On Improving Fault Diagnosis for Synchronous Sequential Circuits

Irith Pomeranz and Sudhakar M. Reddy [+]
Electrical and Computer Engineering Department
University of Iowa
Iowa City, IA 52242

Abstract

The multiple observation times approach was proposed as a test generation approach for fault detection, and was shown to alleviate deficiencies of conventional test generators. In this work, the multiple observation times approach is applied to fault location. It is shown that the use of multiple observation times has the potential of significantly enhancing the resolution of a given test set. A definition of pass/fail diagnosis suitable for the multiple observation times approach is also given. Experimental results are provided to demonstrate the approach and its advantages.

1. Introduction

Fault location, or diagnosis, is performed when a circuit-under-test is known to be faulty. It is performed in order to locate where in the circuit a fault has occurred. Fault location may be required to identify and then replace the faulty subcircuit, or to analyze the defect causing the faulty behavior. The problem of fault location has been considered extensively (e.g., [1-11]). A broad class of fault location procedures rely on comparison of the circuit response with a set of responses produced by the circuit-under-test in the presence of modeled faults. The observed response is attributed to one or more modeled faults, which are used to identify the fault site. We assume the same approach in this work. The quality, or *resolution* [8,10] of the test set applied to the circuit for the purpose of fault location is determined by the number of modeled faults that can potentially explain a given response of the circuit-under-test. The smaller this number, the easier it is to identify the fault site accurately, and hence the resolution of the test set is higher. Resolution is computed based on the subsets of modeled faults that cannot be distinguished by the given test set. When considering synchronous sequential circuits, three-value simulation may result in unspecified output values. In this case, two faults are said to be *indistinguishable* if they have compatible responses, where compatibility is defined as follows. Two responses are said to be *compatible* if they are identical except possibly where one of the responses is unspecified. The specific resolution measures used in this work are described in Section 3.

The reason for using unspecified values when considering synchronous sequential circuits is that they allow to compactly represent the uncertainty as to the initial state of the circuit, and as a result, allow a compact representation of its response. In the context of test generation for fault detection, unspecified values allow a single output sequence to represent all possible responses of a circuit, independent of its initial state. A single response of the fault free circuit can thus be compared to the response of the circuit-under-test, to determine whether it is fault free or faulty. However, it was shown in [12] that the use of unspecified values, and in particular, the use of a fully unspecified initial state, also causes significant loss of fault coverage in some cases. It was also shown that the loss in fault coverage can be corrected by considering several initial states (fully or partially specified), each one separately. The number of initial states that need to be considered separately can be kept low, by using an appropriate test generation method [13]. The resulting test generation approach, called the multiple observation times approach, resulted in a test generation procedure [13], that yields significantly improved fault coverage in benchmark circuits. For example, for ISCAS-89 circuit $s510$, conventional test generation procedures achieve zero fault coverage, whereas the multiple observation times procedure from [13] results in 100% fault coverage. In this work, we show that similar improvements can be obtained with respect to the resolution of a given test set for fault location. In particular, we show that by using incompletely specified output responses, two faults may be indistinguishable, even though they never produce the same output response. Such faults are distinguishable when the multiple observation times approach is used. Moreover, we show that even when the fault coverages (i.e., the percentage of faults detected by a given test sequence) under the conventional approach and under the multiple observation times approach are similar, there can be a significant difference in the fault location capabilities of a given test sequence under these two approaches. Thus, the multiple observation times approach can be used for increased fault location resolution even when it does not result in a significant fault coverage improvement.

This paper presents a study of the effects of the multiple observation times approach on fault location. It is important to note that for some circuits, where the use of unspecified values does not result in loss of accuracy, the multiple observation times approach is not expected to yield significant improvements. The study in this work is aimed at circuits where such loss of accuracy is significant. The paper is organized as follows. In Section 2 we discuss fault diagnosis under the multiple observation times approach, and show its advantages. In Section 3 we present fault diagnosis based on pass/fail responses. In Section 4 we present experimental results. Section 5 concludes the paper.

2. Diagnosis using multiple observation times

In this section, we show the advantages of multiple observation times over conventional approaches through several examples, and then define the measure of resolution we use.

