Testable design of RMC networks with universal tests for detecting stuck-at and bridging faults

B.B. Bhattacharya, B.Sc., M.Tech., B. Gupta, B.Sc., M.Tech., S. Sarkar, M.Sc., and Prof. A.K. Choudhury, Ph.D.

Indexing terms: *Networks, Logic, Fault location*

Abstract: In the paper we investigate whether the function-independent test set for detecting single stuck-at faults in networks realising Reed–Muller canonic (RMC) expansions of switching functions is sufficient to detect all bridging faults in such networks. The investigation, however, reveals its insufficiency, and to circumvent this we propose a technique of augmenting the network with some additional observation points, so that a universal test set can be designed for detecting bridging faults as well.

1 Introduction

For the last few years, several researchers [3, 7, 9] have shown considerable interest in the study of stuck-at fault detection in combinational networks. However, apart from the stuck-at fault model there exists another important class of faults frequently occurring in logic networks, particularly in MOS LSI circuits, which are known as bridging faults. The occurrence of such faults is mainly due to circuit malfunction resulting in a short circuit between two or more lines of the circuit. Also, a breakdown of insulation between the metallisation on a chip, or some accidental short between two conducting paths, may cause the occurrence of bridging faults. With the advent of integrated-circuit technology diagnosis of bridging faults has received considerable attention [1, 2, 11–13]. In this paper, we discuss a relatively unexplored problem of devising a universal test set for detecting bridging faults in combinational logic circuits. However, no remarkable work has yet been reported regarding the determination of test sets for bridging faults in combinational networks. Such determination of test sets is totally dependent on functions and their circuit topology, and no function independent test set is yet known to exist. And as such, a recent trend in the methodology of fault diagnosis in combinational networks gives attention to their testable design with function independent test sets. We thus deal with this aspect, in so far as the problem of bridging fault detection is concerned.

It is well known that only in a very few restricted classes of combinational networks, a universal, i.e. a function independent test set for stuck-at faults, exists. For example, AND-EXOR arrays based on the Reed–Muller canonic (RMC) expansions of switching functions possess universal test sets for detecting stuck-at fault [3]. Besides this, methods for minimal realisation of such networks are known to exist [4–6, 8, 10]. One most important advantage of the test set [3] is that it has a simple algebraic structure and hence can be generated easily. However, the fault coverage of this test set with respect to bridging faults has yet to be examined, and will be discussed in the following Sections.

Paper 3801E (C2, E10), first received 1st February 1983 and in revised form 17th January 1985

Mr. Bhattacharya is with the Electronics Unit, and Mr. Gupta is with the Computer Science Unit of the Indian Statistical Institute, 203 B.T. Road, Calcutta 700035, India. Mr. Sarkar is with the Institute of Radio Physics and Electronics and Prof. Choudhury is with the Department of Computer Science, Calcutta University, 92 Acharya P.C. Road, Calcutta 700009, India

In order to illustrate the effect of a bridging fault let us consider a logic circuit as shown in Fig. 1A. The bridging

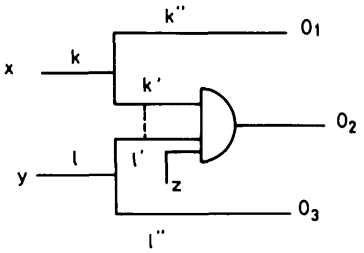

Fig. 1A *Original circuit*

fault between two lines h and m can be modelled as either a wired-AND or wired-OR function [2]. This fault has an effect of an AND function for positive logic (i.e. where logic 1 is represented by higher voltage) and that of an OR function for negative logic (i.e. where logic 0 is represented by higher voltage) [1, 2]. The equivalent faulty networks are shown in Fig. 1B. In other words, a bridging fault

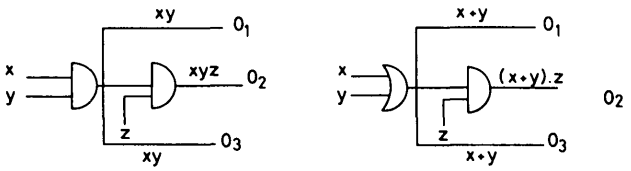

Fig. 1B *Equivalent faulty circuits for AND and OR bridging faults*

between two lines h and m would change both the functional values f_h and f_m on lines h and m to $(f_h \cdot f_m)$ in the case of AND bridging, and to $(f_h + f_m)$ in the case of OR bridging. In addition, all lines that are directly connected to h and m in the physical layout of the concerned network would be equally affected [1].

2 Some properties of Reed–Muller canonic (RMC) expansions of switching functions

The RMC expansion of any switching function $f(x_1, x_2, \ldots, x_n)$ is expressed as

$$f(x_1, x_2, \ldots, x_n) = a_0 \oplus a_1 \overset{*}{x}_1 \oplus a_2 \overset{*}{x}_2 \oplus \cdots \oplus a_j \overset{*}{x}_{j_1} \overset{*}{x}_{j_2} \cdots \overset{*}{x}_{j_m} \oplus \cdots \oplus a_{2^n-1} \overset{*}{x}_1 \overset{*}{x}_2 \cdots \overset{*}{x}_n$$

where

(i) $\overset{*}{x}_j = x_j$ or $\bar{x}_j$ and is fixed for a particular expansion

(ii) a_0 = constant term in the expansion; it may be 0 or 1

(iii) $a_j = 0$ or 1, for $1 \leqslant j \leqslant 2^n - 1$

(iv) $j = 2^{j_1 - 1} + 2^{j_2 - 1} + \cdots + 2^{j_m - 1}$

(v) there are 2^n possible expansions corresponding to 2^n possible combinations of $\overset{*}{x}_1, \overset{*}{x}_2, \ldots, \overset{*}{x}$. These expansions are unique for any given function.

