Network Structure for Testability Improvement in Exclusive-OR Sum of Products Reed-Muller Canonical Circuits

V. Geetha*, N. Devarajan, P.N. Neelakantan

* Associate Professor of Electrical Engg., Government College of Engineering (Srirangam), Tamilnadu,

geetha@gct.ac.in, +91-92451-58225

Abstract— In this paper, a testable design structure with improved fault identification and detection capability is proposed and compared with a reference structure for the analysis and diagnosis of stuck-at and bridging faults in Exclusive-OR Sum of Products Reed-Muller canonical circuits. Further, a compact method of representing the circuit outputs has been adopted for ease of tabulation and comparison. Simulations of Single stuck-at, Double stuck-at, AND-bridging and OR-bridging faults for a few random functions have been carried out through MATLAB coding. From the test results, it was found that the proposed structure yields fault detection of more than 95% for most of the functions considered, with just n+5 test vectors compared to the Reference Structure. The distinguishability factor has also improved for the proposed structure. The location of the fault can also be diagnosed through the output sets.

Keywords—Reed-Muller Canonical Form, Exclusive-OR Sum of Products, Testable Realization, Single Stuck-at, Double stuck-at, AND-bridging fault, OR-bridging fault

INTRODUCTION

The faults in digital circuits can be classified broadly as Single stuck-at-faults, Multiple stuck-at-faults, Stuck-open faults, Stuck-on faults, Bridging faults, Path delay faults, Transient faults etc. Any arbitrary logic function, in general, can be expressed in Reed-Muller Canonical (RMC) form as $F = (a_0 \oplus a_1x_1^* \oplus a_2 x_2^* \oplus \ldots \oplus a_nx_n^* \oplus a_{n+1} x_1^* x_2^* \oplus \ldots \oplus a_m x_1^* x_2^* \ldots x_n^*)$ where, x_n^* can be x_n or its complement, a_n is either 0 or 1 and $m = 2^n-1$. However, there can be variations in such forms. The different types are Fixed Polarity RMC (FPRM), Positive Polarity RMC (PPRM), Generalised RMC (GRM) and Exclusive-OR Sum-of-Products RMC (ESOP). The FPRM has a restriction that the variables in any of the product terms have to be of the same type namely complementary or non-complementary. For PPRM, the complementary form of variables is not allowed. The GRM may contain both complementary and non-complementary types but the combination of the variables should be unique. The ESOP form does not have any such restriction. Also the ESOP form has the least number of product terms and hence needs the least number of AND gates and is very much suitable for hardware implementation.

Extensive research has been carried out in the field of testing of digital circuits to reduce the number of input vectors. The cardinality of the test vectors proposed by many authors becomes prohibitively excessive for large number of input variables. It was demonstrated that single stuck-at fault detection can be achieved with only n+5 test vectors [6]. The same structure was extended for OR-bridging fault analysis [17] and [18]. This paper proposes a modified structure with n+5 test vectors which gives better results compared to [18].

Two quantitative indices, called identifiability factor and distinguishability factor were considered for comparison of the testability nature of given circuits. The identifiability factor is defined as the ratio of the number of faults correctly identified by the test set to the total number of possible faults of the type considered. The existence of faults can be recognized from the set of outputs measured which will be different from the fault-free circuit. The distinguishability factor pertains to the identical set of outputs among different faults, but the output set of each being very much different from the non-faulty case. The set of binary values for an output was converted into its decimal equivalent for convenience in comparison and ease of tabulation.

LITERATURE SURVEY

A PPRM network for detection of stuck-at faults with a universal test set of size n+4, n being the number of data inputs, was proposed in [1]. Though quite good for self-testing, the method is economical only for the specified form, which obviously has more number of product terms than the other forms in most cases. Multiple stuck-at fault detection for ESOP circuits was carried out in [2]. However, since the cardinality is $2n+6+\sum nC_e$, e = 0 to j, the order of ESOP expression, the test set is not universal and also is too large to be practical for large input functions. Stuck-at and bridging faults with a universal test set for PPRM network has been reported in [3]. Multiple fault detecting GRM realizations was proposed in [4]. Reference [5] described an ESOP implementation with a universal test set of size n+6 for single stuck-at faults only. In [6] it was demonstrated that single stuck-at fault detection can be achieved with only n+5 test vectors. It was shown in [7] that 2n+s+3 test vectors are required for single stuck-at fault detections in GRM / ESOP circuits while 2n+s vectors are required for detection of AND/OR-bridging faults in such circuits , where s is the number of product terms in the logic function. Here too, the test set is not universal as it depends on s, the number of product terms of the function. References [8], [9] proved that a test sequence of length 2n+8 vectors is sufficient to detect all single stuck-at and bridging faults.

