Random Limited-Scan to Improve Random Pattern Testing of Scan Circuits+

Irith Pomeranz
School of Electrical & Computer Eng.
Purdue University
W. Lafayette, IN 47907
pomeranz@purdue.edu

Abstract

We propose a method of random pattern generation for at-speed testing of circuits with scan. The proposed method uses limited scan operations to achieve complete fault coverage. Under a limited scan operation, the circuit state is shifted by a number of positions which may be smaller than the number of state variables. Limited scan operations are inserted randomly to ensure that the complete test set can be generated by a random pattern generator with simple control logic.

1. Introduction

A sequential circuit with full scan can be considered as a combinational circuit for the purpose of testing for stuck-at faults. Built-in test generation methods for full scan circuits that take this approach were described in [1]-[4]. Under these approaches, a single primary input vector is applied to the circuit between scan operations. Random-pattern generators are used as a low-cost option for built-in testing of scan circuits. When random patterns are not sufficient to achieve complete fault coverage, methods to improve the fault coverage include the use of weighted random patterns, the insertion of test points, and the use of multiple seeds to initialize the random-pattern generator.

Built-in test generation methods for circuits with full scan that allow at-speed testing of the circuit were described in [5] and [6]. The tests generated in [5] and [6] have the following form. A test τ_i starts with a scan-in operation that initializes the circuit state to a known state SI_i. Next, a sequence T_i of one *or more* primary input vectors is applied at-speed. During the application of T_i, the circuit state is determined through its functional path, and the scan chain is not used. The test ends with a scan out operation. We represent a test τ_i as $\tau_i = (SI_i, T_i)$. The length of a test τ_i is the number of primary input vectors in T_i.

The built-in test generation methods of [5] and [6] assume that an *LFSR* will be used to generate random tests. In [5], the goal was to maximize the fault coverage achieved when a given number of clock cycles is allocated for test application (i.e., a given number of clock cycles is to be used for scan operations and application of primary input vectors). It was observed in [5] that increasing the lengths of the primary input sequences applied at-speed can be used to increase the probability of detection of some faults. The test lengths to be used were selected based on this observation. In [6], an improved procedure for selecting the sequence lengths was described, and the number of different primary input sequence lengths was limited to two in order to simplify the test application process. For the same number of clock cycles allocated for test application, the method of [6] resulted in higher fault coverages than [5], and in longer primary input sequences applied at-speed. However, because of the use of random patterns, incomplete fault coverages are reported in both [5] and [6].

In the tests generated in [5] and [6], scan is used at the beginning and at the end of a test. The scan operations are *complete* in the sense that the values of all the flip-flops are written during scan-in and read during scan-out. This requires N_{SV} shifts of the scan chain, where N_{SV} is the number of state variables (and also the number of scanned flip-flops in the case of full scan). In this work, we improve the fault coverage achieved by random test sets similar to the ones of [5] and [6] by inserting *limited* scan operations into the tests. Under a limited scan operation, the number of shifts of the scan chain is smaller than N_{SV}. A limited scan operation that performs N_{SH} shifts of the scan chain affects the values of all the flip-flops; however, it requires only N_{SH} clock cycles, only N_{SH} new values are scanned in, and the values of only N_{SH} flip-flops are scanned out. Thus, limited scan is useful in partially reading the current circuit state, and in bringing the circuit into a new state. This helps detect new faults that cannot be detected by the original test, as illustrated in Section 2. The importance of limited scan is that its cost, in terms of test application time, is lower than the cost of inserting the same number of complete scan operations.

Limited scan was used in [7]-[11] to reduce the test application time of a deterministic test set for a full scan circuit when the test set contains primary input sequences of length one, i.e., when the circuit is considered as a combinational circuit. In this work, we use limited scan to improve the *fault coverage* achieved by a *random* test set that contains primary input sequences of length one *or more*, and thus allows at-speed testing of the circuit.

The insertion of limited scan operations in this work is done randomly, i.e., the time units where limited scan operations will be inserted and the number of shifts of the scan chain under a limited scan operation are both determined randomly. This is consistent with our goal of obtaining a random pattern generator similar to [5] and [6], but with improved fault coverage compared to [5] and [6]. The random insertion of limited scan operations is controlled by two parameters, denoted by I and D_1. Only values of these parameters need to be stored in order to guide the random insertion of limited scan operations. This is in addition

+ Research supported in part by NSF Grant No. MIP-9725053, and in part by SRC Grant No. 98-TJ-645.

DAC 2001, June 18-22, 2001, Las Vegas, Nevada, USA.

to storage of the test lengths and the number of tests of each length that are also required in [5] and [6].