2.1 Comparison with single observation time diagnosis

Consider the three faulty responses of a two-output circuit shown in Table 1. If the circuit-under-test produces the response (00,00,01), this response can be uniquely attributed to f_1. However, if the circuit-under-test produces the response (00,01,10), this response can be attributed to either f_2 or f_3, and no unique identification of the fault is possible. We represent the information regarding the faults that can produce each output response of the circuit in a fault location tree [15]. The tree is constructed to include for every output response, the fault or faults that can

[+] Research supported in part by NSF Grant No. MIP-9220549, and in part by NSF Grant No. MIP-9357581

explain it. At the same time, the tree should be kept as small as possible, therefore, we represent the possible output responses using as many unspecified values as possible. A fault location tree based on the responses of Table 1 is shown in Figure 1, and its construction is explained next. In the root, we place all three faults. The possible responses to the first input symbol of the test sequence are 22 (for f_1 or f_2) and 20 (for f_3). The two responses have a non-empty intersection, 20. In addition, 22 contains a response, namely 21, that is not contained in 20. To distinguish between 20 and 21 in the tree, the root has two sons, one corresponding to 20 and one corresponding to 21. The son corresponding to 21 contains only f_1 and f_2. In general, whenever output values with non-empty intersection exist, they are intersected until disjoint responses are obtained. A vertex in the tree becomes terminal (or a leaf) when no additional output values exist, or when it contains a single fault. From the tree of Figure 1, we can see that responses like (00,00,01) uniquely identify the fault. Responses like (00,00,00) identify a pair of faults, any one of which can explain the circuit behavior.

Table 1: Fault location - single observation

	time		
fault	1	2	3
f_1	22	22	02
f_2	22	21	10
f_3	20	02	20

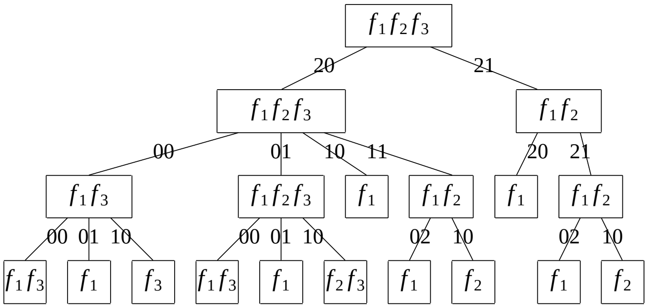

Figure 1: A tree based on Table 1

Suppose now that the circuit of Table 1 has two states, and that fault simulation under the multiple observation times approach yields the responses of Table 2. The first response of every fault is produced by the faulty circuit when it starts from its first state, and the second response is produced when the circuit starts from its second state. Comparison of Tables 1 and 2 shows the main advantage of the multiple observation times approach. Although there are only two possible responses in the presence of f_1, namely, s_1 = (00,01,00) and s_2 = (11,10,01), under conventional fault simulation, the two responses are compactly represented as (22,22,02). A large number of responses that cannot be produced by the circuit are included in this response. For example, (01,00,00) is included in (22,22,02) but not in the set of possible responses of the circuit. There is a total of $2^5 - 2 = 30$ such responses. Moreover, some responses that cannot be obtained can be attributed to two different faults, causing them to appear to be indistinguishable. For example, in Table 1, the response (00,00,00) is contained in the responses of both f_1 and f_3, making them indistinguishable, however, this response is never produced by either f_1 or f_3 according to Table 2. Similarly, (00,01,00) is contained in the responses of both f_1 and f_3 according to Table 1, however, it is produced only by f_1. This significantly reduces the resolution of fault location under the conventional approach, below what can actually be achieved if the responses of modeled faults are computed under the multiple observation times approach. In the example, when multiple observation times are used and the responses of Table 2 are obtained, each one of the responses can be uniquely attributed to a single fault.

A fault location tree for Table 2, similar to the one of Figure 1, is shown in Figure 2. Note that in the construction of the tree, we consider each one of the sequences of every fault separately. For example, although f_2 has output values 01 and 11 in time unit 2, only 11 follows 00 at time unit 1, therefore, it is the only response of f_2 in the leftmost part of the tree, corresponding to the sequence s_4. We give the sequence numbers of Table 2 below the fault numbers in the vertices of the tree. In accordance with our observation that every response is unique to one fault, the leaves of the tree contain single faults and single sequences.