The realisations of such expansions of switching functions are known as AND-EXOR arrays based on RMC expansions or simply RMC networks. These arrays may be two-dimensional cellular cascades. The vertical cascades consist of a set of AND gates, and the horizontal cascade (collector row) consists of a set of EXOR gates. One such realisation for a function $F_0 = x_1 \oplus x_1 x_2 \oplus x_2 x_3 x_4 \oplus x_2 x_3 x_4 x_5$ is shown in Fig. 2. A significant saving in logic

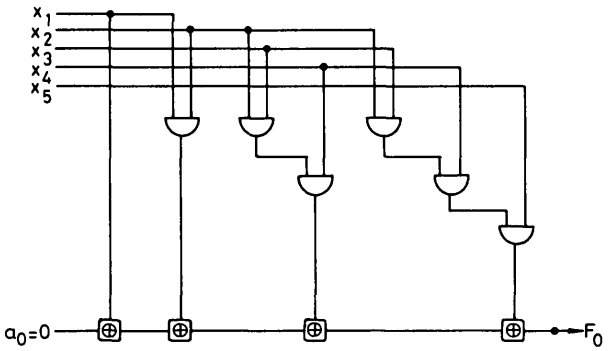

Fig. 2 *Network realising the function* F_0
$F_0 = x_1 \oplus x_1 x_2 \oplus x_2 x_3 x_4 \oplus x_2 x_3 x_4 x_5$

levels can be achieved if the restriction over the number of inputs to the AND gates is withdrawn. In other words, all the AND gates need not necessarily have the same number of inputs and this number may vary from 2 to n, where n is the number of variables present in the function realised. This type of realisation has notionally two levels only. However, from the point of view of single stuck-at fault detection, the same function independent test set is sufficient in each of the above two realisations. In this paper we are interested in detecting bridging faults that can occur involving lines in the same level of the network, and hence the two-level realisation described above is the circuit of choice because of its smaller number of levels compared to that of two-dimensional cellular cascade realisation. Fig. 3 shows such a realisation for the function F_0.

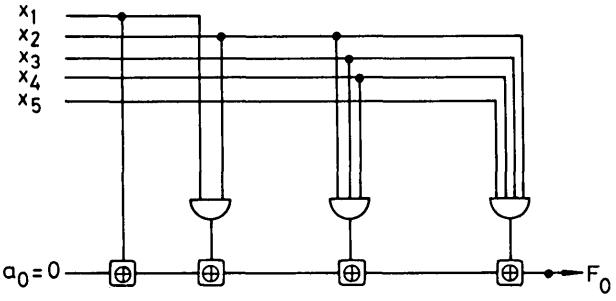

Fig. 3 *Network realising the same function* F_0 *as in Fig. 2*

The following two lemmas follow from some simple observations regarding RMC expansions of switching functions.

Lemma 1: Let P_i and P_j be any two product terms in an RMC expansion of a switching function F_0. Then there must be at least one literal which will be present either in P_i, but not in P_j, or in P_j, and not in P_i. Proof of Lemma 1 is obvious.

Lemma 2: Any two product terms P_i and P_j of an RMC expansion are related to each other by one of the following two relations R_1 and R_2:

R_1: Either $P_i \supset P_j$, or $P_i \subset P_j$

R_2: $P_i \not\supset P_j$, $P_i \not\subset P_j$ and $P_i \cap P_j \neq \varnothing$ (null)

Proof: Suppose that neither of the two relations R_1 and R_2 hold good in the case of the two product terms P_i and P_j taken. Hence the only possible relations left are $P_i = P_j$ and $P_i \cap P_j = \varnothing$ (null). The first one is immediately ruled out because otherwise the RMC expansion will be devoid of P_i and P_j, contradicting our assumption that P_i and P_j are two of its product terms. The second one can never exist because in any RMC expansion the polarity of the variables is fixed. Hence the lemma follows.

Definition 1: Consider two product terms P_i and P_j in an RMC expansion of an n-variable function F_0. Then the literal/literals present in P_i and not in P_j, and vice versa, is/are called the control literal/literals with respect to P_i and P_j.

We now state the following function independent test set T for single stuck-at faults in an AND-EXOR array realising an RMC expansion of an n-variable function F_0. This was originally derived by Reddy [3]. He has shown that this test set T fails to detect single stuck-at faults at the primary inputs if the corresponding input variables appear in an even number of product terms of the RMC expansion. However, Reddy has shown that the test set will be sufficient if the original network is augmented by adding some extra AND gates [3]. The test set T is given by

$$T = T_1 \cup T_2$$

where

$$T_1 = \begin{array}{c} \begin{matrix} a_0 & x_1 & x_2 & x_3 & \cdots & x_n \end{matrix} \\ \begin{bmatrix} 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 1 & 1 & \cdots & 1 \\ 1 & 0 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 1 & 1 & \cdots & 1 \end{bmatrix} \end{array}$$

and

$$T_2 = \begin{array}{c} \begin{matrix} a_0 & x_1 & x_2 & x_3 & \cdots & x_n \end{matrix} \\ \begin{bmatrix} d & 0 & 1 & 1 & \cdots & 1 \\ d & 1 & 0 & 1 & \cdots & 1 \\ d & 1 & 1 & 0 & \cdots & 1 \\ \vdots & & & & & \vdots \\ d & 1 & 1 & 1 & \cdots & 0 \end{bmatrix} \end{array}$$

where d can be taken as 0 or 1 and the cardinality of the test set T is

$$|T| = |T_1| + |T_2| = 4 + n$$

We will show in the later Sections that this universal test set T is sufficient to detect any bridging fault included in our proposed fault model in the network, provided the fault is an OR-type, and we will also show that some more function independent tests are necessary in case the fault is AND-type. However, in either case it requires augmentation of the network.

