Reducing Test Application Time for Full Scan Embedded Cores *

Ilker Hamzaoglu and Janak H. Patel
Center for Reliable & High-Performance Computing
University of Illinois, Urbana, IL 61801

Abstract

We propose a new design for testability technique, Parallel Serial Full Scan (PSFS), for reducing the test application time for full scan embedded cores. Test application time reduction is achieved by dividing the scan chain into multiple partitions and shifting in the same vector to each scan chain through a single scan in input. The experimental results for the ISCAS89 circuits showed that PSFS technique significantly reduces both the test application time and the amount of test data for full scan embedded cores.

1 Introduction

Chips with multiple embedded cores, called *system-on-chip*, are becoming more prevalent in the industry, since the core-based design approach considerably reduces the design time through design reuse. However, testing embedded cores is a challenging problem, because of the limited access to the core peripheries, i.e. inputs and outputs [3, 6, 17, 18]. There are three major peripheral access techniques; *parallel direct access*, *functional access* and *serial boundary scan access* [18].

Parallel access technique provides direct access to core peripheries by either using dedicated chip I/O pins or by sharing the chip I/O pins by multiplexing and demultiplexing them with other signals. Functional access technique provides access to core peripheries by justifying the test patterns and propagating the fault effects using the user defined logic surrounding the core. Serial access technique provides indirect but full access to core peripheries by using a boundary scan chain around the core. Although the serial access technique increases the test application time for the cores due to serial access to the core peripheries, it is currently the most realistic approach for providing peripheral access to the embedded cores [18].

The full scan technique is a widely adopted DFT technique in the industry for reducing the complexity of test generation for sequential circuits to a combinational circuit test generation problem by making all the memory elements in the circuit both controllable and observable through a scan chain. However, since controlling (observing) the flip-flops are done by serially shifting in (out) the values to (from) the memory elements, full scan technique increases the test application time. Several techniques are proposed to reduce the test application time for full scan circuits, e.g. parallel direct access [9], hybrid test generation scheme [10, 15, 16], multiple scan chains [12, 15] and ordering flip-flops in a single scan chain [13]. However, these techniques are not effective for the full scan circuits used as embedded cores which we refer to as *full scan embedded cores*. The general structure of a full scan embedded core is shown in Figure 1.

Since hybrid test generation (HTG) technique uses sequential test generation, it generates large test sets. Since some of these vectors do not require shifting in the values to the memory elements and the input values are directly applied to the primary inputs, it reduces the test application time for standalone full scan circuits. However, if serial access technique is used for test access, then the test application time can be quite large because the primary input values in each vector should be serially shifted into the boundary scan. In addition, since it generates a large test set, HTG scheme makes it more difficult to use the functional access technique by increasing the number of vectors that have to be justified at the core inputs. Finally, since HTG technique requires sequential test generation, it may not be applicable to large cores.

Multiple scan chains (MSC) technique requires separate input/output pins for each scan chain. This requirement makes it more difficult to use the parallel access technique by increasing the number of core pins and the routing overhead. The same requirement makes it more difficult to use the functional access technique too by increasing the number of values that should be justified at the core inputs. On the other hand, if the serial access technique is used for test access, the MSC technique does not reduce the test application time. Because the input values that will be shifted into each scan chain first should be seri-

*This research was supported in part by the Semiconductor Research Corporation under contract SRC 97-DJ-482 and in part by DARPA under contract DABT63-95-C-0069.

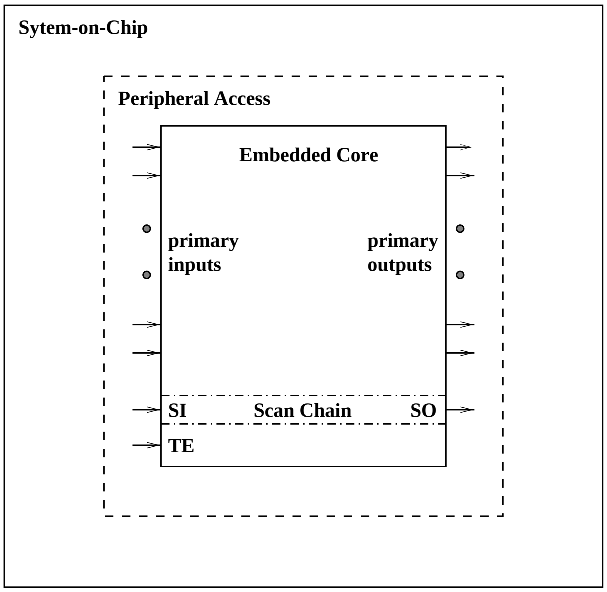

Figure 1: Full Scan Embedded Core

ally shifted into the boundary scan and then shifted into the scan chains in parallel. Similarly, the test responses in the scan chains first should be shifted out to the boundary scan in parallel and then serially shifted out from the boundary scan. Therefore, the test application time becomes the same as the test application time for full scan cores using a single scan chain.