Table 2: Fault location - multiple observations

		time		
fault	sequence	1	2	3
f_1	s_1	00	01	00
	s_2	11	10	01
f_2	s_3	11	01	10
	s_4	00	11	10
f_3	s_5	00	00	10
	s_6	10	01	00

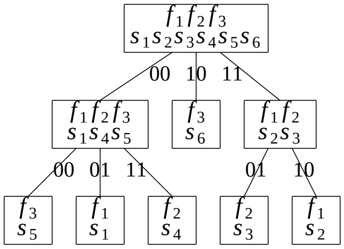

Figure 2: A tree based on Table 2

A practical multiple observation times fault simulator [16] used to obtain the responses of modeled faults may not expand the faulty responses in full, i.e., it may implicitly consider subsets of states by using partially specified initial states. For example, the faulty responses shown in Table 3 may be obtained for the same circuit considered in Tables 1 and 2. The corresponding fault location tree is shown in Figure 3. In this case, there are two output sequences represented by (20,01,00), that do not allow f_1 and f_3 to be distinguished. All other output responses can be uniquely attributed to one fault. In addition, f_2 can always be uniquely identified.

Table 3: Fault location - practical multiple observation times

		time		
fault	sequence	1	2	3
f_1	s_1	22	21	00
	s_2	22	20	01
f_2	s_3	22	21	10
f_3	s_4	20	00	10
	s_5	20	01	00

3.2 A resolution measure

Next, we select a measure of the effectiveness of a test sequence in fault location, that will be used later in our experiments. A

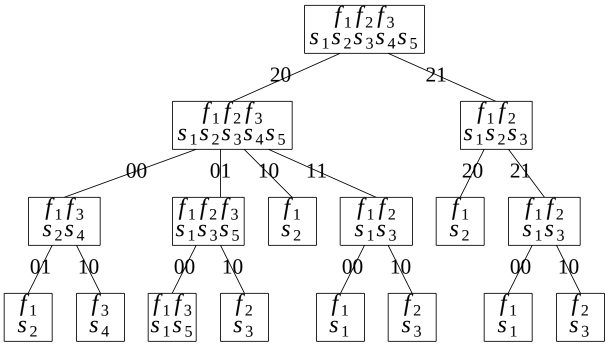

Figure 3: A tree based on Table 3

measure based on a fault location tree is most detailed and accurate in describing the indistinguishable faults. However, the construction of a fault location tree may be computationally intensive. To avoid the need to compute a tree, we use a measure similar to the measure of [11], which is based on maximal subsets of indistinguishable faults. We first illustrate it based on the examples above, and then give a procedure for computing it.

In Figure 1, we have two pairs of faults in the leaves of the fault location tree, namely, $F_1 = \{f_1, f_3\}$ and $F_2 = \{f_2, f_3\}$. No triples exist. Thus, F_1 and F_2 are maximal indistinguishable fault sets. In addition, all the faults are contained in $F_1 \bigcup F_2$. Our measure should indicate that the maximal indistinguishable sets consist of two pairs.

In Figures 2 and 3, we compute the measure based on the sequences stored in the leaves of the tree. Note that in Figure 1, there is one sequence corresponding to each fault, therefore, we can use faults or sequences interchangeably. In Figures 2 and 3, some faults are associated with multiple sequences. We have two options in this case. (1) To translate the sequences into their corresponding faults and consider maximal indistinguishable fault sets, or (2) to consider maximal sets of compatible (or indistinguishable) sequences in the computation of the measure. To show the difference between the two options, consider Figure 3. In Figure 3, we have one pair of compatible sequences $\{s_1, s_5\}$, and all other sequences uniquely identify the fault producing them. The pair $\{s_1, s_5\}$ belongs to faults $\{f_1, f_3\}$. According to Option 1, we would say that the maximal indistinguishable fault sets are $\{f_1, f_3\}$ and $\{f_2\}$. According to Option 2, we would say that the maximal indistinguishable sequence sets are $\{s_1, s_5\}$, $\{s_2\}$, $\{s_3\}$, $\{s_4\}$. Thus, a resolution measure based on sequences is higher than a resolution measure based on faults (there are more size-one sets of compatible sequences than size-one sets of indistinguishable faults). We maintain that the use of sequences results in a more accurate measure. For example, in Figure 3, saying that f_1 and f_3 are indistinguishable implies that there is no output sequence that distinguishes them, however, we know that s_2 produced by f_1 and s_4 produced by f_3 are distinguishable. In the following discussion, we use Option 2. We discuss below its similarity to the notion of partially detectable faults [14] and the fault coverage measure (called a fault detection coefficient) from [16]. We point out that the measure has a disadvantage in that it depends on the number of sequences used for every modeled fault, and to what extent the outputs are specified. Thus, different simulation processes used to obtain the multiple observation times sequences for modeled faults may result in different measures. However, this effect is expected to be small, and overridden by the general effect of using the multiple observation times approach.