3 Bridging faults in a RMC network

3.1 Definitions

Before proceeding to the detection problem of bridging faults in such networks, we will first present the following definitions and auxiliary results.

Definition 2: Any bridging fault involving two lines h and m in a logic network is said to be a single bridging fault if none of the lines h and m is a fanout stem or a fanout branch line.

Definition 3: Any bridging fault involving more than two lines, either physically or logically, (by the term logical we mean to incorporate the fact that if a line, say h, is involved in a bridging fault, and if h happens to be a fanout stem or fanout branch line, then all the lines emanating from the parent stem line would also be logically involved in the fault) is called a multiple bridging fault.

Definition 4: Let f_{m_1} and f_{m_2} be two bridging faults denoted by $(h_1, h_2, \ldots, h_{k_1})$ involving lines $h_1, h_2, \ldots, h_{k_1}$ and $(l_1, l_2, \ldots, l_{k_2})$ involving lines $l_1, l_2, \ldots, l_{k_2}$ respectively. Note that, if any of the lines involved in f_{m_1} (f_{m_2}) happen to be a fanout stem or branch line, then all the lines emanating from the parent stem line would be automatically involved in f_{m_1} (f_{m_2}), so far as the logical effect of the bridging fault is concerned and thereby an augmented set f'_{m_1} (f'_{m_2}) of involved lines is created. If $f'_{m_1} \cap f'_{m_2} = \varnothing$ (null), then the fault instance defined by the simultaneous occurrence of $\{f'_{m_1}, f'_{m_2}\}$ is called a multiple group bridging fault of multiplicity two. Similarly the idea can be extended to define a multiple group bridging fault of higher multiplicity, say n: $\{f'_{m_1}, f'_{m_2}, \ldots, f'_{m_n}\}$ such that

$$\forall\, i, j, \quad i, j \in (m_1, m_2, \ldots, m_n), \quad i \neq j,$$

$$f'_i \cap f'_j = \varnothing \text{ (null)}.$$

Definition 5: Any bridging fault involving the input lines of a logic gate in a logic network is said to be an intragate bridging fault.

Definition 6: Any bridging fault involving input lines to different logic gates in a logic network is said to be an intergate bridging fault.

Definition 7: Any bridging fault involving lines of the same logic level in a logic network is said to be an intralevel bridging fault.

Definition 8: Let f_b be an AND/OR bridging fault between two lines h and m. Then it is symbolically represented as $f_b = *(h/m)/+(h/m)$.

Definition 9: Let h be a line in a logic network. Then the functional value at line h is referred to as the line function of h and it is denoted by $f(h)$.

Theorem 1: Let F_0 be a linear function of n-variables, say $x_1, x_2, \ldots, x_n$, and let it be realised by a one-dimensional EXOR array. Then, any OR bridging fault, whether it is single, multiple or multiple group bridging, involving the primary input lines, is always detected by some test belonging to a function independent test set T^*, under the assumption that no bridging fault causes the network to oscillate or to behave as an asynchronous machine. The test set T^* is given by

$$T^* = \begin{array}{c} \begin{matrix} x_1 & x_2 & x_3 & \cdots & x_n \end{matrix} \\ \begin{bmatrix} 0 & 1 & 1 & \cdots & 1 \\ 1 & 0 & 1 & \cdots & 1 \\ 1 & 1 & 0 & \cdots & 1 \\ \vdots & & & & \vdots \\ 1 & 1 & 1 & \cdots & 0 \end{bmatrix} \end{array} \quad n \text{ input vectors}$$

Proof: The proof is constructive.

First, consider a single bridging fault. Let it be $f_b = +(h/m)$ and the line functions of the primary input lines h and m be $f(h)$ and $f(m)$, where $f(h) = x_h$ and $f(m) = x_m$. It is clear that in the presence of f_b, $f(h) = f(m) = x_h + x_m$. To detect the fault, an input vector is to be applied that makes either $f(h) = 1, f(m) = 0$, or, $f(h) = 0, f(m) = 1$ in the fault free condition. The following two input vectors are able to produce the above assignments of values of lines h and m. These are:

(i) $x_h = 0$, and all other x_is $= 1$, for $1 \leqslant i \leqslant n$, $i \neq h$,

(ii) $x_m = 0$, and all other x_is $= 1$, for $1 \leqslant i \leqslant n$, $i \neq m$.

In any case, in the presence of f_b, $f(h) = f(m) = 1$. Thus it is seen that $f(h)$ and $f(m)$ are of opposite values in the absence of f_b, and of same values when f_b is present. This causes a logic value at the network output, which is different from the expected one. Hence f_b is detected. Clearly the two input vectors considered above belong to the set T^*.

Next, we consider a multiple bridging fault. Let it be f_b involving the primary input lines $h, m, \ldots, l$, the corresponding line functions being $f(h) = x_h$, $f(m) = x_m, \ldots, f(l) = x_l$ in the fault free condition. In the presence of f_b all the above line functions become identical and each of them becomes equal to $(x_h + x_m + \cdots + x_l)$. To detect the fault, an input vector t is chosen that imparts 0 at one of the involved lines, say h, and imparts 1 at all other lines, in the fault free condition. In other words, it sets $x_h = 0$ to make $f(h) = 0$ and all other x_is $= 1$, for $1 \leqslant i \leqslant n$, $i \neq h$. Under the application of this input vector $f(h)$ becomes 1, due to the bridging fault, which is otherwise zero in the absence of the fault. This in turn causes a logic value at the network output, which is different from that expected, showing therefore that f_b is detected. Evidently this input vector t belongs to the set T^*.

