Clock Grouping: A Low Cost DFT Methodology for Delay Testing

Wen-Chang Fang and Sandeep K. Gupta

CENG Technical Report 94-04

Department of Electrical Engineering - Systems
University of Southern California
Los Angeles, California 90089-2562
(213)740-2251

March 1994

Clock Grouping: A Low Cost DFT Methodology for Delay Testing

Abstract

A low overhead DFT technique, called clock-grouping, for delay testing of sequential synchronous circuits is presented. The proposed technique increases robust path delay fault coverage for circuits by exercising greater control over flip-flop clocks in the test mode. In the test mode, the flip-flops are partitioned into different clock-groups. The flip-flops in each clock group can be either clocked or not clocked, independent of the flip-flops in the other groups. This flexibility is used to enhance the number of different (v_1, v_2) test pairs that can be applied to the state inputs of the circuit thereby increasing coverage of delay faults. Experimental data on benchmark circuits shows that high fault coverage can be obtained by using only two clock groups in most circuits.

The proposed clock grouping methodology can be applied to non-scan circuits as well. In fact, it can provide a DFT solution for high speed data path circuits where the performance penalties of conventional DFT techniques are unacceptable.

1 Introduction

Growing circuit speeds are making it essential to test circuits for delay faults. With the use of aggressive statistical timing, a certain fraction of manufactured chips fail to perform at the desired clock rate *even when all devices perform within specified performance tolerance.* With the popularization of MCM technology, it is becoming imperative to ensure that the dies are not only free of any logical failures, but are also guaranteed to perform at the specified clock rates. All these reasons and the general concern for improving quality of the shipped products is leading to the popularization of delay testing.

Delay testing differs from the conventional (stuck-at) testing significantly since it requires two pattern tests (v_1, v_2) [LR87, Smi85]. The initialization vector (v_1) is applied to the circuit to obtain desired initial values at various circuit nodes by charging/discharging appropriate node capacitances. The test vector (v_2) is then applied to launch the desired transition to a circuit output. Under path delay fault model [LR87, Smi85] the transition is created along the target path, starting at a primary (state) input and ending at a primary (state) output. A (v_1, v_2) pair is said to be a robust test for a given path delay fault if cannot be invalidated by any other delays in the circuit [LR87].

Since the purpose of applying v_1 is to initialize circuit nodes to desired values, typically presence of a hazard between v_1 and v_2 at any primary input can jeopardize initialization. Hence, the application of two pattern test (v_1, v_2) at inputs to the combinational logic should be hazard free. Examples where presence of hazard between (v_1, v_2) can cause test invalidation can be found in [Pra91].

Design for testability (DFT) techniques to enable comprehensive delay testing of synchronous sequential circuits differ significantly from the scan (or partial scan) techniques employed for stuck-at testing. The need to apply two-pattern tests at inputs of the combinational logic, hazard free, adds new dimensions to the DFT problem. This paper presents a low cost DFT methodology which can enable application of desired test patterns to the circuit input. We begin by a discussion of existing DFT methodologies for delay testing.

Review of Related Research It was proposed in [Mal83] that the scan chain be enhanced by the use of additional hardware to enable shifting in any desired v_1 as well as v_2, for each state input of the circuit. Any two patterns can be applied to the combinational inputs of the circuit in succession. This DFT technique guarantees detection of all detectable delay faults in the combinational part of the CUT. The test generation can be carried out using delay test generator for combinational circuits. However the area overhead of this scheme is very high.

Recently, two main test application strategies have been considered. The first one is called functional justification (e.g. see [CDK91a]). In this scheme, the circuit flip-flops are designed the same way as they are in conventional scan design for stuck-at testing. In the test mode, the flip-flops are configured as a scan chain to shift in a desired v_1 value at the previous state inputs of the

circuit. The circuit nodes are then allowed to stabilize to their steady state values. The flip-flops, now configured in functional mode, load the next-state output of the circuit. Appropriate primary input values, along with the values loaded in the flip-flops, constitute v_2. Clearly, v_2 applied to the state inputs to the combinational logic is a function of v_1 and the next state logic of the function. In this mode, if the next state of the circuit for a given value of v_1, matches the value for v_2, then the fault is detected. It was reported in [CAB93] that most undetectable in sequential circuits were due to the incompatibility between the next state output for input v_1 and the desired v_2 value.

The other mode for test application also employs conventional scan design. It is called scan-shifting [CDK91a] or skewed-load [SB91, Sav92, PS92]. In this test application mode, v_1 is shifted into the flip-flops via the scan chain. After the desired initial state is obtained the v_2 values at the state inputs are obtained by shifting the contents of the scan chain (the flip-flops still configured in scan mode) by one bit. That is, the value of v_2 at the state input is one bit shift of the v_1 bits. In this mode, the order of flip-flops in the scan chain is crucial. Various methods have been proposed to order the flip-flops in the scan chain [CDK91a, MC90, MC91, SB91]. Note that, not all possible (v_1, v_2) combinations can be applied due to the correlation between the v_1 and v_2 bits. Hence, to obtain high fault coverage, it is often necessary to add extra (dummy) flip-flops to the scan chain to enable application of desired v_2. This selective enhancement of the scan chain is called partial enhanced scan in [CDK91a]. In [OKM92] clock suppression is used where the clock of each flip-flop can be controlled individually during test application. However the area overhead of this scheme is high. In [ESA93] a two clock grouping DFT methodology for testing *stuck-at faults* in sequential circuits has been reported. A partial scan methodology that replaces subset of circuit flip-flops by scan flip-flops with hold mode is recently reported in [CAB93]. A novel flip-flop design to enhance the capability of scan flip-flops to apply path delay tests is presented in [DS91].