The paper is organized as follows. In Section 2 we demonstrate the effects of inserting limited scan operations into a test. In Section 3 we describe how limited scan operations are randomly inserted into a random test set TS_0. Experimental results demonstrating the improvement in fault coverage that can be achieved by limited scan operations are given in Section 4. Section 5 concludes the paper.

2. Limited scan

We consider ISCAS-89 benchmark circuit $s27$ under the test $\tau = (SI, T)$ with $SI = 001$ and $T =$ (0111, 1001, 0111, 1001, 0100). In Table 1(a), we show the states and output vectors obtained by simulating this test in the fault free circuit and in the faulty circuit in the presence of a fault f which is not detected by the test. In Table 1, the time unit is shown under column u, the vector of T at time unit u is shown under column $T(u)$, the fault free and faulty states at time unit u are shown under column $S(u)$, and the fault free and faulty output vectors at time unit u are shown under column $Z(u)$. Fault free and faulty entries are separated by a slash. Note that $S(0) = SI/SI$.

Table 1: A test for $s27$

	(a) Without limited scan			(b) With limited scan			
u	$T(u)$	$S(u)$	$Z(u)$	shift(u)	$T(u)$	$S(u)$	$Z(u)$
0	0111	001/001	1/1	0	0111	001/001	1/1
1	1001	000/000	0/0	0	1001	000/000	0/0
2	0111	010/010	0/0	0	0111	010/010	0/0
3	1001	010/010	0/0	1	1001	001/001	1/0
4	0100	010/010	0/0	0	0100	101/010	1/1
5		011/011				001/001	

Table 2: Timing information for the test of Table 1(b)

u	$T(u)$	$S(u)$	$Z(u)$
0	0111	001/001	1/1
1	1001	000/000	0/0
2	0111	010/010	0/0
3	-	010/010	-
4	1001	001/001	1/0
5	0100	101/010	1/1
6		001/001	

Next, we simulate the same test, this time using a limited scan operation that shifts the circuit state at time unit three by one position. In our implementation, we always shift states to the right. During the shift, we assign to the leftmost bits random values. Shifting the state 010/010 obtained for $s27$ at time unit 3 and assigning the value 0 to the leftmost bit, we obtain the state 001/001. This causes the states and output values obtained following time unit 3 to change as shown in Table 1(b). Column $shift(u)$ in Table 1(b) shows the number of positions by which the state at time unit u is shifted. Although we still show the input vectors at their original time units in Table 1(b), in practice, there is another time unit between time units 2 and 3 where the state is shifted by one bit. A more accurate description of the test of Table 1(b) is shown in Table 2. Considering Table 1(b) (or Table 2), the fault is now detected on the primary output at time unit three (or four taking into account the time unit required for the limited scan operation).

For ease of presentation, we keep the time units of the original input sequence as we did in Table 1(b). However, it is important to note that if $shift(u) > 0$ for time unit u, then the test vector of time unit u is delayed by $shift(u)$ time units, and applied at time unit $u+shift(u)$.

The example of Table 1 shows that an undetected fault may be detected due to the change in the circuit states and outputs caused by a limited scan operation. In addition, a fault may be detected during the scan out operation that occurs during a limited scan operation. For example, suppose that a circuit reaches the state 00000/00010 at time unit u, and suppose that the state at time unit u is shifted by two positions. In the fault free circuit, the two bits shifted out are 00, while the same bits are 10 in the faulty circuit. Thus, the fault is detected during the scan out operation. In the following section, we take advantage of these additional detections to improve the fault coverage of a random test set.

3. Random limited scan insertion

In this section, we describe the insertion of limited scan operations into a random test set. The limited scan operations are randomly determined to ensure that the test generation process can be performed by *LFSR*s with minimal additional control logic. Although we use high-level procedures to describe the insertion of limited scan operations, these procedures can be easily implemented using *LFSR*s and additional logic.

We define an initial random test set TS_0 that does not include any limited scan operations. We use two test lengths as in [6] to define TS_0. The lengths are denoted by L_A and L_B. We use N tests of each length. Thus, our initial test set is $TS_0 = \{\tau_1, \tau_2, \cdots, \tau_N, \tau_{N+1}, \cdots, \tau_{2N}\}$, where τ_i for $1 \le i \le N$ is of length L_A, and τ_i for $N+1 \le i \le 2N$ is of length L_B (the length of a test τ_i is the number of vectors in its primary input sequence T_i). The values of L_A, L_B and N are determined as explained later. For a test $\tau_i = (SI_i, T_i) \in TS_0$, we select SI_i and T_i randomly. We perform the random value selection such that it can be repeated, i.e., such that it would be possible to apply the same test set TS_0 multiple times. This can be achieved by using a dedicated random pattern generator (such as an *LFSR*) for TS_0, and always using the same seed to initialize it before TS_0 is generated.