Based on the maximal sets of indistinguishable sequences, the resolution measure is given as a set of pairs $\{(i, n(i))\}$, indicating that there are $n(i)$ maximal indistinguishable sets of size i. Thus, for Figure 1 we have {(2,2)}, for Figure 2 we have {(1,6)}, and for Figure 3 we have {(1,3), (2,1)}. In addition, we compute the proportion of sequences that allow unique fault identification as $P = \frac{n(1)}{N_S}$, where N_S is the total number of sequences. Thus, for Figure 1 we have $P = 0$, since $n(1) = 0$; for Figure 2 we have $P = 6/6 = 1$; and for Figure 3 we have $P = 3/5 = 0.6$. The examples show that the measure improves as the output sequences for modeled faults are specified. The same trend is shown for benchmark circuits in Section 4.

If a given fault f has two compatible sequences s_1 and s_2, then s_1 and s_2 are not included in the same maximal set. The reason is that if the circuit-under-test produces a response that is compatible with either s_1 or s_2, then the fault in the circuit is identified in either case as f. Thus, the resolution measure should not decrease if s_1 and s_2 are compatible.

When computing the maximal sets of indistinguishable sequences to derive the resolution measure, we note that the number of sets of a given size i may be exponential in i. We therefore compute the sets in steps. At step i, we compute indistinguishable sets of size i, using information regarding indistinguishable sets of size $i-1$. The procedure is given next.

Procedure 1: Computing indistinguishable subsets

(1) Set U to contain every output sequence. Mark every sequence as *maximal*. Set $i = 2$.

(2) For every set of sequences S of size i, such that all its subsets of size $i-1$ are contained in U, and S does not contain sequences belonging to the same fault, add S to U and mark every subset of size $i-1$ of S as *non−maximal*.

(3) If any set was added to U, and if $i+1$ and $|U|$ do not exceed predetermined bounds, set $i = i+1$ and go to Step 2.

Procedure 1 terminates when the number of sets of size i exceeds a predetermined limit, or when it is zero. In the latter case, it is ensured that all maximal sets are obtained. In the former case, if the computation is stopped after the sets of size i are computed, then we have all the maximal sets of sizes $1, 2, \cdots, i-1$. The reason is that all subsets of sizes $1, 2, \cdots, i-1$ of an indistinguishable set of size i must also be indistinguishable sets. Thus, if no indistinguishable set of size i was obtained that marked as non-maximal a set S_j of size $j < i$, then set S_j will not be marked as non-maximal even when sets larger than i are computed. In particular, once all compatible pairs are computed, all the maximal sets of size 1 are obtained, and P can be computed. In our experiments, to report the results in a uniform manner, we set a fixed bound on the sizes of the indistinguishable sets computed and on their number. We point out that a procedure similar to the one of [11] can be used to generate the maximal subsets. However, Procedure 1 is sufficient for our goal of demonstrating the improvement in resolution obtained by the multiple observation times approach.

2.3 Partially distinguishable faults

In the study of fault detection under multiple observation times [14], it was shown that some undetectable faults exhibit faulty responses under some initial states, whereas under other initial states, they exhibit fault free responses. Such faults were called *partially detectable* in [14]. A similar situation can occur when considering fault location. For example, in Table 4, the faults f_1 and f_2 cannot be distinguished if the output response (0,0,0,0) is observed, however, all the other sequences they produce distinguish the two faults. The faults f_1 and f_2 are said to be *partially distinguishable*. Our resolution measure takes into account such faults, similar to the fault coverage measure from [16]. This is due to the fact that we defined the measure based on the individual sequences produced by the circuit in the presence of modeled faults (Option 2 above).

