhancing the compression efficiency. The basic concept is that the code length for frequent scan vectors is shortened in the manner that the code length for rare scan vectors is designed as double of that for frequent ones. The proposed compressor is shown in Figure 4. It composes of a single/double control signal generator, a decoder, and a translation table (dictionary table). The control signal generator generates signals for deciding whether an internal scan vector is produced from one or two external scan vectors (single or double length indices). Though the proposed decompressor is a kind of sequential circuit, it achieves not only high compression efficiency, but also has a feature that the decompressor is rather simple like combinational ones.

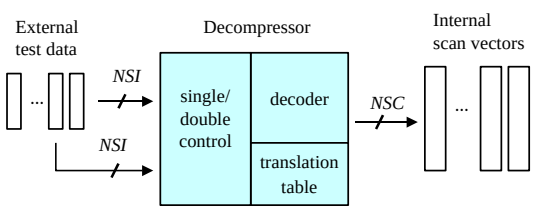


Figure 4: Proposed decompressor

[Example 3]
In Figure 5, *NDO* = 7 and $NSI_0 = \lceil \log_2 7 \rceil = 3$ are determined for the combinational decompressor. On the other hand, we obtain *NSI* = 2 for the proposed decompressor by the assignment that each of more frequent three vectors has single length and each of rest four vectors has double length. Since *NSC* equals 6, the compression ratio is 3/6=0.5 for the combinational decompressor. On the other hand, it becomes (2/6)*(32/40)+(4/6)*(8/40)= 0.4 for the proposed method. Thus, the proposed method can improve the

compression ratio by 10% for this example.

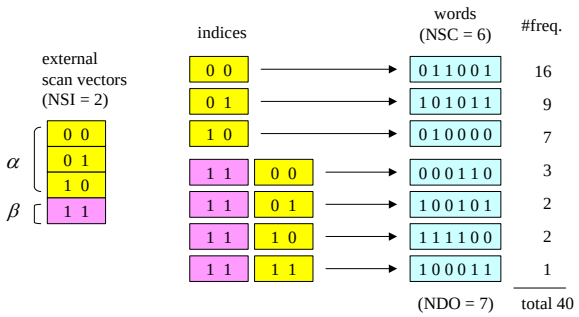


Figure 5: Single/double length coding

In this coding method, the compression efficiency depends on the values *NSI*, α and β. Therefore, these values need to be determined well. In the rest of this section, we make a generalized discussion for this problem. The number of external scan vectors is *NT*MSCL* as described in Section II. By applying the greedy method presented in Section III to these vectors, we obtain *NDO* words, $w_1, w_2, \cdots, w_{NDO}$ and their frequencies in external scan vectors, $n_1, n_2, \cdots, n_{NDO}$, where $n_i \geq n_j$ if $i < j$. Suppose that the bit width of external scan vectors is *NSI*. Let α and β be the numbers of indices with single and double length, respectively. Then, $\alpha + \beta = 2^{NSI}$ holds, where $1 \leq \alpha \leq 2^{NSI}$. If the values of *NSI* and α are fixed, then the maximum number of possible indices, $NDO_{\max}(NSI, \alpha)$ is given by the following equation.

$$NDO_{\max}(NSI, \alpha) = \alpha + \beta \cdot 2^{NSI}$$
$$= \alpha + (2^{NSI} - \alpha)2^{NSI}$$

If *NSI* is fixed and α is variable, then $NDO_{\max}(NSI, \alpha)$ is maximum in the case of α=1. $NDO_{\max}(NSI, 1)$ is shown in the following equation.

$$NDO_{\max}(NSI, 1) = 1 + (2^{NSI} - 1)2^{NSI}$$

This equation shows that the proposed method can represent many indices by a small number of *NSI*. However, it does not necessarily mean that the smaller *NSI* is, the higher the compression efficiency is. It depends on the distribution of occurrence frequencies of words in internal scan vectors. Therefore, the proposed method computes the value α that brings the highest compression efficiency for some candidates of *NSI*. After that, the pair of *NSI* and α with the highest compression efficiency is chosen in them.