Based on TS_0, we define test sets denoted by $TS(I, D_1)$ whose tests contain limited scan operations. We define $TS(I, D_1)$ such that every test $\hat{\tau}_i \in TS(I, D_1)$ is identical to a different test $\tau_i \in TS_0$, except that $\hat{\tau}_i$ contains limited scan operations. Two parameters, I and D_1, determine the time units where limited scan operations will be performed as part of the different tests, the numbers of bits by which the states will be shifted, and the new bits that will be scanned in. The parameter I designates an iteration of the limited-scan insertion procedure, and is aimed at ensuring that different test sets will be obtained at different iterations. The parameter D_1 determines the probability of inserting a limited scan operation at any given time unit of a test. A higher value of D_1 implies a larger number of time units between every two limited scan operations, or a lower probability of inserting a limited scan operation at a given time unit. A third parameter, D_2, determines the maximum number of positions by which a state will be shifted under a limited scan operation, and it is fixed to a constant as described below. We denote by $shift(i, u)$ the number of bits by which the state at time unit u under test $\hat{\tau}_i \in TS(I, D_1)$ will be shifted. If $shift(i, u) = 0$, a limited scan operation does not occur at time unit u under $\hat{\tau}_i$, and primary input vectors continue to be applied at-speed at time unit u.

A test set $TS(I,D_1)$ for specific values of I and D_1 is defined using Procedure 1 given next. In Procedure 1, r_1 is a random number in the range $[0,R_1]$, where $R_1 \gg D_1$. We use $r_1 \; mod \; D_1$, where *mod* is the modulo operation, to obtain a number which is zero with probability $1/D_1$. When $r_1 \; mod \; D_1 = 0$, we insert a limited scan operation. In a similar way, we use $r_2 \; mod \; D_2$, where r_2 is in the range $[0,R_2]$, to obtain a random number between 0 and D_2-1. This number determines the number of shifts under a limited scan operation. Alternatively, *LFSR*s can be used to obtain these numbers in a hardware implementation of the random pattern generator.

Procedure 1: Defining the test set $TS(I,D_1)$

For every test $\tau_i \in TS_0$:

Initialize the random number generator with a seed $seed(I)$.

For $0 < u < L_i$, where L_i is the length of τ_i:

Select a random number r_1. If $r_1 \; mod \; D_1 = 0$:

Select a random number r_2. Set $shift(i,u) = r_2 \; mod \; D_2$.

Else, set $shift(i,u) = 0$.

Procedure 1 uses a constant D_2 to restrict the number of bits by which a state may be shifted under a limited scan operation. In our implementation, we use $D_2 = N_{SV}+1$, where N_{SV} is the number of state variables. This value of D_2 allows a limited scan operation to be anywhere between no scan (when $r_2 \; mod \; D_2 = 0$) and a complete scan operation (when $r_2 \; mod \; D_2 = N_{SV}$).

We select several combinations of I and D_1 to generate several test sets $TS(I,D_1)$ so as to maximize the fault coverage. This is done using Procedure 2 given next. Procedure 2 terminates when the fault coverage reaches 100%, or after a constant number of iterations with no improvement in the fault coverage. The constant is denoted by $N_{SAME.F.C}$.

Procedure 2: Selecting test sets $TS(I,D_1)$

(1) Let F be the set of target faults. Set $ID1\_PAIRS = \phi$.

(2) Generate TS_0. Simulate all the faults in F under TS_0 and drop from F the faults detected by TS_0. If $F = \phi$, stop.

(3) Set $I = 1$. Set $n_{same.f.c} = 0$.

(4) Set $improve.f.c = 0$. For $D_1 = 1,2,\cdots,10$:

(a) Call Procedure 1 to generate a test set $TS(I,D_1)$.

(b) Simulate all the faults in F under $TS(I,D_1)$ and drop from F the faults detected by $TS(I,D_1)$.

(c) If any faults are detected by $TS(I,D_1)$, add (I,D_1) to $ID1\_PAIRS$ and set $improve.f.c = 1$.

(d) If $F = \phi$, stop.

(5) If $improve.f.c = 0$, set $n_{same.f.c} = n_{same.f.c}+1$. Else, set $n_{same.f.c} = 0$.

(6) If $n_{same.f.c} < N_{SAME.F.C}$, set $I = I+1$ and go to Step 4.