A partial scan methodology that replaces subset of circuit flip-flops by scan flip-flops with hold mode is recently reported in [CAB93]. A novel flip-flop design to enhance the capability of scan flip-flops to apply path delay tests is presented in [DS91]. Test generators for path delay faults for sequential circuits have been recently reported in [CAB92, Che92, CDK91b, KBC+92, OKM92].

The paper is organized into six main sections. The following section introduces the DFT methodology. The test application modes considered are discussed in Section 3. The application of clock grouping to functional justification and scan shift test application modes is discussed in Sections 4 and 5, respectively. Experimental results are presented in Section 6. Finally, the conclusions and other other areas of application of the proposed technique are presented in the last section.

2 Proposed DFT Technique

In this paper, a new DFT technique for delay testing has been proposed that can improve delay fault coverage for sequential circuits without incurring high area overhead. The objective of DFT for delay testing is to enable the application of required two-pattern tests at the state inputs (i.e.

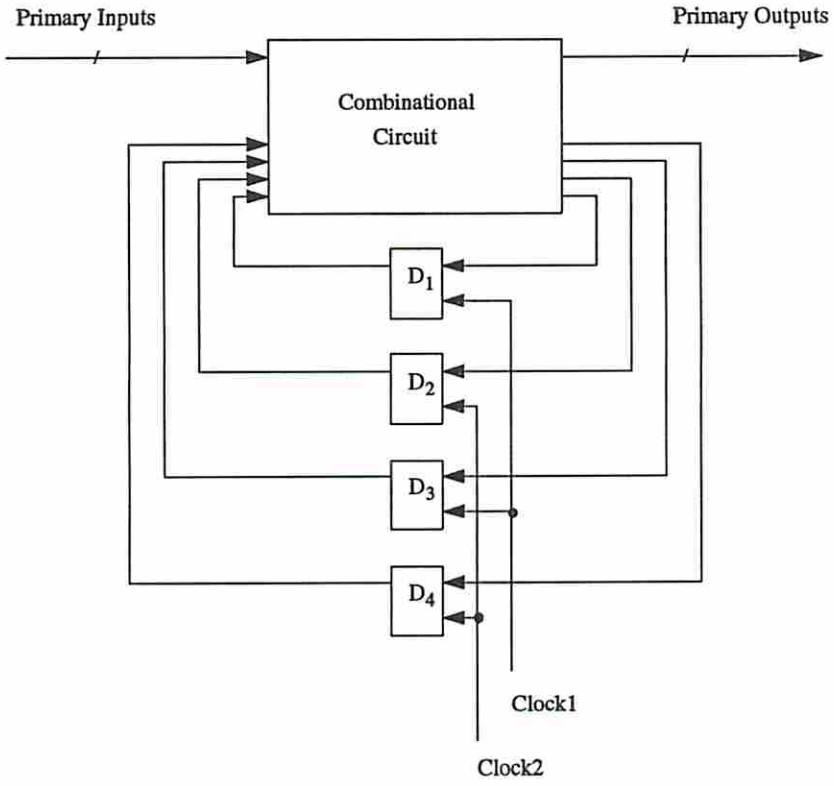


Figure 1: Circuit with Clock Grouping

the flip-flop outputs) of the circuit. Use of conventional scan is assumed in this paper. This enables the application of any desired v_1 value at the previous-state inputs of the circuit. However, the clock grouping DFT methodology can be applied to circuits without scan as well.

The main idea behind the proposed DFT technique is to enchance the scan chain's capability to apply two pattern tests by exercising a greater control on the flip flop clocks. The technique is called clock grouping. In normal mode, the flip-flops are clocked in their usual fashion. The flip-flops are partitioned into *clock groups*. In the test mode, the flip-flops in each clock group can be clocked independently. Consider the model of sequential circuit shown in Figure 1. $\{D_1, D_3\}$ and $\{D_2, D_4\}$ form two clock groups. In test mode the flip-flops in the first group are clocked by *clock1*, while those in the second group are clocked by *clock2*. Typically 2 or 3 flip-flop clock groups are considered since the overhead of this scheme grows with the number of groups.

Selective clocking of flip-flops between the application of v_1 and v_2 enables application of two-pattern tests to the CUT which could otherwise be applied only by using enhanced scan designs. Due to selective clocking of flip-flops between the application of v_1 and v_2, it is assumed that the flip-flops can retain their state for two clock cycles.

It shall be shown that the proposed clock grouping can be useful both under the functional justification as well as scan-shifting modes of two pattern test applications. Experimental results show significant improvement in fault coverage without the use of extra latches. A small number

of groups is typically sufficient to improve the testability of the circuit significantly.

The overhead of the clock grouping DFT methodology, over conventional scan design, is minimal. It is limited to small modifications in the clock distribution and extra pins for the test clock signals. Typically this can be accomplished in much lower overhead than the extensive use of extra flip-flops in the scan chain.

Though scan is assumed in the following, the idea of clock grouping is useful for testability enhancement, whether or not scan is employed. In non-scan designs, clock grouping can be used to achieve higher fault coverage not only due to its capability to apply a large number of v_2 vectors for any given v_1, but also by enabling the application of a larger set of initialization vectors as well. In fact, in high speed data path circuits (that have simple control structure), the application of clock grouping can provide a DFT methodology for comprehensive delay testing *with no performance penalty.*

3 Test Application Methodology

In the following, the circuit under test (CUT) is assumed to be synchronous sequential circuit with full scan. It is assumed that the flip-flops can hold their contents for at least two consecutive clock cycles if they are not clocked. It is further assumed that the scan chain is used to scan in any desired v_1. Since, any desired v_1 and v_2 values can be applied at the primary inputs, the issue is that of application of appropriate v_2 values at the state inputs. There are three different ways to obtain v_2.