Table 4: Partially distinguishable faults

time	sequence	1	2	3	4
f_1	s_1	0	0	0	0
	s_2	0	1	0	0
	s_3	1	0	1	0
f_2	s_4	0	0	0	0
	s_5	1	1	1	0

3. Diagnosis based on pass/fail responses

Next, we consider diagnosis based on pass/fail responses. The discussion above was applied to full responses, i.e., the complete output response of the circuit-under-test and the complete responses of the circuit in the presence of the modeled faults were used to diagnose the circuit. The use of full responses may result in large amounts of data that have to be manipulated during diagnosis. To overcome this problem, pass/fail responses are used [15]. For single output responses (the single observation time case), a *pass/fail* response is computed by comparing the fault free circuit response to the response of the faulty circuit (the circuit-under-test or the circuit in the presence of a modeled fault) one time unit at a time. For a given time unit, if the fault free response is v_1 and the faulty response is v_2, the pass/fail response is

pass (*P*) if v_1 and v_2 are fully specified and identical on all outputs,
fail (*F*) if v_1 and v_2 are different when both are specified, and
undefined (2) otherwise.

For example, the pass/fail responses for the responses of Table 5(a) are shown in Table 5(b).

Table 5: Pass/fail responses under single observation

(a) full responses

	time		
fault	1	2	3
fault free	22	02	11
f_1	20	12	21
f_2	02	02	11

(b) pass/fail responses

	time		
fault	1	2	3
f_1	2	F	2
f_2	2	2	P

To define pass/fail responses under the multiple observation times approach, we must consider the existence of multiple fault free responses. An example is shown in Table 6(a), where two fault free responses are possible (s_1,s_2), fault f_1 has three possible responses (s_3,s_4,s_5), and fault f_2 has two possible responses (s_6,s_7). We define for every faulty sequence two pass/fail responses, one for each fault free response. For example, the pass/fail responses for Table 6(a) are shown in Table 6(b), where s_{ij} is obtained by computing the pass/fail response as above, based on faulty sequence s_i and fault free sequence s_j. The pass/fail sequences in Table 6(b) are grouped in sets corresponding to original faulty sequences s_i. In general, for a given faulty response *s*, we compute a set of pass/fail sequences $S(s)$. The size of $S(s)$ is equal to the number of fault free sequences, and every sequence in $S(s)$ is obtained by computing the pass/fail response based on *s* and a different fault free response. It is important to note that the fault simulation process of [16] may compute different numbers of fault free sequences for different faults, depending on the number of states that need to be considered explicitly in order to conclude that the fault is detected. To simplify the examples in this section, we assume that a fixed number of fault free responses is obtained. However, the definitions we give are not limited to this case.

The diagnosis process based on multiple pass/fail responses proceeds as follows. The response obtained from the circuit-under-test is compared against every possible fault free response, to obtain a set of pass/fail responses, *S*. The resulting set *S* is then compared against the pass/fail sets of modeled faults. A faulty sequence *s* is considered as a possible *match* with the observed response if a match is found between *S* and $S(s)$, i.e., for every sequence in *S* there is a compatible sequence in $S(s)$, and vice versa. More formally, we say that two sets of pass/fail sequences S_1 and S_2 are *pass/fail−set−compatible*, if for every sequence $s_1 \in S_1$, there is a pass/fail-compatible sequence $s_2 \in S_2$, and vice versa. We say that two pass/fail sequences are *pass/fail−compatible* if they are equal except where one of them is 2. For example, consider the response (00,01,00,01) obtained for the circuit of Table 6. The pass/fail responses, based on the fault free sequences of Table 6(a), are shown in Table 7. These responses are compatible only with s_5. It is not compatible with s_6 for example, since s_9 is not compatible with either s_{61} or s_{62}.