Until now, we assumed that the values of L_A, L_B and N used to define TS_0 are given. To select values for L_A, L_B and N, we consider the number of clock cycles required for test application of TS_0 and the test sets $TS(I,D_1)$ selected by Procedure 2. Our goal is to select L_A, L_B and N so as to minimize the test application time while maximizing the fault coverage. Let us first consider the test application time of TS_0. We have $2N$ tests that require $2N+1$ complete scan operations, or $(2N+1)N_{SV}$ clock cycles. We have N tests of length L_A and N tests of length L_B, for a total of $N(L_A+L_B)$ primary input vectors. These vectors require $N(L_A+L_B)$ clock cycles. The number of clock cycles required for application of TS_0 is thus $N_{cyc_0} = (2N+1)N_{SV}+N(L_A+L_B)$. Here, we assume that the scan clock and the functional clock have the same cycle time. The formula for N_{cyc_0} can be adjusted to account for a faster functional clock if necessary.

Next, we consider a test set $TS(I,D_1)$. As part of $TS(I,D_1)$, we apply every test included in TS_0. This requires N_{cyc_0} clock cycles. In addition, limited scan operations have the following contribution to the test application time of $TS(I,D_1)$. At time unit u of $\hat{\tau}_i \in TS(I,D_1)$, we perform a limited scan operation that shifts the state by $shift(i,u)$ bits (if $shift(i,u) = 0$, no scan shifts are made at time unit u, and primary input vectors continue to be applied at-speed). Thus, the limited scan operation at time unit u requires $shift(i,u)$ clock cycles. Adding over all the tests and all the time units, limited scan operations contribute $\sum_{i=1}^{2N} \sum_{u=0}^{L_i} shift(i,u)$ clock cycles to the test application time (here, L_i is the length of T_i). We denote this number by $N_{SH}(I,D_1)$. The total number of clock cycles required for application of $TS(I,D_1)$ is $N_{cyc}(I,D_1) = N_{cyc_0}+N_{SH}(I,D_1)$.

Procedure 2 may select several pairs (I,D_1) that are included in the set $ID1\_PAIRS$. The test application time considering TS_0 and all the test sets $TS(I,D_1)$, $(I,D_1) \in ID1\_PAIRS$, is

$$N_{cyc} = N_{cyc_0} + \sum\{N_{cyc}(I,D_1) : (I,D_1) \in ID1\_PAIRS\}.$$

The tradeoff between L_A, L_B, N and the test application time is as follows. Smaller values of L_A, L_B and N result in smaller values of N_{cyc_0}. Since N_{cyc_0} appears in the total number of clock cycles N_{cyc} several times (by itself, and as part of every $N_{cyc}(I,D_1)$), it is important to keep N_{cyc_0} low by keeping L_A, L_B and N small. However, if L_A, L_B and N are too small, it may be necessary to use large numbers of (I,D_1) pairs in order to reach the desired fault coverage. In addition, especially for small values of D_1, the numbers of limited scan operations may be large, and the contribution of $N_{SH}(I,D_1)$ to the number of clock cycles may be high.

We study the tradeoff between L_A, L_B, N and the test application time experimentally by considering $s208$ and $s420$. Our goal is to achieve 100% fault coverage for these circuits. For each circuit, we consider $L_A \in \{8, 16, 32, 64, 128, 256\}$, $L_B \in \{16, 32, 64, 128, 256\}$, and $N \in \{64, 128, 256\}$. For every combination of L_A, L_B and N such that $L_A < L_B$, we show in Tables 3 and 4 the numbers of clock cycles N_{cyc} required for application of the test sets selected by Procedure 2. We also show the value of N_{cyc_0}. We enter a dash when Procedure 2 cannot achieve 100% fault coverage. We leave empty the entries where $L_A \geq L_B$. The following points can be seen from Tables 3 and 4.

The number of clock cycles for application of TS_0, N_{cyc_0}, increases with every one of L_A, L_B and N. Most of the time, N_{cyc} tracks this increase. However, occasionally, N_{cyc} is smaller for a larger value of N_{cyc_0}. For example, for $s208$ with $L_A = 8$, $L_B = 16$ and $N = 128$ we have $N_{cyc_0} = 5128$ and $N_{cyc} = 47317$, while with $L_A = 8$, $L_B = 16$ and $N = 256$ we have $N_{cyc_0} = 10248$ but $N_{cyc} = 32577$. This is because with $L_A = 8$, $L_B = 16$ and $N = 128$, we need four pairs (I,D_1), and the initial test set is applied four times with limited scan operations to achieve 100%

Table 3: Numbers of clock cycles for $s208$

				Ncyc		
	LA	LB=16	LB=32	LB=64	LB=128	LB=256
N=64	8	25450	27102	37876	43785	84720
	16		59370	62198	46246	87251
	32			50008	82707	92357
	64				61572	102616
N=128	8	47317	53037	46732	87799	169826
	16		51298	51862	92846	174928
	32			62103	103094	185353
	64				123661	206054
N=256	8	32577	52919	93863	175969	340592
	16		63166	104140	186417	350705
	32			124644	207039	370970
	64				247814	411950