Parallel direct access to all the memory elements are impractical for embedded cores, since even the direct access to the core inputs and outputs are limited. Even though single scan chain reconfiguration technique reduces the test application time to some degree, it is possible to reduce the test application time further.

In this paper, we propose a new DFT technique, *Parallel Serial Full Scan (PSFS)*, for reducing the test application time for the full scan embedded cores. Since the PSFS technique does not use any additional test access pins other than the ones used in the full scan technique and it does not require large number of test vectors, it is applicable to the full scan cores using any of the three peripheral access techniques; serial, parallel or functional access.

The PSFS techniques reduces the test application time by dividing the scan chain into multiple partitions and shifting in the same vector to each scan chain through a single scan in input. The outputs of the scan chains are observed through a multiple input signature analyzer (MISR). In order to test the faults that cannot be detected by this organization and the faults aliased by the MISR, PSFS technique also preserves the single scan chain structure using extra multiplexers and a simple control structure. Since the performance of the PSFS technique is dependent on the scan chain configuration, i.e. the number of the scan chains used and the assignment of the memory elements to the scan chains, we propose a heuristic technique for computing an optimal scan chain configuration that produces a minimal test application time.

In addition to reducing the test application time, PSFS technique has another important advantage for testing full scan cores. Since for each test vector, in addition to the primary input values, only the input values for the longest scan chain should be stored in the tester, PSFS technique reduces the tester storage requirements and the amount of test data that should be transfered from the tester to the core and from the core back to the tester. This is especially important for large system-on-chips with multiple cores [7].

We enhanced ATOM, our advanced ATPG system for combinational circuits [4], to generate test vectors under the single stuck at fault model for the embedded cores incorporating the PSFS technique. The experimental results showed that for the ISCAS89 circuits the PSFS technique reduces the total test application time by 82%, and it reduces the total test data by 67%.

In addition to significantly reducing the test application time and the amount of test data, the PSFS technique is also very effective in terms of the other performance metrics. In particular, it only uses combinational test generation, thus it is applicable to large cores. It doesn't degrade the performance of the core in normal mode more than the full scan technique. It doesn't require any additional test access pins other than the scan in, scan out and test enable pins used by the full scan technique. It only incurs a small amount of hardware overhead, and it provides 100% combinational fault coverage.

In this paper, we applied the PSFS technique only to the flip-flops of a full scan core, and we assumed that the inputs and the outputs of the core will be accessed by any one of the peripheral access techniques. However, if the serial boundary scan access technique is used for test access to the core, then it is possible to apply the PSFS technique to both the flip-flops of the core and the boundary scan flip-flops for the inputs and the outputs of the core by considering all of these flip-flops as a single scan chain and by dividing this scan chain into multiple partitions and shifting in the same vector to each scan chain through a single scan in input and observing the outputs of the scan chains through a MISR. This way the test application time can be reduced further.

A similar technique to PSFS is proposed in [11] for testing multiple independent circuits simultaneously by using a single input to support the scan chains in all