1. *Hold:* The contents of the flip-flops can be held constant after the application of v_1, while the primary inputs are changed to apply appropriate v_2 values.
2. *Functional Justification:* In this mode, the response of the next state logic, to initialization vector v_1, is clocked into the flip flops and constitutes v_2 at the state inputs.
3. *Scan Shifting:* The contents of the scan-chain (v_1) are shifted by one bit and the resulting bits form v_2 at state inputs.

First it shall be shown in the following that the hold mode can be useful to test all testable path delay faults in paths originating at the primary inputs. Then in the following two sections, the functional justification and scan shifting shall be discussed from the point of view of application of the clock grouping DFT methodology to obtain high fault coverage.

Hold Mode This test application methodology is useful to reduce the number of paths that need to be considered by the DFT procedure. The hold test application mode can be shown to be sufficient to detect all robustly detectable path delay faults starting at primary inputs. This is a consequence of the following well known result.

Lemma 1 *[Smi85] If a delay fault in a path starting at the input x_i of a combinational circuit is robustly testable by a (v_1, v_2) pair, then there exists a test pair (w_1, w_2) where w_1 and w_2 differ only in input x_i.*

Corollary 1 *All robustly detectable delay faults in paths starting at primary inputs of the CUT are robustly tested by using the hold mode of test application.*

The advantages of using the hold mode are two-fold. Firstly, it reduces the number of faults that need to be considered by the DFT procedure reducing its computational complexity. Secondly, only the faults that need to targeted by scan chain ordering and clock grouping are considered, leading to better design for a given hardware overhead.

4 Functional Justification

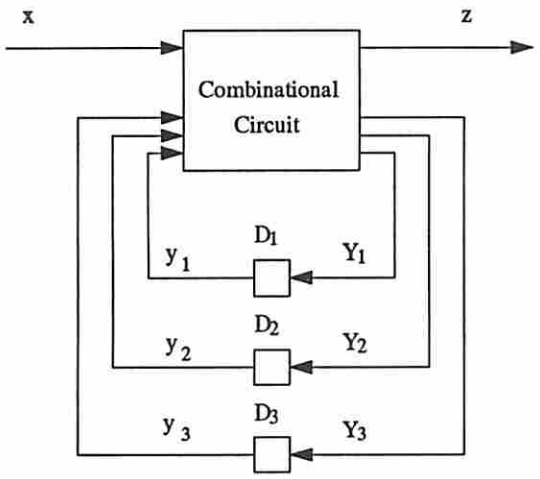
In this test application mode, the next state output of the CUT response to v_1 is latched into the flip-flops as the state bits of v_2. Note that, this test application mode does not require any special hardware beyond the conventional scan chain for shifting in the desired v_1. Further, this methodology does not impose any constraint on the scan chain order. Hence, the faults detected by this method should be identified and deleted from the fault list used in the following steps.

A fault is detected by the functional justification if the next state output obtained with v_1 as an input is compatible with the v_2 value needed at the state inputs. This may not be the case for many faults [CAB93]. For many faults some bits of v_2 at state inputs are compatible with the next state output of the circuit while other bits may not be. In such cases selectively clocking the flip-flops, between the application of v_1 and v_2, can help detect the fault. Note that, since only the paths originating at state inputs are being considered, the flip-flops in the clock group that contains the flip-flop connected to the path input must be clocked. However, the flip-flops in other clock group(s) may not be clocked if the v_2 value desired at their outputs is compatible with the v_1 values already present. The following example illustrates these concepts.

Example 1 *Consider a circuit with one input (x), one output (z) and three flip-flops. Let (y_1, y_2, y_3) be the previous state inputs to the combinational logic and let (Y_1, Y_2, Y_3) be the corresponding next state outputs (see Figure 2 (a)). The flip-flops D_1, D_2, D_3 are configured as a scan chain (not shown in figure) to shift any desired v_1 into the flip-flops.*

Consider path delay fault along a path p originating at y_1. The v_1 and v_2 vectors required for robust testing of this fault are shown in Figure 2 (b). Also, the steady state value of the next state output corresponding to v_1 is shown in column marked $Y = f(v_1)$. Note that this two pattern test cannot be applied to the CUT by using functional justification since functional justification can apply the desired value at y_1 but will apply an incorrect value at y_2.

Now consider a case where, in the test mode, D_1 and D_3 are clocked together while D_2 is controlled by a different clock. In such a case the flip-flips D_1 and D_3 can be clocked without clocking



(a) Example Circuit

	v_1	v_2	$Y=f(v_1)$
x	0	0	--
D_1	1	0	0
D_2	0	0	1
D_3	1	1	1

(b) Test Vectors

Figure 2: Example for Partial Parallel Load

D_2. This leads to the application of desired two pattern test. The flip flops are said to be partitioned into two clock groups: $\{D_1, D_3\}$ and $\{D_2\}$. Each group of flip-flops can be clocked independently in the test mode.

The above example illustrates the advantage of selectively clocking flip-flops to apply (v_1,v_2) tests where the state inputs in v_2 are not compatible with the next state output for v_1. In the limit, each flip-flop clock may be individually controlled for test application [OKM92]. However the area overhead for such a design would be high. Hence, for practical reasons, the flip-flops in the circuit are assumed to be grouped into k groups where k=2 or 3. In the following discussion, $\{D_1, D_2, \ldots, D_l\}$ is the set of circuit flip-flops. $\{y_1, y_2, \ldots, y_l\}$ are the (previous) state inputs to the combinational circuit corresponding to the outputs of these flip-flops. Finally, $\{Y_1, Y_2, \ldots, Y_l\}$ are the next state outputs of the combination circuit that are the functional inputs to the flip-flops. The clock grouping problem can now be stated.