Table 6: Pass/fail responses under multiple observations

(a) full responses

		1	2	3	4
fault free	s_1	00	01	11	10
	s_2	10	01	11	01
f_1	s_3	02	20	11	12
	s_4	02	21	11	02
	s_5	22	21	10	12
f_2	s_6	12	22	12	00
	s_7	22	22	02	11

(b) pass/fail responses

			1	2	3	4
f_1	s_3	s_{31}	2	*F*	*P*	2
		s_{32}	*F*	*F*	*P*	*F*
	s_4	s_{41}	2	2	*P*	*F*
		s_{42}	*F*	2	*P*	2
	s_5	s_{51}	2	2	*F*	2
		s_{52}	2	2	*F*	*F*
f_1	s_6	s_{61}	*F*	2	2	*F*
		s_{62}	2	2	2	*F*
	s_7	s_{71},s_{72}	2	2	*F*	*F*

Table 7: An example of an observed response

	1	2	3	4
s_8	*P*	*P*	*F*	*F*
s_9	*F*	*P*	*F*	*P*

To compute a measure of resolution for pass/fail responses based on indistinguishable sequences, we apply Procedure 1 above, except that now, instead of single sequences, sets of sequences are considered. As in Procedure 1, we do not include in the same indistinguishable set sequences belonging to the same fault. For example, in Table 6(b), we check the following pairs of sets, and arrive at the following conclusions.

$S(s_3) = \{s_{31}, s_{32}\}$ and $S(s_6) = \{s_{61}, s_{62}\}$ are compatible
$S(s_3) = \{s_{31}, s_{32}\}$ and $S(s_7) = \{s_{71}, s_{72}\}$ are not compatible
$S(s_4) = \{s_{41}, s_{42}\}$ and $S(s_6) = \{s_{61}, s_{62}\}$ are compatible
$S(s_4) = \{s_{41}, s_{42}\}$ and $S(s_7) = \{s_{71}, s_{72}\}$ are not compatible
$S(s_5) = \{s_{51}, s_{52}\}$ and $S(s_6) = \{s_{61}, s_{62}\}$ are compatible
$S(s_5) = \{s_{51}, s_{52}\}$ and $S(s_7) = \{s_{71}, s_{72}\}$ are compatible

In this example, we do not form larger groups of indistinguishable sequences, since we have only two faults, and any larger set would have to include two sequences belonging to the same fault. The maximal indistinguishable sets are $\{s_3, s_6\}$, $\{s_4, s_6\}$, $\{s_5, s_6\}$ and $\{s_5, s_7\}$.

4. Experimental results

We considered MCNC Finite-state machine benchmarks. The modeled faults were all the collapsed single stuck-at faults in a gate-level implementation. We used the test generation procedure from [13] to obtain test sequences under the multiple observation times approach. We then used the simulator from [16] to obtain faulty responses under single and multiple observation times. The simulator first performs fault simulation under the conventional, single observation time approach. For every undetected fault, it then selects a time unit where the states of the fault free and faulty circuits are maximally specified, and it specifies the unspecified state variables one at a time in all possible ways, until it can deduce whether or not the fault is detected. It also identifies partially detected faults. We considered full responses as well as pass/fail responses. Each experiment consists of three parts.

Part 1: Computation of the resolution measure based on single observation time responses.

Part 2: Computation of the resolution measure based on multiple observation times responses. Here, expansion of unspecified state variables starts from the time unit with a minimum number of unspecified state variables. In addition, expansion is done only for faults which are undetected under the single observation time approach. Thus it may not fully exploit the potential for fault resolution by the given test sequence.

Part 3: Computation of the resolution measure based on multiple observation times responses for faults undetected by the single observation time approach, as in Part 2. However, here, expansion of unspecified state variables starts from time unit 1. Thus, the fault resolution potential of the given test sequence is fully realized for the faults considered.

Parts 2 and 3 can be regarded as two extremes of the multiple observation times approach. It is also possible to explore other variations, with smaller amounts of data than Part 3, and larger resolution than Part 2. In addition, the use of the multiple observation times approach can also improve the resolution for faults that are detected by the single observation time approach, which are not considered under the multiple observation times approach in our experiment.

The circuits considered are described in Table 8. A "*" near the number of faults indicates that the number of faults was limited to restrict the amount of data that had to be handled. The third column of Table 8 gives the number of outputs, and the fourth column gives the test length produced by the procedure from [13]. For each part of the experiment, we then give the number of faults detected according to the simulator of [16]. In parentheses, we give the number of partially detectable faults, if any are found.