The compression ratio R_0 in the case of the combinational decompressor is given by $R_0 = (NSI_0 / NSC)$. While, The compression ratio *R* in the proposed method is represented by the following.

$$R = \left(\frac{NSI}{NSC}\right) \cdot \left(1 + \sum_{i=\alpha+1}^{2^{NSI}} \frac{n_i}{NT \cdot MSCL}\right)$$

R_0 and R do not include any overhead by dictionary data, which are required to translate into internal scan vectors from external ones. If the volume of the dictionary data is small, then R_0 and *R* can be reasonable measures. However, if not so, the overhead should be included for making comparisons fair. We introduce the following measures CR_0 and CR that include dictionary data.

$$CR_0 = R_0 + \frac{NDO}{NT \cdot MSCL} \qquad CR = R + \frac{NDO}{NT \cdot MSCL}$$

V. Experimental results

We implemented the proposed method using the C language on a Free BSD PC with Pentium IV processor, and first carried out experiments for ISCAS'89 benchmark circuits. We assumed that each circuit has a multiple full-scan structure, and all primary inputs are treated as scan flip-flops. In our experiments, the numbers of scan chain *NSC* was set to 16, 24, 32, 48 or 64. For each *NSC*, the order of scan elements in scan chains was determined by the same way as shown in [10].

Table 1 shows the experimental results of test data compression by the proposed method, together with those by the method using the combinational decompressor based on [9]. In Table 1, the column T_D represents bits of test data volume to be compressed for each circuit. We used "x-maximal" test sets generated by the method in [12] for our experiments.

Also, *NT*, *MSCL*, *NDO*, *NSI*, α, β, *R*, *CR*, R_0 and CR_0 are the same as presented in Section III and IV. V_β is the number of words assigned to double length indices, and N_β is the number of internal scan vectors to be decompressed by double length indices. Besides, the columns T_E and $T_E + C$ represent compressed test data volume excluding dictionary data, and that including dictionary data, respectively. They are expressed as the following equations.

$$T_E = NSI \cdot (NT \cdot MSCL + N_\beta)$$
$$T_E + C = NSI \cdot (NT \cdot MSCL + N_\beta) + NDO \cdot NSC$$

Also, NDO_0 is the number of words before applying the selective don't-care identification procedure.

The selective don't-care identification in the proposed method has reduced the number of words by about 48% on average. It shows that selective don't-care identification technique is very effective for reducing words in a test set. As for test data volume, the sizes of compressed test data depend on the number of scan-chains. The proposed method has reduced test data volume by about 65% on average compared with those by the combinational decompressor. The best value of $T_E + C$ for each circuit is shown with the corresponding value T_E by bold type characters in Table 1

Table 2 shows the comparisons of compressed test data volume with some of related work [5,6,10]. The proposed method has brought better results than previous methods for s5378, s9234, s15850, s38417 and s38584, even though dictionary data are included.

Besides, we carried out our experiments for ITC'99 benchmark circuits [14]. In this experiments, *NSC* was set to 16, 24, 32, 64 or 96. Table 3 represents the best result for each circuit. It shows that the proposed method is very effective in test data volume reduction.

VI. Conclusions

We have proposed an effective method for reducing test data volume under multiple scan chain designs. The proposed method is based on (1) reducing the number of words using selective don't-care identification, and (2) reducing total data volume using single/double length coding technique. The proposed method achieves not only high compression efficiency, but also has a feature that the decompressor for a CUT is simple as well as com-

binational ones. The effectiveness of the proposed method has been shown through experiments for ISCAS'89 and ITC'99 benchmark circuits

References

[1] S. Kajihara, I. Pomeranz, K. Kinoshita and S.M. Reddy, "Cost-Effective Generation of Minimal Test Sets for Stuck-at Faults in Combinational Logic Circuits," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, Vol.14, No.12, pp.1496-1504, 1995.

[2] I. Hamzaoglu and J. H.Patel, "Test Set Compaction Algorithms for Combinational Circuits," *Proc. Int. Conf. On Computer-Aided Design*, pp.283-289, 1998.

[3] A. Jas, J. Ghosh-Dastidar and N. A. Touba, "Scan Vector Compression/Decompression Using Statistical Coding," *Proc. 17th VLSI Test Symposium*, pp.114-120, 1999.

[4] I. Hamzaoglu and J. H. Patel, "Reducing Test Application Time for Full Scan Embedded Cores," *Proc. 29th Int. Symp. on Fault-Tolerant Computing*, pp.260-1504, 1999.