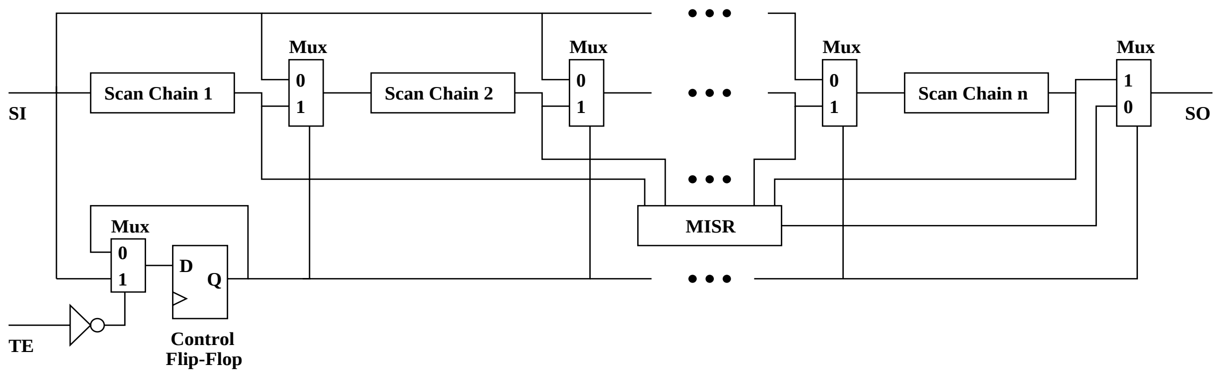

Figure 2: Parallel Serial Full Scan (PSFS) Technique

the circuits. The number of scan chains is equal to the number of circuits, and the memory elements of each circuit is assigned to its scan chain. The only design parameter is the order of the memory elements in each scan chain. Since the circuits are independent, all the detectable faults in each circuit are still detectable under this organization. PSFS technique, on the other hand, is proposed for testing a single full scan embedded core. The number of scan chains, the distribution of the memory elements to the scan chains, and the order of the memory elements in each scan chain are all design parameters for the PSFS technique. In addition, since PSFS technique is applied to a single full scan core, some of the detectable faults in the core may not be detectable when the same logic values are shifted into the multiple scan chains in the core.

The rest of the paper is organized as follows. Section 2 presents the PSFS technique. The algorithm for finding an optimal scan chain configuration is described in Section 3. Section 4 describes the test generation algorithm used to generate a minimal test set for a given scan chain configuration. The experimental results are given in Section 5. Finally, Section 6 presents the conclusions.

2 Parallel Serial Full Scan Technique

We propose a new DFT technique, *Parallel Serial Full Scan (PSFS)*, for reducing the test application time for full scan embedded cores. The full scan embedded cores incorporating the PSFS technique are called *PSFS cores*. The organization of the flip-flops of a PSFS core is shown in Figure 2. The PSFS technique reduces the test application time for the embedded cores without using any additional test access pins other than the ones used in the full scan technique, i.e. scan in (SI), scan out (SO), and test enable (TE).

As in the full scan cores, the TE input controls the operation of the PSFS cores. When the TE is 0, the core operates in the normal mode. The operation of the PSFS cores in the normal mode is same as the full scan cores. When the TE is 1, the core operates in the test mode. The operation of the PSFS cores in the test mode is controlled by the control flip-flop (CFF). When the CFF is 1, the PSFS core operates in the *serial test mode*. The operation of the PSFS core in the serial test mode is same as the operation of the full scan cores in the test mode, i.e. the new values are shifted into each flip-flop serially through SI input, and the previous values of all the flip-flops are shifted out serially through SO output. When the CFF is 0, the PSFS core operates in the *parallel test mode*. In the parallel test mode, shifting in the input values into the flip-flops and shifting out the test response from the flip-flops are done in parallel.

We will now explain the operation of the PSFS cores in the test mode in more detail. PSFS technique divides the single scan chain in a full scan core into multiple partitions. The input of the first scan chain is driven by the SI input. The input of the other scan chains is driven either by the output of the previous scan chain or by the SI input. The selection is done by a 2-to-1 multiplexer (MUX) which is controlled by the CFF. When the CFF is 0, each scan chain is driven by the SI input, thus the same logic values are shifted into each scan chain in parallel. When the CFF is 1, each scan chain is driven by the previous scan chain, thus the flip-flops behave as a single serial scan chain.

The output of each scan chain is connected both to the MUX driving the next scan chain and to a multiple input signature analyzer (MISR). The PSFS technique uses the MISR to compress the test response of the embedded core that is stored in its flip-flops. Since the output of each scan chain is connected to a different data input of the MISR, the size of the MISR used in a PSFS core is equal to the number of scan chains used in the core. In this work, we used two different type 2 (internal-XOR) MISRs, a 6-bit and a 16-bit MISR with the primitive characteristic polynomials $P(x) =$

x^6+x+1 and $P(x) = x^{16}+x^9+x^7+x^4+1$ respectively.

The SO output of the PSFS core is driven either by the output of the last scan chain or by the output of the MISR. The selection is done by a 2-to-1 MUX which is controlled by the CFF. When the CFF is 0, the output of the MISR is shifted out through the SO output. When the CFF is 1, SO output is driven by the output of the last scan chain, thus the flip-flops behave as a single serial scan chain.