Problem Statement Given two pattern robust path delay test set T for faults in the circuit. Determine a partition of circuit flip-flops $\{D_1, D_2, \ldots, D_l\}$ into disjoint partitions $(\pi_1, \pi_2, \ldots, \pi_k)$, such that all flip-flops in each partition are controlled by a separate clock in the test mode. The objective of this partitioning is to maximize the number of path delay faults that can be tested by applying a suitable clock/no-clock signal to each group.

There are 2^k different clock/no-clock combinations possible under k clock groups. Of this, the all no-clock corresponds to the hold mode and can be used only to detect delay faults in the paths starting at primary inputs. Also, the all clock case corresponds to the conventional *full functional justification* mode. Hence, grouping flip-flops into k groups allows $2^k - 2$ new test

application configurations. Test application using these new configurations shall be called *partial functional justification*. Note that the faults detectable in hold mode or full functional justification are independent of the clock grouping. Hence the faults detected in these modes should be deleted from the fault list to be used during the clock grouping procedure.

4.1 Clock Grouping Procedure

Consider the vectors shown in Figure 2 (b) again. The application of the desired two-pattern test requires that flip-flop D_1, must be clocked, while flip-flop D_2 must not be clocked. This indicates that for the application of this two pattern test, D_1 and D_2 must belong to different clock groups while D_3 can be grouped with either flip-flop. Let, v_{1i} (v_{2i}) indicate the i-th bit of v_1 (v_2). Note that v_{1i} is the input that is applied to the state input y_i as a part of v_1. Similarly, v_{2i} is the input to y_i as a part of v_2. Let $Y_i(v_1)$ be the value implied at the next state output Y_i by vector v_1.

Definition 1 *A two pattern test is said to be applicable under functional justification by some clock grouping if for each state input y_i ($i = 1, 2, \ldots, l$), either $v_{2i} = v_{1i}$ or $v_{2i} = Y_i(v_1)$.*

Note that a two pattern test that does not satisfy the above conditions can not be applied to the circuit, by full/partial functional justification, no matter how the clocks are grouped.

Definition 2 *Two flip-flops D_i and D_j are said to compatible under functional justification for a given two pattern test (v_1,v_2), if* **either;** $v_{2i} = v_{1i}$ **and** $v_{2j} = v_{1j}$; **or** $v_{2i} = Y_i(v_1)$ **and** $v_{2j} = Y_j(v_1)$.

The compatible flip-flops may be assigned to the same group and still permit the application of the given two pattern test. Two flip-flops D_i and D_j are said to be incompatible for a given two pattern test (v_1, v_2) if they do not satisfy either condition in the above definition. If incompatible flip-flops are assigned to the same group, the given two-pattern test cannot be applied by using any combination of clock/no-clock before the application of v_2.

Definition 3 *Two flip-flops D_i and D_j are said to be compatible under functional justification for a given fault f, with respect to a two pattern test set T, if there exits a test for f in T for which they are compatible under functional justification.*

If two flip-flops D_i and D_j are not compatible for any two pattern tests for a given path delay fault f, then D_i and D_j are said to be *essentially incompatible* with respect to the given delay fault f.

Now, we shall present the clock grouping procedure employed to enhance the coverage of path delay faults. Only the paths from the state inputs are considered in the following analysis since all paths from primary inputs can be tested by the use of hold mode. The fault list F consists of all delay faults in paths starting at state inputs (SI paths).

1. Robust delay tests are generated for all faults in F. If the test can be applied by full functional justification, then the faults detected by it are dropped from the fault list F. Otherwise, the two pattern test is added to the set of test vectors T.

2. For each path delay fault in the fault list F, essential incompatibility is computed for all flip-flop pairs (D_i, D_j) with respect to the test set T.

3. The information is stored in an $l \times l$ essential incompatibility matrix (EIM) where EIM(i, j) contains the number of faults that cannot be detected under functional justification (only considering two pattern tests in T), if D_i and D_j are grouped together.

4. A greedy heuristic is used to partition flip-flops into k groups. After every decision, two pattern tests that cannot be applied by the partial grouping are eliminated from T and EIM is recomputed.

5. Finally, the flip-flops that need not be assigned to any group are assigned to groups to equalize the number of flip-flops in various groups.

4.2 Experimental Results

The impact of clock grouping procedure on fault coverage obtainable by using functional justification was studied using ISCAS 89 sequential benchmark circuits. Firstly, it was assumed that all detectable delay faults in the paths originating at primary inputs were detected by applying tests using hold mode. Hence, in the following only the paths starting at the state inputs (called SI paths in the tables) are considered.

An automatic test pattern generator (ATPG) has been developed to generate tests under different test application modes. This ATPG was used to generate two pattern robust tests under full functional justification. The number of paths detected under this test application mode are shown in Table 1, under the column *1 clock*.

Subsequently, the ATPG was used to generate robust tests for the delay faults in the remaining SI paths. The ATPG ensured that test vectors that were not applicable under *any* clocking scheme were eliminated from consideration. The resulting test set T was used to partition the circuit flip-flops into 2 and 3 clock groups.

The number of additional path delay faults detected by using two/three clock groups are shown in Table 1 under the columns marked *2 clocks/3 clocks*, respectively. Note that for many circuits, the use of two clock groups more than doubled the number of path delay faults detectable under functional justification mode. In the remaining circuits, the improvement was substantial. If three clock groups were used, the fault coverage was even higher. However, in this case the increase over the two clock group case was smaller.