[5] I. Bayraktaroglu and A. Orailoglu, "Test Volume and Application Time Reduction through Scan Chain Concealment," *Proc. 38th Design Automation Conference*, pp.151-155, 2001.

[6] A. Chandra and K. Chakrabarty, "Frequency- Directed Run-Length (FDR) Codes with Application to System-on-a-Chip Test Data Compression," *Proc. 19th VLSI Test Symposium*, pp.42-47, 2001.

[7] H. Ichihara, A. Ogawa, T. Inoue and A. Tamura, "Dynamic Test Compression Using Statistical Coding," *Proc. 10th Asian Test Symposium*, pp.143-148, 2001.

[8] S. Kajihara, K. Taniguchi, K. Miyase, I. Pomeranz and S. M. Reddy, "Test Data Compression Using Don't-Care Identification and Statistical Encoding," *Proc. 11th Asian. Test Symposium*, pp.67-72, 2002.

[9] S. M. Reddy, K. Miyase, S. Kajihara and I. Pomeranz, "On Test Data Volume Reduction for Multiple Scan Chain Designs," *Proc. 20th VLSI Test Symposium*, pp.103-108, 2002.

[10] L. Li and K. Chakrabarty, "Test Data Compression Using Dictionaries with Fixed-Length Indices," *Proc. 21th IEEE VLSI Test Symposium*, pp.219-224, 2003.

[11] S. Kajihara and K. Miyase, "On Identifying Don't Care Inputs of Test Patterns for Combinational Circuits," *Proc. Int. Conf. on Computer-Aided Design*, pp.364-369, 2001.

[12] T. Hayashi, Y. Morimoto, T. Shinogi, H. Kita and H. Takase, "X-Maximal Test Set Generation for Combinational Circuits," *Proc. 3rd Workshop on RTL and High Level Testing*, pp.88-93, 2002.

[13] K. Miyase, S. Kajihara, I. Pomeranz and M. Reddy, "Don't-Care Identification on Specific Bits of Test Patterns," *Proc. Int. Conf. On Computer Design*, pp.194-199, 2002.

[14] S. Davidson and Panelists, "ITC'99 Benchmark Circuits – Preliminary Results," *Proc. Int. Test Conf*,. p.1125, 1999. (*http://www.cad.polito.it/tools/itc99.html*)