The resolution measures obtained for full responses are reported in Table 9. The "part" column refers to Parts 1-3 above. The following columns give the number of maximal indistinguishable sets of sizes one and two, and the total number of sets of size three (larger sets were not computed). It can be seen that the resolution improves significantly from one part to the next in most cases. It is important to note that in most cases, the difference in fault coverage (the percentage of faults detected) between the single and the multiple observation times simulations is not significant (cf. Table 8). Thus, the improvement in the resolution of the fault location process cannot be explained by the number of faults detected, and thus results mainly from the use of multiple observation times fault location as proposed here. The number of indistinguishable triples is marked NA in Table 9 when it was larger than 200,000. Triples were not considered in such a case. In addition, Part 3 is not reported for three circuits, for which it created large amounts of data when all initial states were considered.

We also computed the resolution measures for diagnosis based on pass/fail responses. The results are shown in Table 10, in a similar format to Table 9. It can be seen that the resolution in Table 10 is lower than the corresponding resolution in Table 9. This is expected due to the lower amount of information used, and was observed in other works as well. Comparing the results for the three parts of the experiment, it can be seen that significant improvements in resolution are obtained when going from Part 1 to Part 2 and to Part 3.

5. Concluding remarks

We showed the application of the multiple observation times approach to fault location. Fault location was based on comparison of the response of the circuit-under-test with responses of modeled faults. We demonstrated the advantages of the multiple observation times approach in this context, and showed that it allows modeled faults, and therefore fault sites, to be identified more accurately. A definition of pass/fail diagnosis suitable for the multiple observation times approach was also given, that allows pass/fail responses to be used in order to reduce the amounts of data required for fault diagnosis. Experimental results were presented to demonstrate the different aspects of fault location based on multiple observation times.

References

[1] E. J. McCluskey, "Test and Diagnosis Procedure for Digital Networks", Computer, Jan. 1971, pp. 17-20.

[2] M. Abramovici and M. A. Breuer, "Multiple Fault Diagnosis in Combinational Circuits Based on Effect-Cause Analysis", IEEE Trans. on Computers, June 1980, pp. 451-460.

[3] R. E. Tulloss, "Fault Dictionary Compression: Recognizing when a Fault May Be Unambiguously Represented by a Single Failure Detection", 1980 Test Conf., Nov. 1980, pp. 368-370.

[4] J. Savir and J. P. Roth, "Testing for, and Distinguishing between Failures", 12th Intl. Symp. on Fault-Tolerant Computing, June 1982, pp. 165-172.

[5] J. Richman and K. R. Bowden, "The Modern Fault Dictionary", 1985 Intl. Test Conf., Sept. 1985, pp. 696-702.

[6] H. Cox and J. Rajski, "A Method of Fault Analysis for Test Generation and Fault Diagnosis", IEEE Transactions on CAD, July 1988.

[7] J. A. Waicukauski and E. Lindbloom, "Failure Diagnosis of Structured VLSI", IEEE Design and Test of Comput-

ers, Aug. 1989, pp. 49-60.

[8] P. Camurati, A. Lioy, P. Prinetto and M. Sonza Reorda, "Diagnosis Oriented Test Pattern Generation", European Design Autom. Conf., March 1990, pp. 470-474.

[9] S. D. Millman, E. J. McCluskey and J. M. Acken, "Diagnosing CMOS Bridging Faults with Stuck-at Fault Dictionaries", 1990 Intl. Test Conf., Sept. 1990, pp. 860-870.

[10] K. Kubiak, S. Parkes, W. K. Fuchs and R. Saleh, "Exact Evaluation of Diagnostic Test Resolution", 29th Design Autom. Conf, June 1992, pp. 347-352.

[11] E. M. Rudnick, W. K. Fuchs and J. H. Patel, "Diagnostic Fault Simulation of Sequential Circuits", Intl. Test Conf., 1992, pp. 178-186.

[12] I. Pomeranz and S.M. Reddy, "Test Generation for Synchronous Sequential Circuits Using Multiple Observation Times", 21st Fault-Tolerant Comput. Symp. June 1991, pp. 52-59, also in I. Pomeranz and S.M. Reddy, "The Multiple Observation Time Test Strategy", IEEE Trans. on Computers, May 1992, pp. 627-637.

[13] I. Pomeranz and S. M. Reddy, "Applications of Homing Sequences for Synchronous Sequential Circuits", to appear in the IEEE Trans. on Computers, also available as Technical Report No. 4-7-1992, Dept. of Electrical and Computer Eng., U. of Iowa.