Two new methods, each with a small modification in this scheme with ESOP RMC circuits had been proposed for analysis and diagnosis of single stuck-at faults [10], [11]. In [12],[13],[14], it was demonstrated how the RMC forms help in the detection of various digital faults and how to determine the best polarity among them. It was proved that test vectors for multiple fault detection and diagnosis in digital circuits could be generated using Neural Network with different training algorithms [15]. Reference [16] proposed a new test pattern generation algorithm using Neural Network which requires additional gates. The analysis and diagnosis of OR-bridging faults in any of the pairs of data and control lines and OR-bridging faults including intermediate gate outputs of the ESOP RMC circuits was proposed in [17], [18]. This paper proposes a modified structure that shows the testability improvement in the analysis and diagnosis of Single stuck-at, Double stuck-at, AND-bridging and OR-bridging faults including the intermediate gate outputs of the ESOP RMC circuits with minimal test vectors.

MATERIALS AND METHODS

NETWORK STRUCTURES

Reference Structure:

The network structure of the scheme is the same as that proposed in [6] and is shown in Fig. 1. It comprises literal complementing XOR block, an AND block, an XOR function tree block, which implements the required logic function as also two additional outputs O_1 and O_2 obtained through a separate AND gate and an OR gate. The actual data inputs to the system are x_1, x_2 …. x_n. Additionally, the scheme requires four control inputs c_1 to c_4. The literal-complementing block uses c_1 to produce the complements of the literals used in the function. Only those literals appearing in complemented form require an XOR gate in this block.

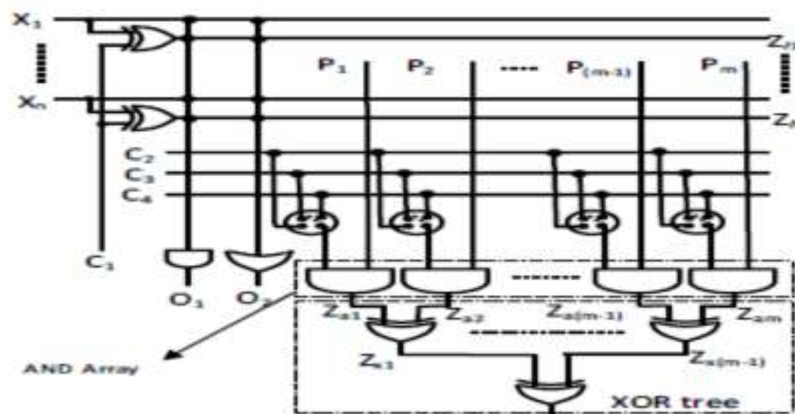

Fig.. 1. Generalized Network Structure (Reference)

The literals of each product term $P_1, P_2, \ldots P_m$ are combined through an AND gate and hence the number of AND gates required is the same as the number of product terms in the logic function. Further, each of the AND gates of this block has an additional input from one of the control lines depending on the number of gates used in the XOR tree block producing the final function F. Finally, all the data and complementary gate outputs are applied to a separate AND gate and an OR gate, producing auxiliary outputs O_1 and O_2, to aid in the detection of faults which cannot be differentiated by the main function output F alone.

Proposed Structure:

The network structure of the proposed scheme is shown in Fig. 2. It consists of an AND block, an XOR function tree block, which implements the required logic function as also one additional output O obtained through a separate XOR gate. The inputs to the system are $x_1, x_2 \ldots x_n$ the data variables and $zl_1, zl_2, \ldots, zl_m$ corresponding to the actual complementary variables available in the function. Additionally, the scheme requires three control inputs c_1 to c_3 that are connected to the AND gates present in the system as explained below.