In summary, when the CFF is 0, the same logic values are shifted into each scan chain in parallel through the SI input, the outputs of the scan chains are compressed using a MISR, and the output of the MISR is observed through the SO output. On the other hand, when the CFF is 1, the flip-flops behave as a single serial scan chain, i.e. the new values are shifted into each flip-flop serially through SI input, and the previous values of all the flip-flops are shifted out serially through SO output.

The parallel test mode is used to reduce the test application time. However, since the same logic values are shifted into each scan chain, some of the testable faults may not be detected under the parallel test mode. Moreover, the use of a MISR for response compaction may cause some of the faults to be aliased. The serial test mode is used to test these faults, thus providing 100% combinational fault coverage. The test vectors applied under the serial test mode are called *serial vectors*, and the test vectors applied under the parallel test mode are called *parallel vectors*.

3 Scan Chain Configuration

The effectiveness of the PSFS technique for reducing the test application time is dependent on the scan chain configuration, i.e. the number of the scan chains used and the assignment of the flip-flops to the scan chains. Therefore, we propose a heuristic technique for computing an optimal scan chain configuration that provides a minimal test application time.

3.1 Number of Scan Chains

As the number of scan chains increases, the amount of parallelism also increases and the parallel vector length decreases, thus decreasing the test application time. However, using a large number of scan chains increases the number of faults that cannot be tested using the parallel test mode, thus increasing the number of serial vectors, which starts increasing the test application time. As the number of scan chains increases, the hardware overhead of the PSFS technique also increases, i.e. large number of MUXes, a large MISR and extra routing are needed.

Since, for a given core, it is computationally prohibitive to compute the optimum number of scan chains, in this work we selected the number of scan chains by trying to avoid using a small or a large number of scan chains relative to the number of flip-flops used in the core. Based on this criterion, in our experiments, we used 6 scan chains for the cores with less than 200 flip-flops, and 16 scan chains for the cores with less than 2000 flip-flops.

3.2 Mapping Flip-Flops to Scan Chains

Finding the optimum scan chain configuration that will produce the minimum number of test cycles is an NP-hard problem. Because it requires solving the problem of generating a minimum size stuck at test set for a given combinational circuit which is proven to be NP-hard [8]. Even for the special case of evenly distributing the flip-flops to the scan chains, the number of different scan chain configurations with possibly different test application times is

$$\prod_{N_{SC}=1}^{MaxN_{SC}} \frac{F!}{\left(\frac{F}{N_{SC}}\right) * N_{SC}!}$$

where F is the total number of flip-flops, N_{SC} is the number of scan chains used, and $MaxN_{SC}$ is the maximum number of scan chains that can be used. Thus, it is computationally prohibitive to search the entire scan chain configuration space for finding the optimum scan chain configuration. Therefore, we propose a heuristic mapping technique for finding an optimal scan chain configuration. The heuristic assumes that the number of scan chains in the circuit (N_{SC}) is given and it tries to map the flip-flops to these scan chains such that a minimal test application time is achieved.

Two inputs of a circuit are said to be *compatible* if they can be shorted together without introducing any redundant stuck at fault in the circuit, otherwise these two inputs are called *incompatible* [2]. In the parallel testing mode, the corresponding flip-flops in each scan chain is assigned the same logic value. This restriction may prevent the detection of some of the combinationally testable faults thus increasing the test application time by increasing the number of serial vectors necessary to achieve 100% fault coverage. However, if all of these flip-flops are compatible with each other, then this restriction cannot introduce any artificial redundancies. Therefore, our heuristic mapping technique tries to assign pairwise compatible flip-flops to the same position in each scan chain.

Since the exact techniques for checking the compatibility of two flip-flops are computationally expensive, they may not be applicable to large circuits. Therefore, we used the heuristic technique described in [2] for deciding whether two flip-flops are compatible or not.

$$\begin{array}{c|cccc|}
 & x_1 & x_2 & x_3 & x_4 \\
\hline
t_1 & X & 1 & 0 & 1 \\
t_2 & 1 & 0 & 1 & 1 \\
t_3 & 0 & 0 & 0 & 0 \\
t_4 & 1 & 1 & X & 0 \\
t_5 & X & 0 & X & X \\
\hline
\end{array}$$

Figure 3: Partially Specified Test Set Matrix

The heuristic technique first generates a complete partially specified test set for the full scan circuit without using any test set compaction algorithm. The resulting test set can be represented as a two dimensional matrix where each row is a test vector and each column is the values that will be assigned to a single flip-flop of the circuit. Two columns, in this matrix, are compatible if for every row their corresponding logic values are same or at least one of them is a don't care (X). An example test set matrix is shown in Figure 3 in which only the columns x_1 and x_3 are pairwise compatible. If two columns in a given test set matrix are compatible, then the corresponding two flip-flops are compatible. However, if two columns are incompatible, it is not possible to conclude that the corresponding flip-flops are incompatible. Because it is possible that in a different test set the columns corresponding to these two flip-flops may be compatible. Therefore, this heuristic technique may fail to prove the compatibility of two compatible flip-flops.