The results indicate that clock grouping can significantly improve the fault coverage obtainable using partial functional justification mode of test application. As shall be seen in the following,

Table 1: Fault Coverage with Clock Grouping for Functional Justification

Circuit name	*# PI*	*# F.F*	*Coverage of SI PDF with F.J.*		
			1 clock	*2 clock*	*3 clock*
s27	4	3	4	5	5
s208	11	8	57	79	109
s298	3	14	102	173	219
s344	9	15	213	316	347
s349	9	15	213	316	346
s400	3	21	115	251	312
s420	19	16	79	167	249
s444	3	21	114	231	269
s510	19	6	131	258	345
s641	35	19	232	328	361
s820	18	5	177	291	384
s953	16	29	375	601	661
s1196	14	18	204	230	237
s1238	14	18	207	228	238
s1488	8	6	330	791	1028
s1494	8	6	334	792	1030

clock grouping can improve the fault coverage obtainable by using scan-shift as well. Comprehensive experimental data shall be presented at the end of the paper.

5 Scan Shift with Clock Grouping

Clock grouping can have a tremendous impact on the fault coverage for a small overhead. In this section, it shall be shown that clock grouping is very useful even with scan-shift mode of test application. In scan shift mode, scan chain is used to scan in v_1 into the flip-flops. After the combinational circuit settles down at its steady state value under input v_1, the contents of the scan chain are shifted by one bit, providing the state inputs for v_2. Simultaneously, appropriate values are applied at primary inputs.

For any given v_1, only a subset of all possible v_2 values can be applied at the state inputs of the CUT (connected to the outputs of the flip-flops in the scan-chain). In fact if v_1 (for state inputs) is fully specified, then only two different v_2 values can be applied following the given v_1. Even these two v_2 values differ only in the state input connected to the flip-flop output that is the closest to the scan-in pin. This correlation between v_1 and v_2 at the state inputs is a major cause for obtaining insufficient fault coverage. However, most path delay faults do not require all state input bits to be specified in their test vectors. This is due to the fact that logic in each output may typically depend only on a subset of inputs. Even when this is not strictly true, the effect of correlation between some

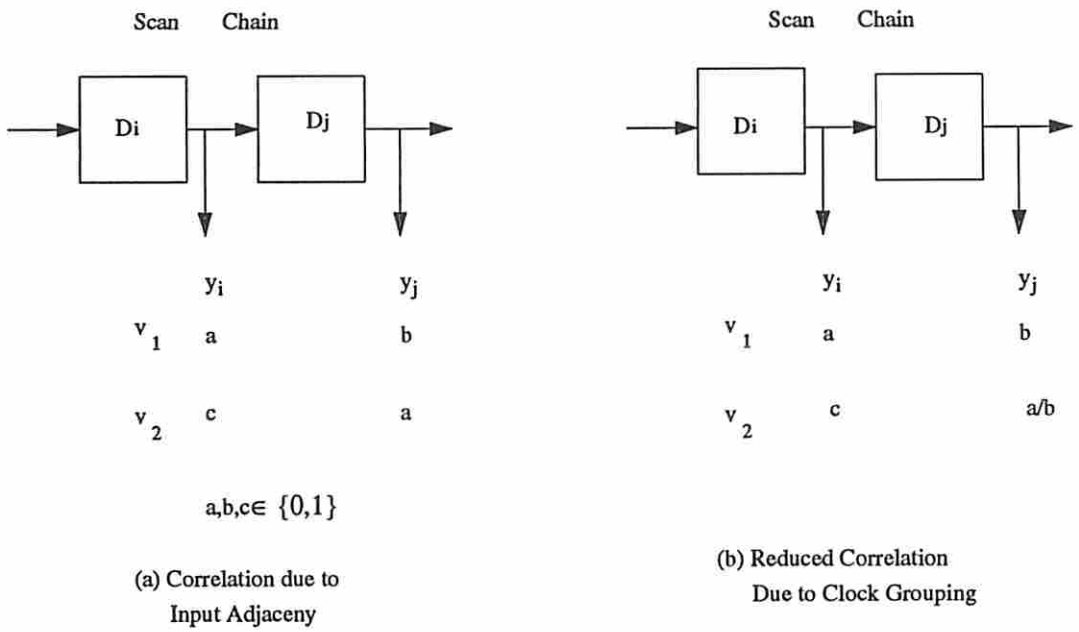


Figure 3: Reduced correlation due to clock grouping

state inputs may be less detrimental as opposed to others. Many techniques have been proposed in the literature for ordering flip-flops in the scan chain to improve fault coverage. The methods based on empirical measure of correlation [MC91], structure of each output cone [SB91], and on a test set derived for delay faults of interest [CDK91a] have been presented. Note that in these techniques extra flip-flops is typically required to obtain high robust path delay fault coverage.

Our objective to study the impact of clock grouping on the fault coverage obtainable by the scan shifting test application. As has been mentioned in [SB91] that if two inputs do not fan out to any output simultaneously, then the corresponding flip-flops may be assigned consecutive positions in the scan chain without impacting the fault coverage. However, inputs that fan out to common outputs should not be assigned to consecutive stages in the scan chain. Since it may not be possible to meet this requirement without adding extra flip-flops, it is necessary to consider another way to reduce the adverse effect of correlation due to scan-shift. The discussion that follows shows how clock grouping can help reduce the correlation.

5.1 Scan Chain Correlation and Clock Grouping

Consider state inputs y_i and y_j connected to consecutive flip-flops on the scan chain. Assume for the simplicity of the following discussion that the values for both y_i and y_j in both v_1 and v_2 are fully specified. Figure 3 (a) illustrates the nature of the correlation. Only, 8 out of 16 possible fully specified values can be applied at y_i and y_j in this configuration. Only those (v_1, v_2) values can be applied that have $v_{1i} = v_{2j}$. It is assumed here that the flip-flop that precedes D_i can provide any desired input for the v_{2i} (bit applied to y_i).