				Ncyc0		
	LA	LB=16	LB=32	LB=64	LB=128	LB=256
N=64	8	2568	3592	5640	9736	17928
	16		4104	6152	10248	18440
	32			7176	11272	19464
	64				13320	21512
N=128	8	5128	7176	11272	19464	35848
	16		8200	12296	20488	36872
	32			14344	22536	38920
	64				26632	43016
N=256	8	10248	14344	22536	38920	71688
	16		16392	24584	40968	73736
	32			28680	45064	77832
	64				53256	86024

Table 4: Numbers of clock cycles for $s420$

				Ncyc		
	LA	LB=16	LB=32	LB=64	LB=128	LB=256
N=64	8	-	-	-	-	645573
	16		-	-	-	-
	32			-	-	880854
	64				-	469742
N=128	8	-	316472	475756	-	886564
	16		-	667376	730978	1051922
	32			397805	861565	1032612
	64				1012958	1435611
N=256	8	-	596377	-	1001374	1688693
	16		478143	723187	-	1614658
	32			885110	1425561	1517590
	64				986936	1771359

				Ncyc0		
	LA	LB=16	LB=32	LB=64	LB=128	LB=256
N=64	8	3600	4624	6672	10768	18960
	16		5136	7184	11280	19472
	32			8208	12304	20496
	64				14352	22544
N=128	8	7184	9232	13328	21520	37904
	16		10256	14352	22544	38928
	32			16400	24592	40976
	64				28688	45072
N=256	8	14352	18448	26640	43024	75792
	16		20496	28688	45072	77840
	32			32784	49168	81936
	64				57360	90128

fault coverage; whereas with $L_A = 8$, $L_B = 16$ and $N = 256$, we need only one pair (I, D_1) and the test set is applied only once with limited scan operations to achieve 100% fault coverage. Another effect that can be seen by considering $s420$ is that values of L_A, L_B and N that are too small may prevent us from achieving 100% fault coverage.

Since N_{cyc_0} has a strong effect on N_{cyc}, we prefer values of L_A, L_B and N that result in the smallest values of N_{cyc_0}. In Table 5, we show how N_{cyc_0} increases with L_A, L_B and N for several values of N_{SV} that appear in the benchmark circuits we consider. We use the values of L_A, L_B and N used in Tables 3 and 4. Table 5 shows the first 10 combinations of L_A, L_B and N by increasing value of N_{cyc_0}. To select an appropriate combination, we apply Procedure 2 with different combinations. The order by which we consider the various combinations is the one shown in Table 5, i.e., increasing value of N_{cyc_0}. If the goal is to achieve complete fault coverage, we stop when we find the first combination that allows us to achieve complete fault coverage. We explore additional combinations with increasing values of N_{cyc_0} in order to optimize parameters other than the fault coverage and the test application time.

Table 5: N_{cyc_0} as a function of L_A, L_B and N

NSV=21				NSV=74			
LA	LB	N	Ncyc0	LA	LB	N	Ncyc0
8	16	64	4245	8	16	64	11082
8	32	64	5269	8	32	64	12106
16	32	64	5781	16	32	64	12618
8	64	64	7317	8	64	64	14154
16	64	64	7829	16	64	64	14666
8	16	128	8469	32	64	64	15690
32	64	64	8853	8	128	64	18250
8	32	128	10517	16	128	64	18762
8	128	64	11413	32	128	64	19786
16	32	128	11541	64	128	64	21834

4. Experimental results

We considered ISCAS-89 and ITC-99 benchmark circuits. We first report the results obtained by Procedure 2 for the first combination of L_A, L_B and N that yields complete fault coverage. We then consider two parameters. (1) The ability of the resulting test sets to test the circuit at-speed. This is enhanced by using larger values of D_1 that increase the lengths of the primary input sequences applied between (limited and full) scan operations. It is important to note, however, that even if limited scan operations occur often in the test sets $TS(I, D_1)$, the test set TS_0 consists of tests of lengths L_A and L_B that are applied at-speed without any limited scan operations. (2) The number of pairs (I, D_1) that need to be used to achieve complete fault coverage. We consider different combinations of L_A, L_B and N in order to reduce this number.

The results for the first combination of L_A, L_B and N that yields complete fault coverage are reported in Table 6. In Table 6, after the circuit name, we show the values of L_A, L_B and N for which we report the results. Under column *initial* we show the number of faults detected by the initial test set TS_0, and the number of clock cycles N_{cyc_0} required for applying it. Under column *with lim. scan* we show the number of test sets $TS(I, D_1)$ applied with limited scan operations (this is also the number of pairs (I, D_1) selected by Procedure 2). We then show the number of faults detected using limited scan operations, and the number of clock cycles N_{cyc} required for applying all the test sets $TS(I, D_1)$. All the detectable circuit faults are detected in every case.