Lastly we consider multiple group bridging fault. Let it be $f_b = \{f_{b_1}, f_{b_2}, \ldots, f_{b_m}\}$. To detect the fault, we select one of the component faults, say f_{b_1}, and excepting only one line, say h, involved in the bridging f_{b_1} all other primary input lines involved in the bridging fault f_b are fixed at logic value 1, as well as all other primary input lines not involved in the bridging. In other words, an input vector is chosen that sets $x_h = 0$, where $f(h) = x_h$ and all other x_is $= 1$, for $1 \leqslant i \leqslant n$, $i \neq h$. As in the previous two cases $f(h)$ becomes 1 due to f_b causing a change in logic value at the network output line. Thus f_b is detected. Clearly this test also belongs to the set T^*.

Hence we conclude that T^* is sufficient for detecting the different OR-bridging faults considered in the theorem.

Extension of theorem 1 to the case of AND-bridging faults reveals the fact that the universal test set T^* is not sufficient to detect all the AND-bridging faults considered in theorem 1. One such case where the test set T^* fails to detect a fault is the presence of any multiple bridging fault that involves odd number of input lines. The reason behind this failure is quite obvious. However, it can be shown in an identical way as in the case of OR-bridging faults that another function independent (universal) test set T^{**} of cardinality n is sufficient to detect different single, multiple and multiple group AND-bridging faults involving the primary input lines in the EXOR array realising a linear function, $F_0 = x_1 \oplus x_2 \oplus \cdots \oplus x_n$. The test set T^{**} is given by

$$T^{**} = \begin{array}{c} \begin{matrix} x_1 & x_2 & x_3 & \cdots & x_n \end{matrix} \\ \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \end{array}$$

In Fig. 4 a general model of a complement free RMC network is shown, where the two levels of the network

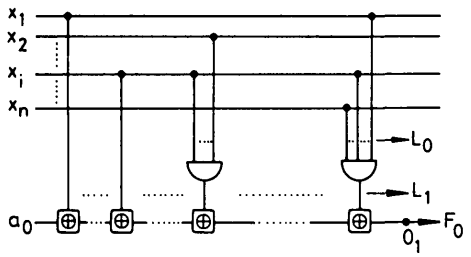

Fig. 4 *General model of a complement free RMC network*

have been designated by L_0 and L_1. However, networks realised in other polarity vectors of input variables will be discussed subsequently.

We now state the following fault model which we consider in the present work.

Fault model:

(*a*) Only intralevel bridging faults will be dealt with. These are mainly;

(i) bridging faults involving only two lines in the L_1-level. These lines may be those that are directly connected to the collector-row EXOR gates from the primary input stems. This type of bridging also includes bridging between two inputs of the same EXOR gate in the collector row.

(ii) the following different types of bridging faults are considered in L_0-level. These are: bridging involving only primary input lines, intragate and intergate bridging faults and any bridging fault involving some input lines of AND gates and some lines that are directly connected to the collector-row EXOR gates from their primary input stems.

(*b*) all single stuck-at faults are considered.

(*c*) feedback bridging faults are not considered.

Without any loss of generality we first assume OR-bridging faults in the network in the following Section.

3.2 OR-bridging

We consider here the network of Fig. 4 which has been realised using negative logic. Each intralevel bridging is considered separately in the following way:

Bridging faults in L_1-level: We classify the different cases of bridging faults in this level in the following subclasses:

Class A: Let the bridging fault be $f_b = +(i/j)$. The line functions of lines i and j are $f(i) = P_i$ and $f(j) = P_j$, respectively, where P_i and P_j are product terms. According to Lemma 2, P_i and P_j are related to each other either by relation R_1 or by R_2. We consider each case separately.

Case 1: When relation R_1 holds good, then either $P_i \supset P_j$ or $P_i \subset P_j$. To detect the fault an input vector is to be applied that sets opposite logic values at the lines i and j in absence of the fault. In other words, it will set either $P_i = 1$, and $P_j = 0$, or $P_i = 0$ and $P_j = 1$. According to Lemma 1 the product term, covered by the other whenever relation R_1 exists between them, always contains at least one control literal. Then clearly an input vector that sets this control literal at 0 and all other $(n-1)$ literals in the binary n-vector at 1, with $a_0 = 0$ or 1, is a test. Under this input assignment, in the absence of the fault $f(i) = 1$, $f(j) = 0$ if $P_i \supset P_j$ and vice versa if $P_i \subset P_j$. However, in either case, the presence of the fault makes both $f(i) = 1$ and $f(j) = 1$. The effect of this change of logic value of either line i or line j will propagate to the network output causing there a change of logic value. Thus the fault is seen to be detected and this input vector is clearly a member of the universal test set T, described earlier.

Case 2: When relation R_2 holds good, i.e. when $P_i \not\supset P_j$, $P_i \not\subset P_j$ and $P_i \cap P_j \neq \varnothing$ (null), then each of P_i and P_j contains at least one control literal. In this case we start with either P_i or P_j, say P_i, and an input vector $\boldsymbol{t}$ is chosen that sets one of the control literals in P_i at 0 to make $P_i = 0$ and all other x_is in the binary n-vector at 1 to make $P_j = 1$, with $a_0 = 0$ or 1. As in the previous case this input vector detects the fault at the network output. Here also $\boldsymbol{t} \in T$.