Now, assume that the flip-flops are assigned to at least two different clock groups. Consider the case that flip-flops D_i and D_j are clocked using different clocks. If the groups containing D_i and D_j are clocked simultaneously, then all 8 different two pattern combinations can be applied to y_i and y_j as shown before. However, if D_i is clocked while D_j is not, it is possible for y_j to retain its old value. Hence, by this scheme, 12 (out-of 16) two pattern combinations can be applied to the inputs y_i and y_j. Now two pattern tests that have $v_{2j} = v_{1j}$ can be applied, even when $v_{2j} \neq v_{1i}$. (Note that it is again assumed that the flip-flop that precedes D_i in the scan chain can provide the required bit value for v_2 for y_i.) This reduced correlation due to clock grouping can drastically improve pattern pair coverage at the inputs that belong to same outputs but the corresponding flip-flops must be assigned consecutive stages in the scan chain. It can be easily shown that the test patterns that can be applied to the circuit inputs under multiple clock groups are a super-set of the patterns that can be applied using full scan-shift. Given k-clock groups, 2^k different clock/no-clock combinations can be used. If none of the clock groups are clocked between the application of v_1 and v_2, then it corresponds to the hold mode. If all the groups are clocked, then we get the conventional *full scan shift* case. The remaining $2^k - 2$ cases are called *partial scan shift.*

5.2 Design Procedure

At this stage, the order of flip-flops in the scan chain and clock grouping may need to be determined. In either case, the delay faults in paths originating at primary inputs and the delay faults in the paths starting at state inputs that are detected by full functional justification can be eliminated from consideration. Two main scenarios are considered.

Clock Grouping for Partial Functional Justification The full functional justification can be followed by clock grouping to maximize extra fault coverage obtained by partial functional justification. The faults detected by partial functional justification are dropped from the fault list. ATPG is used to generate tests for the faults remaining in the fault list. A scan chain ordering procedure has been developed that takes into account the difference between the set of possible two pattern values that can be applied to the successive flip-flop outputs depending on whether the two flip-flops belong to the same or different clock groups. A greedy heuristic is used to determine chain order to maximize the number of path delay faults that can be detected by using full and partial scan-shift modes.

Clock Grouping for Partial Scan Shift In this case both clock grouping and scan chain ordering need to be performed. The fault list contains all SI paths that are not detected by full functional testing. ATPG is used to generate two pattern tests for the faults in the fault list. Only those patterns that can be applied using scan-shifting are retained. Then a procedure similar to the one used in [CDK91a] is used to determine the order of flip-flops in the scan chain. The faults detected by full scan-shift are eliminated from the fault list. The clock grouping is then performed

using the remaining test vectors to maximize the additional fault coverage obtained by partial scan-shift (the procedure is similar to the one outlined for partial functional justification). Finally, the ATPG uses the given clock grouping to generate partial functional justification tests to determine the overall fault coverage. The experimental results for both these cases are presented next.

6 Experimental Results

In this section, the experimental results shall be presented for ISCAS 89 benchmark circuits for the two cases described above. In both the cases, only the delay faults in paths originating at state inputs are considered. Tests are applied under full and partial, functional justification as well as scan shifting. However, the two cases differ in the way the clock grouping and scan chain ordering are determined. In the first case, the scan chain ordering is determined mainly from considerations of partial functional justification. This is followed by scan chain ordering. In the second case, the scan chain ordering is performed first. The clock grouping is then determined mainly to maximize the coverage of remaining delay faults via partial scan shifting.

First consider the case without the application of clock grouping DFT methodology. Without clock grouping only full functional justification and full scan shift can be used (in addition to the hold test application mode). After ordering the scan chain, the total fault coverage obtained for various benchmark circuits is as shown in Table 2. Fault coverage is very low for many circuits. (All results presented here are for the SI paths only. Further, the fault coverage is computed as a percentage of all faults for which ATPG found a two pattern test.) In the following, fault coverage obtained by using clock grouping shall be presented.

Clock Grouping for Partial Functional Justification In this case the clock groups are determined to maximize the coverage obtained by partial functional justification. The scan chain order is then determined to maximize the coverage of remaining faults via full and partial scan shifting. Tables 3 and 4 present the fault coverage information for two and three clock groups, respectively. The use of clock grouping (even two clock groups) provides a significant increase in fault coverage over the case without clock grouping (Table 2). For many circuits (s298, s344, s349, s444, etc.) the fault coverage is very high for two clock case. Further, circuits such as s510, s1488, and s1494, that had extremely low fault coverage without clock grouping, had significant improvements. Note that these circuits otherwise require *one extra flip-flop for each circuit flip-flop* to obtain high fault coverage (e.g. see data in [CDK91a]). The fault coverage increased further when three clock groups are used. However, the increase in fault coverage over two clock case is lower. The overall fault coverage is very high for most circuits.

Clock Grouping for Partial Scan Shift Performing chain ordering before clock grouping allows one to explore the possibility of performing clock grouping to maximize the coverage of remaining faults by partial scan shifting. This clock grouping can then be used to eliminate any faults detected

Table 2: Fault Coverage (of SI paths) without Clock Grouping

Circuit name	*Fault coverage*		*Total fault coverage*	
	F.J.	*S.Shift*	*# of path detected*	*%*
s27	4	16	20	90.91
s208	57	73	130	61.32
s298	102	132	234	73.82
s344	213	247	460	81.41
s349	213	247	460	81.41
s400	115	332	447	76.8
s420	79	352	431	71.83
s444	114	261	375	74.55
s510	131	135	266	39.76
s641	232	187	419	84.14
s820	177	142	319	48.41
s953	375	1054	1429	99.86
s1196	204	80	284	100
s1238	207	79	286	100
s1488	330	230	560	36.27
s1494	334	225	559	36.06

Table 3: Fault Coverage with 2 Clock Groups - Clock Grouping for Partial F.J.