A *compatibility class* is a set of flip-flops that are pairwise compatible. Assigning pairwise compatible flip-flops to the same position in each scan chain requires finding the compatibility classes of size at most N_{SC}. We used a common greedy heuristic for solving this problem. The heuristic technique considers each flip-flop one at a time. The first flip-flop is assigned to the first compatibility class. If the next flip-flop is compatible with the first one, it is also assigned to the same compatibility class, otherwise it is assigned to a new compatibility class. If the next flip-flop is compatible with all the flip-flops in an existing compatibility class which has less than N_{SC} flip-flops, then it is added to that class, otherwise it is assigned to a new compatibility class and so on.

Once the compatibility classes are computed, our heuristic mapping technique uses this information for mapping the flip-flops to the scan chains. It first sorts the compatibility classes in the descending order of their cardinality. Starting from the first position in each scan chain, it assigns the flip-flops in each compatibility class of size N_{SC} to the same position in each scan chain, i.e. the flip-flops in the first compatibility class of size N_{SC} is assigned to the first position in each scan chain and so on. Then, it assigns the flip-flops in each compatibility class of size $S > \frac{2}{3} * N_{SC}$ to the same position in the first S scan chains and fills in the same position in the remaining scan chains using the flip-flops in the compatibility classes of size $S \leq \frac{2}{3} * N_{SC}$. Finally, it assigns the flip-flops in each of the remaining compatibility classes of size $S \leq \frac{2}{3} * N_{SC}$ to the same position in the first S scan chains leaving the same position in the remaining scan chains empty.

The second step of the heuristic mapping technique may cause some of the faults to be untestable in the parallel test mode, since some of the flip-flops that are assigned to the same position in each scan chain may be incompatible. This increases the number of serial vectors necessary to achieve 100 % fault coverage, thus increasing the test application time. However, this assignment reduces the size of the longest scan chain, thus reducing the test application time. The number of artificial redundancies introduced and the amount of reduction achieved in the longest scan chain size depends on the number of incompatible flip-flops assigned to the same position in each scan chain. If this number is large, then this may increase the test application time more than it helps to decrease it. Therefore, in order to reduce the size of the longest scan chain without causing a large number of artificial redundancies, the heuristic technique assigns at most $\frac{1}{3} * N_{SC}$ incompatible flip-flops to the same position in each scan chain.

In order to assess the effectiveness of the heuristic optimal mapping technique, we also measured the performance of the PSFS technique using a different mapping technique, called default mapping, which does not try to find an optimal scan chain configuration that produces a minimal test application time. The default mapping technique distributes the flip-flops evenly to each scan chain using the flip-flop order given in the circuit description, i.e. the flip-flops 1 to n are assigned to scan chain 1, the flip-flops $n + 1$ to $2n$ are assigned to scan chain 2, and so on, where $n = (number\, of\, flip-flops)\,/\,(number\, of\, scan\, chains)$.

Example: Consider a full scan embedded core with 27 flip-flops. Suppose that 6 scan chains are used in the circuit. As shown in Figure 4, the default mapping technique assigns the flip-flops evenly to each scan chain using the flip-flop order given in the circuit description, i.e. 1 to 27. Suppose that the heuristic for computing compatibility classes computed 7 compatibility classes for the core flip-flops as shown in Figure 4. The optimal mapping technique assigns the flip-flops to the scan chains using this compatibility in-