Class B: Here we consider bridging between two lines h and m, such that $f(h) = x_h$ and $f(m) = P_m$, a product term. The line h may be a fanout branch line. To detect the fault, we design a test vector $\boldsymbol{t}$ such that $f(h) = x_h = 1$ and one of the control literals in P_m, say $x_k = 0$ to make $P_m = 0$, and all other $(n-2)$ variables in the test vector are set at 1, with $a_0 = 0$ or 1. In the presence of the bridging fault, $f(m)$ becomes 1 and this change in value of line m produces a change in value at the network output. This test vector $\boldsymbol{t}$ is also included in the set T. In fact by making $x_h = 1$, the propagation of the effect of the bridging fault to the parent stem line of line h, if it is a fanout branch line has been stopped.

Class C: Let us consider a bridging fault $f_b = +(h/m)$, where $f(h) = x_h$ and $f(m) = x_m$. Unless both of the lines h and m are fanout branches, tests can be found included in the set T to detect f_b as in the previous two cases. However, problems arise when both h and m are fanout branches. Because any input vector chosen from the set T that has a chance to detect f_b at network output must fix either $x_h = 0$ and $x_m = 1$ or $x_h = 1$ and $x_m = 0$. In either case, in the presence of f_b, $x_h = x_m = 1$ and this change of logic value at line h or m, which can be considered as the effect of the bridging fault, propagates to the corresponding stem line and hence to all other fanout branch lines emanating from this stem line. In effect all these lines will have a logic value 1 instead of the applied value 0. This may cause the logic value at the output to remain the same. Thus T may fail to detect f_b. In fact, the detection of such faults by the set T depends on how many times, and in what way, x_h and x_m appear in the RMC expansion. However, we will show later that a proper augmentation of the network will enable T to detect such faults.

Class D: Here we consider a bridging fault $f_b = +(h/m)$, where the lines h and m are the two inputs of a collector row EXOR gate. It is seen that the third row vector of the test set T_1 ($\in T$) always brings a 1 on the horizontal input line and a 0 on the vertical input line simultaneously. Hence, it is a test vector for this type of faults. If one of the input lines is the control line, then by using the second row vector of the test set T_1 we can detect the fault because this input vector always brings a 0 on the horizontal input line, i.e. the control line and a 1 on the vertical input line.

Bridging faults in L_0-level: We consider the different cases of bridging faults in this level in the following way.

Intragate bridging faults: Unless all the lines involved in the bridging are fanout branches, the test set T is able to detect such faults. Let us assume an intragate bridging fault f_b and let the involved lines be $(h, k, \ldots, l)$ such that at

least one of them, say h, is not a fanout branch line. Then a test vector is always found in T that sets $x_h = 0$, where $f(h) = x_h$ and all other x_is $= 1$, for $1 \leqslant i \leqslant n$, $i \neq h$ with $a_0 = 0$ or 1. Under the application of this test vector the faulty output of the affected AND gate will be 1, which is otherwise 0 in the fault-free condition. This will cause a change of the logic value at the output. Hence f_b is detected. However, if all the lines involved in the bridging are fanout branches then it cannot be guaranteed that T will detect the bridging fault. Reasons behind this are similar to that in the class C case.

Intergate bridging faults: Here also T may fail to detect such faults if all the lines involved in the bridging are fanout branches. Reasons are similar as in the case of intragate bridging faults.

Bridging fault involving the primary input lines: The test set T may fail to detect such faults because their detection by the test set T depends on how many times, and in what way, the literals connected to the input lines involved in the bridging appear in the RMC expansion of the realised function.

Lastly we consider a bridging fault f_b that involves some lines that are inputs to some AND gates, and some lines that are directly connected from their input stems to some collector row EXOR gates. In this case also the presence of the fault may not be ensured by applying test vectors included in T. The explanation of this failure is similar to the previous cases.

We shall now show that in the proposed testable network the bridging faults in L_0- and L_1-level, which are undetected by the test set T at network output before augmentation, will be detected by this test set T.

We now propose the following testable design to detect the above mentioned undetectable bridging faults.

A testable design: This is a technique to add an extra AND gate to make the universal test set T sufficient to detect the different types of bridging faults and stuck-at faults considered in the fault model. The output of the added AND gate is assumed to be observed during testing. The augmented network is shown in Fig. 5. This added gate is an n-input AND gate, realising the function $(x_1 \cdot x_2 \cdots x_n)$.

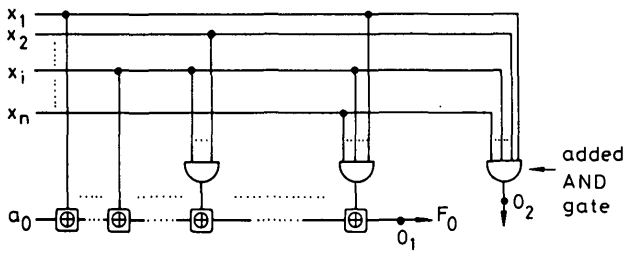

Fig. 5 *Testable design in case of OR-type bridging*

We now show that all the bridging faults that are undetected under the test set T in the unaugmented network will be detected at the observable output point 0_2 by the application of tests from the same test set T. It is interesting to note that the bridging faults in the L_0-level described earlier are nothing other than some single and multiple bridging faults. Any such fault is detected at 0_2 by a test vector that fixes one of the bridged lines, say i, at 0 by making $x_i = 0$, where $f(i) = x_i$ and all other $(n-1)$ variables at 1. In the absence of the fault, the output at 0_2 is 0 under such an input vector, whereas it becomes 1 when the fault happens to occur. This input vector is clearly included in T. Moreover another type of fault, namely multiple group bridging fault in this level, is also detected at 0_2 in the following way. Let the multiple group bridging fault be $f_b = \{f_{b_1}, f_{b_2}, \ldots, f_{b_m}\}$. To detect the fault at 0_2, we select, arbitrarily, one of the component faults, say f_{b_1}, and apply an input vector t that sets only one of the lines, say h, involved in the bridging f_{b_1} at 0, by making $x_h = 0$, where $f(h) = x_h$, and all other x_is in the binary n-vector at 1. In the absence of f_b, the output at 0_2 is 0 whilst it becomes 1 when the fault happens to occur. So, t detects f_b. It is seen that $t \in T$. The class C type of faults described earlier in this Section can also be detected at the observable output 0_2 in a similar way. In each of the above cases, the detection procedure makes use of the fact that the effect of a bridging fault involving a fanout branch line always propagates to the corresponding stem line.