Circuit name	*Fault coverage*				*Total fault coverage*	
	F.J.	*Partial F.J.*	*S.Shift*	*Partial S.Shift*	*# of path detected*	*%*
s27	4	1	15	2	22	100
s208	57	22	54	26	159	75
s298	102	71	106	24	303	95.58
s344	213	103	135	59	510	90.27
s349	213	103	135	59	510	90.27
s400	115	136	107	67	425	73.02
s420	79	88	255	39	461	76.83
s444	114	117	183	70	484	96.22
s510	131	127	83	98	439	65.62
s641	232	96	131	15	474	95.18
s820	177	114	113	81	485	73.6
s953	375	226	830	0	1431	100
s1196	204	26	54	0	284	100
s1238	207	21	58	0	286	100
s1488	330	461	134	165	1090	70.6
s1494	334	458	134	163	1089	70.29

Table 4: Fault Coverage with 3 Clock Groups - Clock Grouping for Partial F.J.

Circuit name	*Fault coverage*				*Total fault coverage*	
	F.J.	*Partial F.J.*	*S.Shift*	*Partial S.Shift*	*# of path detected*	*%*
s27	4	1	15	2	22	100
s208	57	52	32	38	179	84.43
s298	102	117	52	28	299	94.32
s344	213	134	135	46	528	93.45
s349	213	133	136	46	528	93.45
s400	115	197	128	102	542	93.13
s420	79	170	205	51	505	84.17
s444	114	155	123	96	488	97.02
s510	131	214	83	111	539	80.57
s641	232	129	111	17	489	98.19
s820	177	207	56	98	538	81.64
s953	375	286	770	0	1431	100
s1196	204	33	47	0	284	100
s1238	207	31	48	0	286	100
s1488	330	698	94	180	1302	84.33
s1494	334	696	95	181	1306	84.26

by partial parallel load. The clock grouping provides significant increase in fault coverage over the single clock case even with two clock case Table 5. For some circuits, higher fault coverage was obtained over the previous case. Use of three clock group further increases the fault coverage as well (Table 6).

The results obtained by the application of the proposed clock grouping technique show that significant improvements can be obtained in fault coverage by clock grouping. In most cases, grouping circuit flip-flops into just two or three groups can provide high fault coverage. Also, it was found that many circuits that would otherwise require doubling the number of flip-flops in the circuit to obtain high fault coverage, can be tested very comprehensively by the application of clock grouping technique. It was also observed that in most cases, higher fault coverage was obtained by performing clock grouping for partial functional justification than for partial scan shift. An algorithm that can perform scan chain ordering and clock grouping for partial scan shift simultaneously is currently being developed.

7 Conclusion

A low overhead DFT technique, called clock-grouping, for delay testing of sequential synchronous circuits is presented. The proposed technique increases robust path delay fault coverage for circuits by exercising greater control over flip-flop clocks in the test mode. In the test mode, the flip-flops are partitioned into different clock-groups. The flip-flops in each clock group can be either clocked

Table 5: Fault Coverage with 2 Clock Groups - Clock Grouping for Partial Scan Shift

Circuit name	*Fault coverage*				*Total fault coverage*	
	F.J.	*S.Shift*	*Partial S.Shift*	*Partial F.J.*	*# of path detected*	*%*
s27	4	16	2	0	22	100
s208	57	73	24	2	156	73.58
s298	102	132	23	17	274	86.44
s344	213	247	55	4	519	91.86
s349	213	247	55	4	519	91.86
s400	115	332	57	7	511	87.80
s420	79	352	36	14	481	80.17
s444	114	261	42	17	434	86.28
s510	131	135	95	53	414	61.88
s641	232	187	39	12	470	94.38
s820	177	142	80	58	457	69.35
s953	375	1054	0	0	1429	99.86
s1196	204	80	0	0	284	100
s1238	207	79	0	0	286	100
s1488	330	230	234	231	1025	66.39
s1494	334	225	237	233	1029	66.39

Table 6: Fault Coverage with 3 Clock Groups - Clock Grouping for Partial Scan Shift

Circuit name	*Fault coverage*				*Total fault coverage*	
	F.J.	*S.Shift*	*Partial S.Shift*	*Partial F.J.*	*# of path detected*	*%*
s27	4	16	2	0	22	100
s208	57	73	43	5	178	83.96
s298	102	132	44	19	297	93.69
s344	213	247	66	10	536	94.87
s349	213	247	66	10	536	94.89
s400	115	332	87	6	540	92.78
s420	79	352	71	23	525	87.50
s444	114	261	76	11	462	91.85
s510	131	135	158	71	495	73.99
s641	232	187	54	9	482	96.79
s820	177	142	140	76	536	81.18
s953	375	1054	0	2	1431	100
s1196	204	80	0	0	284	100
s1238	207	79	0	0	286	100
s1488	330	230	400	314	1274	82.25
s1494	334	225	404	315	1278	82.45

or not clocked, independent of the flip-flops in the other groups. This flexibility is used to enhance the number of different (v_1, v_2) test pairs that can be applied to the state inputs of the circuit thereby increasing coverage of delay faults. Experimental data on benchmark circuits shows that high fault coverage can be obtained by using only two clock groups in most circuits.