[14] I. Pomeranz and S.M. Reddy, "Classification of Faults in Synchronous Sequential Circuits", to appear in the IEEE Trans. on Computers, also available as Technical Report No. 3-27-1991, Electrical and Computer Engineering Department, University of Iowa.

[15] M. Abramovici, M.A. Breuer and A.D. Friedman, *Digital Systems Testing and Testable Design*, Computer Science Press, 1990.

[16] I. Pomeranz and S.M. Reddy, "Fault Simulation for Synchronous Sequential Circuits Under the Multiple Observation Time Testing Approach", 1993 European Test Conf., pp. 292-300.

Table 8: Circuit parameters

				detected (p.det)		
				1	2	3
circuit	faults	out	length	s.obs	m.obs	m.obs
bbara	119	5	273	107	119	119
bbsse	100*	11	491	76	78	78
bou	100*	11	418	90	98	98(2)
dk17	159	3	159	32	159	159
dk27	78	5	49	3	77	77
ex4	100*	14	428	78	82	82(1)
ex6	213	11	100	200	210	210
opus	100*	10	597	84	89	89(5)
pma	100*	13	437	62	66	66
train11	150	5	205	11	146	148

Table 9: Computation of the measure for full responses

circuit	part	seq	single	pairs	triples
bbara	1	119	0(0.0%)	2	7262
	2	157	22(14.0%)	7	1196
	3	187	58(31.0%)	8	136
bbsse	1	100	5(5.0%)	4	2115
	2	114	15(13.2%)	18	1555
bou	1	100	0(0.0%)	1	2643
	2	132	7(5.3%)	13	130
	3	410	333(81.2%)	40	5
dk17	1	159	2(1.3%)	4	238914
	2	277	105(37.9%)	24	46
	3	1048	907(86.6%)	67	75
dk27	1	78	0(0.0%)	0	52246
	2	271	147(54.2%)	33	15
	3	603	441(73.1%)	153	84
ex4	1	100	3(3.0%)	11	1871
	2	122	25(20.5%)	6	817
	3	430	130(30.2%)	38361	NA
ex6	1	213	0(0.0%)	5	8076
	2	245	2(0.8%)	55	270
	3	290	236(81.4%)	15	57
opus	1	100	15(15.0%)	1	1189
	2	123	38(30.9%)	4	165
	3	340	162(47.7%)	20	177592
pma	1	100	0(0.0%)	1	8682
	2	150	50(33.3%)	8	5984
train11	1	150	2(1.3%)	7687	NA
	2	479	273(57.0%)	62	37
	3	2227	1583(71.1%)	2599	1809

Table 10: Computation of the measure for pass/fail responses

circuit	part	seq	single	pairs	triples
bbara	1	119	0(0.0%)	0	9332
	2	157	1(0.6%)	3	4239
	3	187	42(22.5%)	18	305
bbsse	1	100	0(0.0%)	0	2156
	2	114	0(0.0%)	80	1613
	3	460	60(13.0%)	51954	NA
bou	1	100	0(0.0%)	0	2990
	2	132	0(0.0%)	0	2260
	3	410	155(37.8%)	536	532
dk17	1	159	0(0.0%)	8947	NA
	2	277	84(30.3%)	29	1088
	3	1048	679(64.8%)	98	847
dk27	1	78	0(0.0%)	0	76076
	2	271	111(41.0%)	69	203
	3	603	274(45.4%)	351	3615
ex4	1	100	0(0.0%)	0	2255
	2	122	22(18.0%)	6	851
	3	430	102(23.7%)	38405	NA
ex6	1	213	0(0.0%)	0	9722
	2	245	0(0.0%)	0	2599
	3	290	51(17.6%)	25	545
opus	1	100	0(0.0%)	0	1607
	2	123	23(18.7%)	7	182
	3	340	75(22.1%)	11478	NA
pma	1	100	0(0.0%)	0	8876
	2	150	49(32.7%)	19	5984
	3	1278	138(10.8%)	574513	NA
train11	1	150	0(0.0%)	11175	NA
	2	479	240(50.1%)	134	729
	3	2227	1461(65.6%)	3640	5238