It is clear, following the arguments given by Reddy [3], that all single stuck-at faults in the augmented network are detected by the test set T, either at the primary output or at the output of the added gate.

We now state in theorem 2 the results so far obtained.

Theorem 2: An AND-EXOR array realising the complement free RMC expansion of a switching function F_0 of n-variables can be so augmented by adding an extra AND gate that the universal test set T, of cardinality $(n + 4)$, is sufficient to detect the different intralevel OR-bridging faults and all single stuck-at faults.

It may be noted that any bridging fault involving some lines in the L_0-level and some of the input lines of the testing AND gate, or simply involving some of the input lines of this gate, is always detected by the same test set T at the augmented output 0_2. The detection procedure is identical to that in case of L_0-level bridging described above.

3.3 AND-bridging

Here we consider that the network of Fig. 4 has been realised using positive logic. Also, in the case of AND-bridging similar arguments prevail regarding the failure of the test set T to detect some of the intralevel bridging faults in the unaugmented network. These bridging faults are of the same classes as in the case of OR-bridging.

We have shown earlier that in the case of OR-bridging faults only one extra AND gate is necessary to make the universal test set T sufficient for detecting the different bridging faults and all single stuck-at faults. So naturally the question arises as to whether the addition of only one extra OR gate makes the test set T sufficient in case of AND bridging faults. However, the answer is negative. The following discussion shows that we will have to generate another function independent test set T_α of cardinality n, which, together with the test set T, will be sufficient for detecting the different AND bridging faults considered in the fault model and all single stuck-at faults, if the network is made testable by adding only one extra OR-gate. T_α is given by

$$T_\alpha = \begin{array}{c} \begin{matrix} a_0 & x_1 & x_2 & x_3 & \cdots & x_n \end{matrix} \\ \begin{bmatrix} d & 1 & 0 & 0 & \cdots & 0 \\ d & 0 & 1 & 0 & \cdots & 0 \\ d & 0 & 0 & 1 & \cdots & 0 \\ \vdots & & & & & \vdots \\ d & 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \end{array}$$

We propose the following testable design in which an extra OR gate is added to the AND-EXOR array, as shown in Fig. 6, and the output $0'_2$ of the added gate is assumed to

be observed during testing. This added gate is an n-input OR-gate realising the function $(x_1 + x_2 + \cdots + x_n)$.

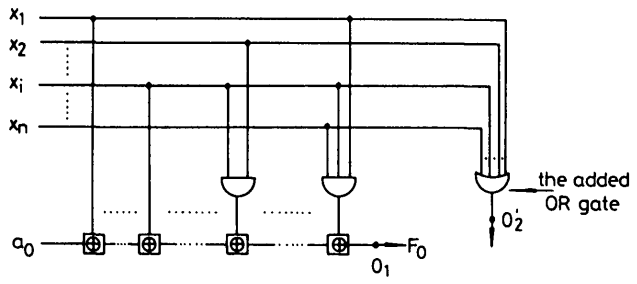

Fig. 6 *Testable design in case of AND-type bridging*

The method used for detecting all the bridging and stuck-at faults of the fault model, by the test sets T and T_α at the observable output points 0_1 and $0'_2$, is similar to that described in the previous Section. That the test set T fails even in this augmented network to detect some bridging faults is clear from the following discussion.

Consider a bridging fault f_b involving some lines h, k, ..., l in the L_0-level and also let us assume that this fault f_b was initially undetectable by T at network output before augmentation. Suppose the corresponding line functions are x_h, x_k, ..., x_l, respectively. In order to detect f_b at the augmented output $0'_2$, at least one of these variables is to be fixed at 0. If we choose some input vector t from the set T to do so, then this input vector sets all other $(n-1)$ variables at 1. Due to the bridging the line functions of all the bridged lines become 0. However, if there exists in the L_0-level at least one line, say i with $f(i) = x_i$, not involved in the bridging, such that $f(i)$ differs from all the line functions of the involved lines, then all lines emanating from the parent stem line of line i together with the line i will not be affected by the bridging fault. So, under this input assignment $f(i)$ will always be 1, as well as an input line to the OR-gate, which is directly connected to the parent stem line of line i, will always have a logic value 1, irrespective of whether the fault is present or not. Hence, the output at $0'_2$ will always have a logic value of 1, regardless of the presence or absence of f_b. Thus t fails to detect f_b. On the other hand, if we select some input vector from the set T_α to fix only one of the variables x_h, x_k, ..., x_l at 1 and all other $(n-1)$ variables at 0, then, in the absence of f_b, output at $0'_2$ will be 1 and it will be 0 when f_b is present. Thus f_b is detected. In fact, in this way it can be easily shown that the test set T together with the test set T_α is sufficient to detect all bridging faults of the fault model.

The following theorem reflects the above obtained results.