Clock grouping can be employed as a DFT technique even if the circuit does not use scan design. In such circuits, clock grouping can aid in four different ways to enhance delay fault coverage. Firstly, starting at an initial state an input sequence needs to be derived to justify the state required by v_1 (e.g. see [CAB92, GDN91]). Clock groups can be useful in this stage, since they can be used to obtain state transitions that are not possible if all the flip-flops are clocked in every cycle. This can increase the number of different v_1 vectors that can be applied. Having applied the v_1 vector, a larger set of v_2 vectors can applied as outlined above for the partial functional justification case. Note that a significant improvement in fault coverage is shown in Table 1. Thirdly, after the application of v_2, the circuit response needs to be observed at the circuit's normal operating clock rate. If the path under test terminates at a state output, then an error is captured in the corresponding flip-flop if the fault is present. However, some other flip-flops may or may not have erroneous values. This uncertainty (that does not exist in stuck-at testing), can lead to long sequences to propagate the fault effect to a primary output. In fact, in many cases, it may not be possible to find a sequence that can find a sequence to propagate the fault effect to a primary output in presence of such uncertainty. Techniques to ensure robust testing in the presence of uncertainty in the values captured at the flip-flops are bound to be restrictive [CAB92]. However, the clock groups can be used to selectively capture the response into the flip-flop at the output of the path under test (and all the flip flops that belong to the same group) while the other flip-flops retain their previous (known) values. This reduction in uncertainty can make it easier to propagate the fault effect to an output. Finally, the clock groups can aid, by allowing transitions not possible in the normal mode, to find sequence of inputs to propagate the fault effect to a primary output. The application of the clock grouping is being studied under this framework. The main advantage of clock grouping in this context is that in high speed data paths that employ simple control strategy, the performance penalty of using DFT can be minimized. This is due to the fact that clock grouping does not add any delay in series with the high speed data path.

References

[CAB92] T. J. Chakraborty, V. D. Agrawal, and M. L. Bushnell. Delay Fault Models and Test Generation for Random Logic Sequential Circuits. In *Proceedings IEEE-ACM Design Automation Conference*, pages 165–172, 1992.

[CAB93] T. J. Chakraborty, V. D. Agrawal, and M. L. Bushnell. Design for Testability for Path Delay Faults in Sequential Circuits. In *Proceedings IEEE-ACM Design Automation Conference*, pages 453–457, 1993.

[CDK91a] K-T Cheng, S. Devadas, and K. Keutzer. A Partial Enhanced-Scan Approach to Robust Delay-Fault Generation for Sequential Circuits. In *Proceedings IEEE International Test Conference*, pages 403–410, 1991.

[CDK91b] K-T Cheng, S. Devadas, and K. Keutzer. Robust Delay-Fault Test Generation and Synthesis for Testability under a Standard Scan Design Methodology. In *Proceedings IEEE-ACM Design Automation Conference*, pages 80–86, 1991.

[Che92] Kwang-Ting Cheng. Test Generation for Delay Faults in Non-scan and Partial Scan Circuits. In *Proceedings IEEE International Conference on Computer-Aided Design*, pages 554–559, 1992.

[DS91] B. I. Dervisoglu and G. E. Stong. Design for Testability: Using Scanpath Techniques for Path-Delay Test and Measurement. In *Proceedings IEEE International Test Conference*, pages 365–374, 1991.

[ESA93] K. L. Einspahr, S. G. Seth, and V. D. Agrawal. Clock Partitioning for Testability. In *Proceedings of Third Great Lakes Symposium on VLSI*, pages 42–46, 1993.

[GDN91] A. Ghosh, S. Devadas, and R. Newton. Test Generation and Verification for Highly Sequential Circuits. *IEEE Trans. on CAD*, 10(5):652–667, May 1991.

[KBC+92] B. Konemann, J. Barlow, P. Chang, B. Gabrielson, C. Goertz, B. Keller, K. McCauley, J. Tischer, V. Iyengar, B. Rosen, and T. Williams. Delay Test: The Next Frontier for LSSD Test Systems. In *Proceedings IEEE International Test Conference*, pages 578–587, 1992.

[LR87] J. C. Lin and S. Reddy. On Delay Fault Testing in Logic Circuits. *IEEE Trans. on CAD*, 6(5):694–703, September 1987.

[Mal83] Y. K. Malaiya. Testing for Timing Faults in Synchronous Sequential Integrated Circuits. In *Proceedings IEEE International Test Conference*, pages 560–571, 1983.

[MC90] W. Mao and M. D. Ciletti. Arrangement of Latches in Scan-path Design to Improve Delay Fault Coverage. In *Proceedings IEEE International Test Conference*, pages 387–393, 1990.

[MC91] W. Mao and M. D. Ciletti. Correlation Reduced Scan-path Design to Improve Delay Fault Coverage. In *Proceedings IEEE-ACM Design Automation Conference*, pages 73–79, 1991.

[OKM92] B. Oomman, P. Kongara, and C. Mallipeddi. Amdahl Corporation Board Delay Test System. In *Proceedings IEEE International Test Conference*, pages 558–567, 1992.

[Pra91] A. K. Pramanick. Delay Testing and Design for Testability for Delay Faults in Combinational Logic Circuits. Technical report, Ph.D. Dissertation, University of Iowa, 1991.

[PS92] S. Patil and J. Savir. Skew-Load Transition: Part II, Coverage. In *Proceedings IEEE International Test Conference*, pages 714–722, 1992.

[Sav92] J. Savir. Skew-Load Transition: Part I, Calculus. In *Proceedings IEEE International Test Conference*, pages 705–713, 1992.

[SB91] Jacob Savir and Robert Berry. At-Speed Test Is Not Necessarily an AC Test. In *Proceedings IEEE International Test Conference*, pages 722–728, 1991.

[Smi85] Gordon L. Smith. Model for Delay Faults Based on Paths. In *Proceedings IEEE International Test Conference*, pages 342–349, 1985.