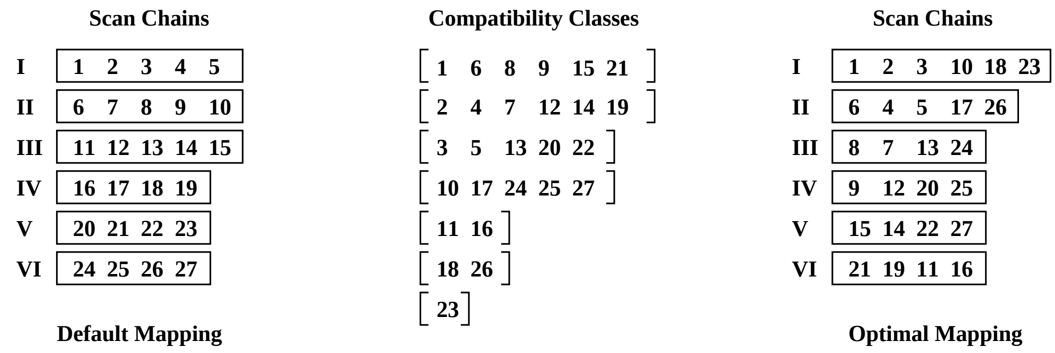

Figure 4: Scan Chain Configuration Example

formation as explained in this section. The resulting scan chain configuration is shown in Figure 4.

4 Test Generation

The problem of generating the minimum size test set for a given scan chain configuration is NP-hard, because it requires solving the problem of generating a minimum size stuck at test set for a given combinational circuit which is proven to be NP-hard [8]. In this work, therefore, we used the heuristic test set compaction algorithms dynamic compaction and redundant vector elimination [5] for generating compact test sets.

Test generation process for a given scan chain configuration proceeds as follows. First, a minimal test set that detects all the faults that are detectable under the parallel test mode is generated. During the test generation process for the parallel test mode, the test generation constraints imposed by the scan chain configuration, i.e. the flip-flops assigned to the same position in each scan chain should be assigned the same logic value in each test vector, are taken into account. Next, the MISR is fault simulated for the vectors in this test set and the faults aliased by the MISR are identified. Finally, using the serial test mode a minimal test set that detects the aliased faults and the faults that are undetectable using the parallel test mode is generated. The final test set is the union of the test set with the parallel test vectors and the one with the serial test vectors.

5 Experimental Results

In order to generate test vectors for the PSFS cores under the single stuck at fault model, we enhanced ATOM, our advanced ATPG system for combinational circuits [4], and we implemented a sequential circuit fault simulator, based on PROOFS algorithm [14], for simulating a MISR and incorporated it into ATOM. ATOM and the new extensions are implemented in C++. The test application time for the small ISCAS89 circuits is already very small, because the number of flip-flops and the size of the minimal test set are quite small for these circuits [5]. Therefore, we tested ATOM on the large ISCAS89 circuits [1]. We assumed that these circuits are used as full scan embedded cores. The performance results are obtained on a 200 MHz Pentium Pro PC with 128MB RAM running Linux 2.0.0 using GNU CC version 2.8.0. In all the experiments, a backtrack limit of 6 is used in ATOM, and all the test sets generated by ATOM have 100% fault coverage.

The experimental results for the full scan embedded cores are presented in Table 1. The last four columns in this table present the number of test vectors in the minimal test set, the number of cycles necessary to test the core, the amount of test data in number of bits that should be stored in the tester for testing the core, and the test generation time in seconds respectively. The test application time for a full scan core is computed as $F + (1 + F) * V$, where F is the number of flip-flops and V is the number of test vectors.

The experimental results for the PSFS cores using the default and the optimal scan chain configuration techniques are presented in Tables 2 and 3 respectively. The columns in these tables present the core name, the number of scan chains used, the number of flip-flops in the longest scan chain, the number of parallel test vectors, the number of faults that cannot be detected under the parallel test mode, the number of faults aliased by the MISR, the number of serial test vectors used for testing both the aliased faults and the faults that cannot be detected under the parallel test mode, the number of cycles necessary to test the core, the amount of test data in number of bits that should be stored in the tester for testing the core, and the test generation time in seconds respectively. The experimental results for the PSFS cores presented in these tables are obtained by assuming that the parallel access technique is used for test access to the core inputs and outputs. Therefore, the

Core	Inputs	Outputs	Flip-Flops	Test Vectors	Test Cycles	Test Data (bits)	Time(s)
s5378	35	49	179	111	**20159**	**49062**	11.5
s9234	36	39	211	159	**33919**	**79023**	57.4
s13207	62	152	638	236	**151442**	**351640**	68.9
s15850	77	150	534	126	**67944**	**163170**	156.3
s35932	35	320	1728	16	**29392**	**60976**	117.3
s38417	28	106	1636	99	**163699**	**337194**	148.0
s38584	38	304	1426	136	**195498**	**434384**	215.5