Theorem 3: An AND-EXOR array, realising the complement free RMC expansion of a switching function F_0 of n-variables, can be so augmented by adding an extra OR-gate that the universal test set T_u of cardinality $(2n + 4)$ is sufficient to detect the different intralevel AND-bridging faults and all single stuck-at faults. T_u is given by

$$T_u = T \cup T_\alpha$$

Here also, any bridging involving some of the inputs of the testing OR-gate and some other lines in the L_0-level, or simply involving some of the inputs of the testing OR-gate, is always detected at the testing gate output by the same test set T_u.

Corollary 1: Any AND-bridging fault in the L_0-level, that does not change the fault free function at network primary output 0_1, is always detected at the observable output $0'_2$ of the testing OR-gate by a test t, where $t \in T_\alpha$, provided that all of the lines involved in the bridging fault are not fanout branch lines of the same input stem line.

Proof: Consider a bridging fault f_b in L_0-level, which does not change the fault free function at network primary output, and hence is basically an undetectable fault. However, the effect of the bridging fault always propagates to the inputs of the testing OR-gate. Let h be a line that is involved in the bridging fault f_b, and let the line function of line h be x_h. Then an input vector t that sets $x_h = 1$ and all other x_is $= 0$, for $1 \leqslant i \leqslant n$, $i \neq h$, is a test, because under the application of t, the output at the observation point $0'_2$ is 1 in the absence of f_b whilst it becomes 0 in the presence of f_b. It is clear that $t \in T_\alpha$.

As a consequence of corollary 1 questions may arise about the necessity of detecting undetectable bridging faults because in the presence of such faults the network operates properly. However, the presence of such undetectable bridging faults may invalidate the valid tests for detecting some detectable stuck-at faults at network primary output 0_1 [13], when, simultaneously, bridging and stuck-at faults occur. We give an example of such a situation. Consider the network shown in Fig. 7. The fault free function at network output 0_1 is

$$F_0 = x_1 x_2 \oplus x_1 x_2 x_3 \oplus x_1 x_2 x_4$$

We consider a bridging fault $f_b = *(h/k)$, as shown by the dotted line in the Figure. It is seen that f_b does not change

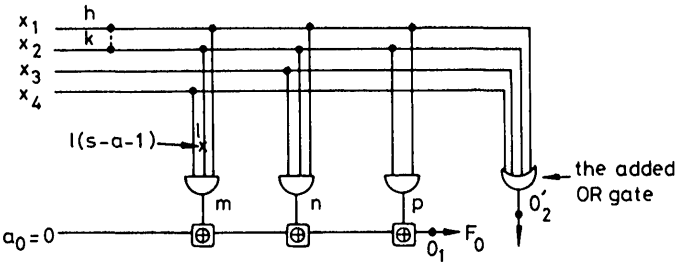

Fig. 7 *Network realising the function* F_0
$F_0 = x_1 x_2 \oplus x_1 x_2 x_3 \oplus x_1 x_2 x_4$

the fault free function F_0. Now, we assume that in presence of f_b the line l has been stuck-at 1. As a result of the simultaneous occurrence of the bridging and the stuck-at fault, the faulty function at network output 0_1 becomes

$$F_f = x_1 x_2 \oplus x_1 x_2 x_3 \oplus x_1 x_4$$

We now try to detect this stuck-at 1 fault by the test designed for it. It is: $x_1 = 1$, $x_2 = 0$, $x_3 = 1$, $x_4 = 1$ and $a_0 = 0$ or 1. Clearly this test belongs to the set T. It is seen that this test fails to detect the stuck-at fault at the output 0_1 because, due to the presence of the bridging and stuck-at fault, the faulty and the fault-free logic values under this test, at lines m, n and p, are identical. In other words, the presence of this undetectable bridging fault has masked the otherwise detectable stuck-at fault at the output 0_1. However, this problem can be totally circumvented by the test set T_α which detects this type of bridging faults at the observable output $0'_2$.

We now give the necessary and sufficient conditions in order that a bridging fault, involving some primary input lines in a complement free RMC network, becomes undetectable by any test at the network primary output. This is illustrated in the following theorems.

Theorem 4a: Let h and m be two primary input lines connected to the literals x_h and x_m, respectively, in a complement free RMC network N realising a Boolean function F_0. Then a bridging fault $f_b = *(h/m)$ is undetectable by any test at the network primary output if, and only if, x_h appears only in all the terms containing the literal x_m in the RMC expansion.

Proof is simple and hence is omitted.

Theorem 4b: Let h and m be two primary input lines connected to the literals x_h and x_m, respectively, in a complement free RMC network N realising a Boolean function F_0. The RMC expansion can always be factored as

$$F_0 = A \oplus x_h \cdot B \oplus x_m \cdot C \oplus x_h x_m \cdot D$$

where the Boolean functions A, B, C and D are independent of x_h and x_m. Then a bridging fault $f_b = +(h/m)$ is undetectable at the network primary output by any test if, and only if,

$$B = C = D$$

Proof is simple and hence is omitted.

Theorems 4*a* and 4*b* depict conditions of undetectability of bridging faults among primary input lines. Note that, whenever the complement free RMC expansion of a Boolean function satisfies such properties, those bridging faults become undetectable irrespective of network topology. Even these bridging faults can be detected at the outputs 0_2 and $0_2'$ of the testing gates.