Table 6: Experimental results

circuit	LA,LB,N	initial det	initial cycles	with lim. scan app	with lim. scan det	with lim. scan cycles	with lim. scan ls
s208	8,16,64	178	2.6K	4	215	25.4K	0.55
s298	8,16,64	298	3.3K	1	308	13.0K	0.85
s344	8,16,64	342	3.5K	0			
s382	8,16,64	365	4.2K	1	399	19.0K	0.88
s400	8,16,64	377	4.2K	1	415	19.0K	0.88
s420	8,32,128	300	9.2K	12	430	316K	0.39
s510	8,16,64	564	2.3K	0			
s641	16,256,128	453	39.7K	9	467	2.4M	0.66
s820	16,32,64	597	3.7K	16	850	113K	0.37
s953	8,16,64	1038	5.3K	7	1079	87.5K	0.31
s1196	16,128,256	1235	46.1K	4	1242	870K	0.49
s1423	16,64,64	1200	14.7K	10	1501	1.2M	0.56
s5378	8,32,64	3908	25.7K	34	4563	3.8M	0.35
s35932	8,16,64	34681	224K	1	35110	1.4M	0.92
b01	8,16,64	130	2.2K	1	135	5.7K	0.76
b02	8,16,64	70	2.1K	0			
b03	8,16,64	382	5.4K	2	452	34.1K	0.49
b04	8,32,64	1251	11.1K	9	1344	575K	0.62
b06	8,16,64	202	2.7K	0			
b09	8,16,64	315	5.1K	14	420	181K	0.35
b10	8,16,64	450	3.7K	3	512	31.6K	0.50
b11	8,64,64	857	8.5K	10	1080	447K	0.51

det - number of detected faults
cycles - number of clock cycles required for test application
app - number of test sets applied with limited scan operations
ls - average number of time units where limited scan operations are performed

In the last column of Table 6, we show the average number of time units where limited scan operations are performed. This parameter is computed as follows. Consider the test set $TS = \cup\{TS(I,D_1):(I,D_1) \in ID1\_PAIRS\}$ that consists of all the tests in all the test sets selected by Procedure 2. Let $TS = \{\tau_1,\tau_2,\cdots,\tau_m\}$. Suppose that during τ_i, there are n_{ls_i} time units where limited scan operations occur (i.e., n_{ls_i} time units for which $shift(i,u) > 0$). We define the average number of time units with limited scan operations during the application of TS as $\overline{n_{ls}} = (\sum_{i=1}^{m} n_{ls_i})/(\sum_{i=1}^{m} L_i)$. We refer to this parameter as the average number of limited scan time units. This parameter indicates the average length of a primary input sequence applied at-speed between (complete and limited) scan operations. For example, with $\overline{n_{ls}} = 0.50$, a limited scan operation occurs every 1/0.50 = 2 time units on the average, which implies that on the average, primary input sequences of length two are applied at-speed. With $\overline{n_{ls}} = 0.10$, a limited scan operation occurs every 1/0.10 = 10 time units on the average, which implies that on the average, 10 consecutive primary input vectors are applied at-speed. It is important to note that the average is computed without taking TS_0 into account. In TS_0, we have primary input sequences of length L_A and L_B that contribute to at-speed testing of the circuit. The following points can be seen from Table 6.

Complete fault coverage is achieved by the proposed approach for all the circuits considered. The number of clock cycles required for achieving complete fault coverage is significantly increased compared to the initial test application time. However, the fault coverage is increased significantly as well. The number of clock cycles should be compared with the 500,000 clock cycles allocated in [5] and [6]. The following points related to [5] and [6] are important in this comparison. In [5] and [6], multiple scan chains are used such that the maximum length of a scan chain is 10. Consequently, at most 10 clock cycles are required for a complete scan operation in [5] and [6], and significantly fewer scan clock cycles are needed in [5] and [6] for every test compared to the method proposed here. In addition, in [5] and [6], the last flip-flop of every scan chain is observed at every time unit, thus improving the observability of the circuit. Nevertheless, incomplete fault coverages are reported in [5] and [6]. The proposed procedure uses fewer than 500,000 clock cycles for most of the circuits to achieve complete fault coverage when the fault coverage achieved in [5] and [6] is incomplete.

One case where the proposed procedure uses more than 500,000 clock cycles is $s5378$. For this circuit, average fault coverages of 98.66% and 98.54% are reported in [6] using the methods of [5] and [6], whereas the method proposed here achieves 99.13% fault coverage, which is the highest possible fault coverage for this circuit.