Table 1: Performance Results for Full Scan Embedded Cores

Core	Scan Chains	Longest Scan Chain	Parallel Test Vectors	Undetectable Faults	Aliased Faults	Serial Test Vectors	Test Cycles	Test Data (bits)	Time(s)
s5378	6	30	132	65	0	28	**9349**	**31384**	18.1
s9234	6	36	181	932	1	57	**19036**	**54936**	156.6
s13207	16	40	136	1174	6	137	**93815**	**244114**	119.2
s15850	16	34	152	472	0	26	**19816**	**78510**	205.4
s35932	16	108	31	0	0	0	**5233**	**17701**	231.2
s38417	16	103	303	116	0	34	**88927**	**218824**	720.5
s38584	16	90	190	891	0	49	**88747**	**255686**	472.7

Table 2: Performance Results for PSFS Cores Using Default Scan Chain Configuration

Core	Scan Chains	Longest Scan Chain	Parallel Test Vectors	Undetectable Faults	Aliased Faults	Serial Test Vectors	Test Cycles	Test Data (bits)	Time(s)
s5378	6	33	186	0	1	1	**6724**	**28342**	22.4
s9234	6	44	214	0	0	0	**9893**	**34882**	109.6
s13207	16	42	277	0	0	0	**12609**	**82546**	124.6
s15850	16	40	235	6	1	3	**11832**	**76030**	314.2
s35932	16	108	16	0	0	0	**3598**	**9136**	238.9
s38417	16	140	264	2	16	6	**48840**	**129732**	593.5
s38584	16	90	236	0	8	2	**25864**	**129580**	526.6

Table 3: Performance Results for PSFS Cores Using Optimal Scan Chain Configuration

test application time for a PSFS core is computed as $1+F_{LSC}+(1+F_{LSC})*V_P+S+1+F+(1+F)*V_S$, where F is the number of flip-flops in the circuit, F_{LSC} is number of flip-flops in the longest scan chain, S is the number of scan chains, V_P is the number of parallel test vectors and V_S is the number of serial test vectors.

The test generation time for the PSFS cores using the heuristic optimal mapping method presented in Table 3 does not include the test generation time for generating the partially specified test sets. It includes the time to compute the optimal mapping given a partially specified test set and the test generation time using this scan chain configuration.

The comparisons of the test application time and the amount of test data for the full scan cores and the PSFS cores are presented in Table 4.

The results show that the PSFS technique is quite effective in reducing the test application time and the test storage requirements for the ISCAS89 circuits even if the default mapping method is used. The test generation time for the default scan chain configuration is also quite small. Though this may not be the case for every full scan embedded core, if the default mapping method provides a satisfactory test application time reduction, this avoids the design time for using the heuristic optimal mapping technique. These results also show that for some embedded cores the layout information can be used to configure the scan chains to reduce the routing overhead of the PSFS technique without sacrificing the advantage of reduced test application time.

The results also show that the PSFS technique using the heuristic optimal mapping method significantly reduces the test application time and the test storage requirements for full scan cores. For the ISCAS89 circuits, the total test application time is reduced by 82%, and the total test data is reduced by 67%. The results show that the heuristic optimal mapping technique is more effective than the default mapping technique. The design time for computing the optimal scan chain

	Test Application Time					Test Data				
	Full Scan	PSFS with Default Scan Chain		PSFS with Optimal Scan Chain		Full Scan	PSFS with Default Scan Chain		PSFS with Optimal Scan Chain	
Core	Test Cycles	Test Cycles	Percent Reduction	Test Cycles	Percent Reduction	Test Data (bits)	Test Data (bits)	Percent Reduction	Test Data (bits)	Percent Reduction
s5378	20159	9349	54 %	6724	67%	49062	31384	36%	28342	42%
s9234	33919	19036	44%	9893	71%	79023	54936	31%	34882	56%
s13207	151442	93815	38%	12609	92%	351640	244114	31%	82546	77%
s15850	67944	19816	71%	11832	83%	163170	78510	52%	76030	53%
s35932	29392	5233	82%	3598	88%	60976	17701	71%	9136	85%
s38417	163699	88927	46%	48840	70%	337194	218824	35%	129732	62%
s38584	195498	88747	55%	25864	87%	434384	255686	41%	129580	70%

Table 4: Comparison of Performance Results

configuration and the test generation time for this configuration is also very small.

The effectiveness of the PSFS technique for a core depends on the existence of the compatible flip-flops in that core. Even though the PSFS technique is very effective for the ISCAS89 circuits, if the flip-flops of a full scan core is highly incompatible, then it may not be very effective for reducing the test application time for that full scan core.