Table 1: Experimental results of test data compression for ISCAS'89 circuits

Circuit	NT	T_D	NSC	$MSCL$	$MSCL$ *NT	Proposed method										Combinational decompressor			
						NDO	NSI	α	β	V_β	N_β	R	CR	T_E	T_E+C	NDO_0	NSI_0	R_0	CR_0
s5378	99	21186	16	14	1386	37	3	3	5	34	589	0.27	0.29	5925	6517	91	7	0.44	0.50
	99	21186	24	9	891	53	4	13	3	40	228	0.21	0.27	**4476**	**5748**	116	7	0.29	0.42
	99	21186	32	7	693	63	4	12	4	51	282	0.18	0.27	3900	5916	121	7	0.22	0.39
	99	21186	48	5	495	84	6	63	1	21	33	0.13	0.30	3168	7200	139	8	0.17	0.45
	99	21186	64	4	396	111	7	111	0	0	0	0.11	0.39	2772	9876	153	8	0.13	0.51
s9234	110	27170	16	16	1760	45	4	14	2	31	243	0.28	0.31	8012	8732	99	7	0.44	0.49
	110	27170	24	11	1210	78	4	11	5	67	464	0.23	0.30	6696	8568	128	7	0.29	0.40
	110	27170	32	8	880	103	4	10	6	93	514	0.20	0.32	**5576**	**8872**	157	8	0.25	0.43
	110	27170	48	6	660	108	4	9	7	99	460	0.14	0.31	4480	9664	174	8	0.17	0.43
	110	27170	64	4	440	141	7	127	1	14	14	0.11	0.43	3178	12202	189	8	0.13	0.55
s13207	233	163100	16	44	10252	22	3	6	2	16	273	0.19	0.19	31575	31927	79	7	0.44	0.45
	233	163100	24	30	6990	24	3	5	3	19	361	0.13	0.13	22053	22629	82	7	0.29	0.30
	233	163100	32	22	5126	29	3	5	3	24	522	0.10	0.11	16944	17872	89	7	0.22	0.24
	233	163100	48	15	3495	40	3	3	5	37	891	0.08	0.09	13158	15078	97	7	0.15	0.17
	233	163100	64	11	2563	43	3	3	5	40	891	0.06	0.08	**10362**	**13114**	99	7	0.11	0.15
s15850	97	59267	16	39	3783	33	3	4	4	29	858	0.23	0.24	13923	14451	108	7	0.44	0.47
	97	59267	24	26	2522	43	3	3	5	40	1146	0.18	0.20	11004	12036	125	7	0.29	0.34
	97	59267	32	20	1940	60	4	13	3	47	475	0.16	0.19	9660	11580	131	8	0.25	0.32
	97	59267	48	13	1261	83	4	11	5	72	586	0.12	0.19	**7388**	**11372**	157	8	0.17	0.29
	97	59267	64	10	970	98	4	10	6	88	573	0.10	0.20	6172	12444	164	8	0.13	0.29
s35932	12	21156	16	111	1332	199	4	3	13	196	908	0.42	0.57	8960	12144	251	8	0.50	0.69
	12	21156	24	74	888	196	4	4	12	192	626	0.28	0.50	6056	10760	238	8	0.33	0.60
	12	21156	32	56	672	98	4	10	6	88	357	0.19	0.34	**4116**	**7252**	138	8	0.25	0.46
	12	21156	48	37	444	193	4	4	12	189	330	0.15	0.58	3096	12360	213	8	0.17	0.65
	12	21156	64	28	336	112	4	9	7	103	206	0.10	0.43	2168	9336	127	7	0.11	0.49
s38417	86	143104	16	104	8944	57	4	13	3	44	867	0.27	0.28	39244	40156	206	8	0.50	0.52
	86	143104	24	70	6020	110	4	9	7	101	2064	0.22	0.24	32336	34976	310	9	0.38	0.43
	86	143104	32	52	4472	117	4	9	7	108	2193	0.19	0.21	**26660**	**30404**	306	9	0.28	0.35
	86	143104	48	35	3010	204	4	3	13	201	2438	0.15	0.22	21792	31584	395	9	0.19	0.32
	86	143104	64	26	2236	239	4	1	15	238	2143	0.12	0.23	17516	32812	372	9	0.14	0.31
s38584	111	162504	16	92	10212	83	4	11	5	72	1101	0.28	0.29	45252	46580	417	9	0.56	0.60
	111	162504	24	61	6771	112	4	9	7	103	1684	0.21	0.22	33820	36508	460	9	0.38	0.44
	111	162504	32	46	5106	126	4	8	8	118	1698	0.17	0.19	27216	31248	468	9	0.28	0.37
	111	162504	48	31	3441	164	4	6	10	158	1718	0.12	0.17	20636	28508	448	9	0.19	0.32
	111	162504	64	23	2553	181	4	5	11	176	1586	0.10	0.17	**16556**	**28140**	419	9	0.14	0.30
average						99.7						0.18	0.27			207.6		0.27	0.41

Table 2: Comparison with other methods

circuit	T_D	Mintest [2]	by Bay. [5]	by Chan. [6]	by Li [10]	Proposed method	
						T_E	(T_E+C)
s5378	21,186	20,758	—	12,346	6,345	4,476	(5,748)
s9234	27,170	25,935	—	22,152	11,498	5,576	(8,872)
s13207	163,100	163,100	25,334	30,880	8,517	10,362	(13,114)
s15850	59,267	57,424	22,784	26,000	13,873	7,388	(11,372)
s35932	21,156	19,393	7,128	22,744	1,400	4,116	(7,252)
s38417	143,104	113,152	89,856	93,466	62,939	26,660	(30,404)
s38584	162,504	161,040	38,976	77,812	53,287	16,556	(28,140)

Table 3: Experimental results for ITC'99 circuits

circuit	NT	T_D	NSC	$MSCL$ *NT	NDO_0	NDO	T_E	(T_E+C)
b14s	283	78,391	32	2,547	182	149	17,012	(22,196)
b15s	249	120,765	64	1,992	68	38	8,829	(11,261)
b17s	256	371,712	96	4,096	163	91	23,628	(32,364)
b20s	282	147,204	24	6,204	264	196	42,380	(47,084)
b21s	278	145,116	32	4,726	331	243	34,435	(42,211)
b22s	306	234,702	32	7,344	360	247	53,935	(61,839)