The parameter D_1 of Procedure 2 is the one that determines how often a limited scan operation occurs. In Procedure 2, we consider $D_1 = 1,2,\cdots,10$, in this order. Consequently, we give preference to the selection of smaller values of D_1 in order to detect target faults. A smaller value of D_1 results in more limited scan operations, and thus shorter primary input sequences applied at-speed between scan operations. To obtain longer primary input sequences applied at-speed, it is possible to consider $D_1 = 10,9,\cdots,1$ in Procedure 2. This would give preference to the selection of larger values of D_1, resulting in fewer limited scan operations. In Table 7, we show the results obtained for some of the circuits of Table 6 when $D_1 = 10,9,\cdots,1$ is considered in Procedure 2. The values of L_A, L_B and N are the same as in Table 6. Comparing the results of Table 7 with the results of Table 6, the following points can be seen.

Table 7: Using $D_1 = 10,9,\cdots,1$ in Procedure 2

circuit	with lim. scan app	with lim. scan det	with lim. scan cycles	with lim. scan ls
s208	10	215	42.0K	0.24
s298	3	308	13.1K	0.09
s344	0			
s382	3	399	17.7K	0.09
s400	3	415	17.7K	0.09
s420	17	430	383K	0.30
s510	0			
s641	9	467	2.1M	0.54
s820	20	850	130K	0.30
s953	9	1079	77.6K	0.15
s1196	5	1242	609K	0.22
s1423	13	1501	1.1M	0.38
s5378	42	4563	4.2M	0.33
b01	2	135	5.1K	0.08
b02	0			
b03	3	452	23.6K	0.11
b04	12	1344	607K	0.46
b06	0			
b09	14	420	163K	0.29
b10	8	512	54.3K	0.22
b11	14	1080	443K	0.33

app - number of test sets applied with limited scan operations
det - number of detected faults
cycles - number of clock cycles required for test application
ls - average number of time units where limited scan operations are performed

As may be expected, the average number of limited scan time units is lower when D_1 is considered in decreasing order. The number of clock cycles required for test application is sometimes higher and sometimes lower than that obtained when D_1 is considered in increasing order. This is due to two competing effects. (1) The number of different (I,D_1) pairs required to achieve complete fault coverage increases when D_1 is considered by decreasing order. This contributes to an increase in the number of clock cycles for test application. (2) Higher values of D_1 imply fewer limited scan operations, and therefore a lower number of clock cycles for test application.

For some of the circuits in Table 6, the number of (I,D_1) pairs required to achieve complete fault coverage is high. This implies that the number of (I,D_1) pairs that need to be stored is high. This number can be reduced by using larger values of L_A, L_B and/or N. To demonstrate this point, we show in Table 8 the results obtained for some of the circuits of Table 6 by using several different combinations of L_A, L_B and N. The first row for every circuit in Table 8 is always the same as that in Table 6.

Table 8: Different combinations of L_A, L_B and N

		initial		with lim. scan			
circuit	LA,LB,N	det	cycles	app	det	cycles	ls
s208	8,16,64	178	2.6K	4	215	25.4K	0.55
	8,32,64	185	3.6K	3	215	27.1K	0.46
	8,64,64	173	5.6K	2	215	37.9K	0.64
	8,128,64	167	9.7K	1	215	43.8K	0.87
s420	8,32,128	300	9.2K	12	430	316K	0.39
	16,64,128	308	14.4K	11	430	667K	0.53
	32,64,128	288	16.4K	6	430	398K	0.47
	64,256,64	258	22.5K	4	430	470K	0.54
	16,256,256	331	77.8K	3	430	1.6M	0.78
s641	16,256,128	453	39.7K	9	467	2.4M	0.66
	8,128,256	454	44.6K	8	467	1.6M	0.44
	16,256,256	454	79.4K	7	467	2.6M	0.42
s953	8,16,64	1038	5.3K	7	1079	87.5K	0.31
	8,32,64	1062	6.3K	4	1079	97.0K	0.44
	8,64,64	1076	8.3K	1	1079	72.7K	0.94
s1196	16,128,256	1235	46.1K	4	1242	870K	0.49
	32,128,256	1233	50.2K	2	1242	499K	0.52
s1423	16,64,64	1200	14.7K	10	1501	1.2M	0.56
	32,64,64	1286	15.7K	8	1501	1.1M	0.55
	8,128,64	1291	18.2K	5	1501	1.1M	0.59
	16,256,64	1283	27.0K	4	1501	1.8M	0.65
	8,256,128	1367	52.8K	2	1501	1.8M	0.65
	32,256,128	1333	55.9K	1	1501	1.4M	0.98
s5378	8,32,64	3908	25.7K	34	4563	3.8M	0.35
	16,32,64	3950	26.2K	27	4563	3.8M	0.41
	8,64,64	3953	27.7K	22	4563	4.1M	0.38
	32,64,64	3952	29.2K	19	4563	4.4M	0.37
	8,128,64	3975	31.8K	17	4563	5.8M	0.40
	16,128,64	3968	32.3K	15	4563	5.1M	0.37
	8,256,64	3968	40.0K	9	4563	6.4M	0.44
	64,256,64	3953	43.6K	7	4563	6.6M	0.49
	16,256,128	4084	80.8K	5	4563	7.3M	0.44
	64,256,128	4060	87.0K	3	4563	6.9M	0.60
	32,256,256	4169	165K	2	4563	10.2M	0.74
b09	8,16,64	315	5.1K	14	420	181K	0.35
	8,32,64	319	6.2K	9	420	209K	0.46
	8,64,64	315	8.2K	5	420	178K	0.41
	32,64,64	295	9.8K	3	420	181K	0.57
	16,128,64	270	12.8K	2	420	215K	0.71
	8,256,64	325	20.5K	1	420	254K	0.96