The results also showed that for some of the circuits parallel vectors alone achieved 100% fault coverage and the MISR did not cause any aliasing. Since no serial vectors are needed for these circuits, there is no need to preserve the single serial scan structure using the extra multiplexers and the control structure. Therefore, the PSFS technique incurs a much smaller hardware overhead for these circuits.

6 Conclusions

We proposed a new DFT technique, Parallel Serial Full Scan (PSFS), for reducing the test application time for full scan embedded cores without using any additional test access pins other than the ones used for the full scan technique. The experimental results for the ISCAS89 circuits showed that this technique significantly reduces both the test application time and the amount of test data for full scan embedded cores.

References

[1] F. Brglez, D. Bryan, and K. Kozminski, "Combinational Profiles of Sequential Benchmark Circuits", in *Proc. of the Int. Symp. on Circuits and Systems*, pp. 1929-1934, May 1989.

[2] C. Chen and S. K. Gupta, "A Methodology to Design Efficient BIST Test Pattern Generators", in *Proc. of the Int. Test Conf.*, pp. 814-823, October 1995.

[3] I. Ghosh, N. K. Jha, and S. Dey, "A Low Overhead Design for Testability and Test Generation Technique for Core-based Systems", in *Proc. of the Int. Test Conf.*, pp. 50-59, October 1997.

[4] I. Hamzaoglu and J. H. Patel, "Deterministic Test Pattern Generation Techniques", in *Proc. of the VLSI Test Symp.*, pp. 446-452, April 1998.

[5] I. Hamzaoglu and J. H. Patel, "Test Set Compaction Algorithms for Combinational Circuits", in *Proc. of the Int. Conf. on Computer-Aided Design*, pp. 283-289, November 1998.

[6] V. Immaneni and S. Raman, "Direct Access Test Scheme - Design for Block and Core Cells for Embedded ASICS", in *Proc. of the Int. Test Conf.*, pp. 488-492, October 1990.

[7] A. Jas and N. A. Touba, "Test Vector Decompression via Cyclical Scan Chains and Its Application to Testing Core-Based Designs", in *Proc. of the Int. Test Conf.*, pp. 458-464, October 1998.

[8] B. Krishnamurthy and S. B. Akers, "On the Complexity of Estimating the Size of a Test Set", *IEEE Trans. on Computers*, pp. 750-753, August 1984.

[9] S. Lee and K. G. Shin, "Design for Test Using Partial Parallel Scan", *IEEE Trans. on Computer-Aided Design*, pp. 203-211, February 1990.

[10] S. Y. Lee and K. K. Saluja, "An Algorithm to Reduce Test Application Time in Full Scan Designs", in *Proc. of the Int. Conf. on Computer-Aided Design*, pp. 17-20, November 1992.

[11] K. Lee, J. Chen, and C. Huang, "Using a Single Input to Support Multiple Scan Chains", in *Proc. of the Int. Conf. on Computer-Aided Design*, pp. 74-78, November 1998.

[12] S. Narayanan, R. Gupta, and M. Breuer, "Optimal Configuring of Multiple Scan Chains", *IEEE Trans. on Computer-Aided Design*, pp. 1121-1131, September 1993.

[13] S. Narayanan and M. Breuer, "Reconfiguration Techniques for a Single Scan Chain", *IEEE Trans. on Computer-Aided Design*, pp. 750-765, June 1995.

[14] T. M. Niermann, W. Cheng, and J. H. Patel, "PROOFS: A fast, memory-efficient sequential circuit fault simulator", *IEEE Trans. on Computer-Aided Design*, pp. 198-207, February 1992.

[15] D. K. Pradhan and J. Saxena, "A Design for Testability Scheme to Reduce Test Application Time in Full Scan", in *Proc. of the VLSI Test Symp.*, pp. 55-60, April 1992.

[16] E. M. Rudnick and J. H. Patel, "A Genetic Approach to Test Application Time Reduction for Full Scan and Partial Scan Circuits", in *Proc. of Int. Conf. on VLSI Design*, pp. 288-293, January 1995.

[17] N. A. Touba and B. Pouya, "Testing Embedded Cores Using Partial Isolation Rings", in *Proc. of the VLSI Test Symp.*, pp. 10-16, April 1997.

[18] Y. Zorian, "Test Requirements for Embedded Core-based Systems and IEEE P1500", in *Proc. of the Int. Test Conf.*, pp. 191-199, October 1997.