4 Conclusion

In the present work, it has been shown that in RMC networks bridging faults can be detected by function independent tests, as is the case with stuck-at faults. It has also been shown that the necessary augmentation for detecting both OR and AND-type bridging faults in a complement free RMC network requires the addition of only one extra gate in each case. For any RMC expansion other than complement free, we propose the following augmentation. In the case of OR-bridging faults one OR-gate and one AND-gate, both of which are n-input gates for an n-variable function realised by the network in question, are required. The output functions of the respective gates are $(x_1 + x_2 + \cdots + x_n)$ and $(x_1 \cdot x_2 \cdots x_n)$. One more AND-gate with n_1-inputs is required, where n_1 is the number of distinct complemented literals that appear in the RMC expansion. The input lines of this AND-gate are lines drawn from the outputs of the corresponding inverters used to obtain the n_1 complemented literals. The required tests are the same as in the case of AND-bridging faults in a complement free RMC network. In the case of AND-bridging faults, a proper augmentation can also be carried out similarly.

Although we have not included in our fault model the occurrence of any feedback bridging faults, still the testable designs proposed in this paper are capable of detecting a large number of such faults. For example, consider a bridging fault $f_b = +(h/k)$, where $f(h) = x_h$ and $f(k) = P_k$, a product term in which the literal x_h is present. We detect this fault by fixing the logic value of x_h at 1 and some control literal x_c in P_k at 0 to make $P_k = 0$, so that the effect of f_b, which causes the line k to be set at 1, cannot propagate to the parent stem line of line h and therefore cannot affect the other branch lines emanating from this stem line.

From the point of view of fault detection, so far almost all existing works are based on stuck-at fault models, and the incorporation of bridging faults has, so far, been ignored, although in MOS-LSI circuits the probability of bridging faults in different layers of metallisation and diffusion is substantially high. That is why our present work includes bridging faults. We have not considered interlevel bridging faults because in that case the behaviour of the circuit would become highly complex and some additional tests might be necessary. Moreover we can avoid the possibility of interlevel bridging faults to a large extent by proper fabrication of the chip so that the proximity effect among lines corresponding to different levels is minimised, with a consequent reduction in the probability of occurrence of interlevel bridging faults.

5 References

1 FRIEDMAN, A.D.: 'Diagnosis of short circuit faults in combinational circuits', *IEEE Trans.*, 1974, **C-23**, pp. 746–732
2 MEI, K.C.Y.: 'Bridging and stuck-at faults', *ibid.*, 1974, **C-23**, pp. 720–727
3 REDDY, S.M.: 'Easily testable realization for logic functions', *ibid.*, 1972, **C-21**, pp. 1183–1188
4 WU, X., CHEN, X., and HURST, S.L.: 'Mapping of Reed-Muller coefficients and the minimization of exclusive OR-switching functions', *IEE Proc. E, Comput & Digital Tech.*, 1982, **129**, pp. 15–20
5 MUKHOPADHYAY, A., and SCHMITZ, G.: 'Minimization of EXCLUSIVE OR and LOGICAL EQUIVALENCE switching circuits', *IEEE Trans.*, 1970, **C-19**, pp. 132–140
6 MARINCOVIC, S.B., and TOSIC, Z.: 'Algorithm for minimal polarized polynomial form determination', *ibid.*, 1974, **C-23**, pp. 1313–1315
7 HAYES, J.P.: 'On modifying logic networks to improve their diagnosability', *ibid.*, 1974, **C-23**, pp. 56–62
8 EVEN, S., KOHAVI, I., and PAZ, A.: 'On minimal modulo-2 sum of products for switching functions', *ibid.*, 1967, **EC-16**, pp. 671–674
9 HAYES, J.P., and FRIEDMAN, A.D.: 'Test point placement to simplify fault detection', *ibid.*, 1974, **C-23**, pp. 727–736
10 SALUJA, K.K., and ONG, E.H.: 'Minimization of Reed-Muller Canonic expansion', *ibid.*, 1979, **C-28**, pp. 535–537
11 ISOUPVICZ, I.: 'Optimal detection of bridge faults and stuck-at faults in two level logic', *ibid.*, 1978, **C-27**, pp. 452–455
12 KARPOVSKY, M., and SU, S.Y.H.: 'Detection and location of input and feedback bridging faults among input and output lines', *ibid.*, 1980, **C-29**, pp. 525–527
13 KODANDAPANI, K.L., and PRADHAN, D.K.: 'Undetectability of bridging faults and validity of stuck-at fault tests', *ibid.*, 1980, **C-29**, pp. 55–59

A.K. Choudhury received his B.Sc. degree in Physics from Patna University and M.Sc. degree in Applied Physics and Ph.D. from Calcutta University. Currently he is a Professor in the Department of Computer Science, University of Calcutta. His fields of research include circuit theory, control engineering, switching theory and logic design, and fault-tolerant computing.

Bhargab B. Bhattacharya obtained his B.Sc. degree in Physics, and his B.Tech. and M.Tech. degrees in Radio Physics and Electronics, from Calcutta University. He has recently submitted his doctoral dissertation in Computer Science to the University of Calcutta. Currently he holds a faculty position with the Indian Statistical Institute, Calcutta, and his research interest includes fault-tolerant computing, logic design, parallel processing, and artificial intelligence.

Satyabrata Sarkar obtained his M.Sc. degree in Physics from Bihar University, India. He is a lecturer in Physics at Jangipur College, Jangipur and is also a visiting lecturer at the Department of Computer Science, Calcutta University. He carried out his research work whilst he was at the Institute of Radio Physics and Electronics. He has recently submitted his doctoral dissertation in Computer Science to the University of Calcutta. His research interests include fault tolerant computing, logic design and applications of graph theory.

Bidyut Gupta obtained his B.Sc. degree in Physics, and his B.Tech. and M.Tech. degrees in Radio Physics and Electronics from Calcutta University. Recently he has submitted his doctoral dissertation in Computer Science to the University of Calcutta. At present he is a faculty member of the Indian Statistical Institute, Calcutta. His research interest includes fault tolerant computing, logic design and applications of graph theory.