det - number of detected faults
cycles - number of clock cycles required for test application
app - number of test sets applied with limited scan operations
ls - average number of time units where limited scan operations are performed

From Table 8 it can be seen that it is possible to reduce the number of applications of the test set (and the number of (I,D_1) pairs) by using larger values of L_A, L_B and/or N; however, in several cases, the number of clock cycles required for test application increases as well.

5. Concluding remarks

We proposed a method of random pattern generation for at-speed testing of full scan circuits. We used tests that consist of a scan-in operation, followed by one or more primary input vectors applied at-speed, and a scan-out operation at the end of a test. The method was based on the insertion of limited scan operations into a random test set. Under a limited scan operation, the circuit state is shifted by a number of positions which may be smaller than the number of state variables. A limited scan operation can increase the number of faults detected by a test either because of the change in the circuit state, or because fault effects are scanned out. Under the proposed method, limited scan operations were inserted randomly, i.e., a random number of shifts was done at randomly selected time units. This ensures that the complete test set can be generated by a random pattern generator with simple control logic. Experimental results were presented to show that limited scan operations allow us to achieve complete fault coverages for benchmark circuits.

We considered full scan circuits in this work. However, limited scan can be used to improve the fault coverage for partial scan circuits as well.

References

[1] P. H. Bardell, W. H. McAnney and J. Savir, *Built-In Test for VLSI Pseudorandom Techniques*, Wiley, 1987.

[2] E. B. Eichelberger, E. Lindbloom, J. A. Waicukauski and T. W. Williams, *Structured Logic Testing*, Prentice-Hall, 1991.

[3] V. D. Agrawal, C. R. Kime and K. K. Saluja, "A Tutorial on Built-In Self-Test, Part 1: Principles", IEEE Design & Test of Computers, March 1993, pp. 73-82.

[4] V. D. Agrawal, C. R. Kime and K. K. Saluja, "A Tutorial on Built-In Self-Test, Part 2: Applications", IEEE Design & Test of Computers, June 1993, pp. 69-77.

[5] H.-C. Tsai, K.-T. Cheng and S. Bhawmik, "Improving the Test Quality for Scan-based BIST Using General Test Application Scheme", in Proc. Design Autom. Conf., June 1999, pp. 748-753.

[6] Y. Huang, I. Pomeranz, S. M. Reddy and J. Rajski, "Improving the Proportion of At-Speed Tests in Scan BIST", in Proc. Intl. Conf. on Computer-Aided Design, Nov. 2000, pp. 459-463.

[7] S. P. Morley and R. A. Martlett, "Selected Length Partial Scan: A Method to Reduce Vector Length", in Proc. Intl. Test Conf., Oct. 1991, pp. 385-392.

[8] P.-C. Chen, B.-D. Liu and J.-F. Wang, "Overall Consideration of Scan Design and Test Generation", in Proc. Intl. Conf. on Computer-Aided Design, Nov. 1992, pp. 9-12.

[9] Y. Higami, S. Kajihara and K. Kinoshita, "Reduced Scan Shift: A New Testing method for Sequential Circuits", in Proc. Intl. Test Conf., Oct. 1994, pp. 624-630.

[10] W.-J. Lai, C.-P. Kung and C.-S. Lin, "Test Time Reduction in Scan Designed Circuits", in Proc. European Design Autom. Conf., March 1993, pp. 489-493.

[11] C. Su and K. Hwang, "A Serial Scan Test Vector Compression Methodology", in Proc. Intl. Test Conf., Oct. 1993, pp. 981-988.