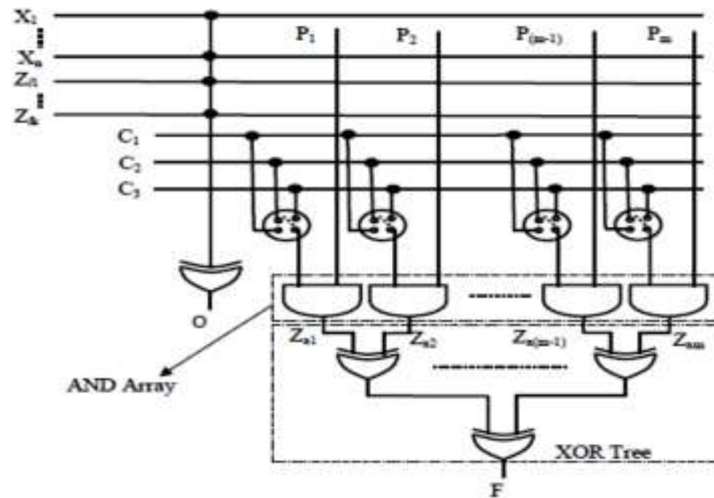

Fig. 2. Generalized Network Structure (Proposed)

Control Inputs:

The required control lines are C_1, C_2, C_3, C_4 for the Reference structure and C_1, C_2, C_3 for the Proposed structure. Draw the XOR gate tree (Fig. 3) for the product terms of the given function. Assign the numerals 1, 2 and 3 respectively to the two inputs and the output of the final XOR gate producing the function output F. Consider each XOR gate connected to the inputs of the final XOR gate. Assign the outputs of these XOR gates with the same numbers as the inputs of the final XOR gate. If the output of the XOR gate considered is 1, then assign 2 and 3 to its inputs. Else if the output is numbered 2, assign 3 and 1 to its inputs. Now consider the next earlier input stage and assign the numerals in the similar manner according to the output points connected. Connect the control lines corresponding to the lines at the first stage.

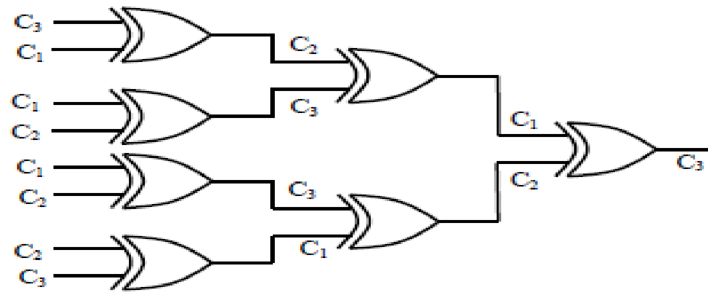

Fig. 3 Control Input Determination

GENERALISED TEST VECTORS

Reference Structure:

The test set has (n+5) vectors; each of the vectors is (n+4) long, ‘n’ being the number of data inputs. The first four columns of the matrix represent the control inputs c_1 to c_4 while the remaining n columns that of the data inputs are x_1 to $x_{n.}$ The generalized test set is shown in Table 1.

Proposed Structure:

The test set has (n+5) vectors; each of the vectors is (n+3+m) long, ‘n’ being the number of data inputs and $m(m \leq n)$ is the number of complementary literals present in the function. The first three columns of the matrix represent the control inputs c_1 to c_3 then the remaining n+m columns that of the data inputs x_1 to $x_{n.}$ and complementary literals zl_1 to zl_m. The generalized test set is shown in Table 2.

Table 1. Generalized Test Set for Reference Structure

C_1	C_2	C_3	C_4	x_1	x_2	..	x_n
0	0	0	0	0	0	..	0
0	0	1	1	1	1	..	1
0	1	0	1	1	1	..	1
0	1	1	1	1	1	..	1
0	1	1	1	0	1	..	1
0	1	1	1	1	0	..	1
0	1	1	1	1	1	..	1
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.
0	1	1	1	1	1	..	0
1	0	0	0	0	0	..	0

Table 2. Generalized Test Set for Reference Structure

C_1	C_2	C_3	x_1	x_2	…	x_n	zl_1	zl_2	…	zlm
0	0	0	0	0	…	0	0	0	…	0
0	1	1	1	1	…	1	1	1	…	1
1	0	1	1	1	…	1	1	1	…	1
1	1	1	1	1	…	1	1	1	…	1
1	1	1	0	1	…	1	0	1	…	1
1	1	1	1	0	…	1	1	0	…	1
1	1	1	1	1	…	1	1	1	…	1
.	.	.	.	.			.	.		.
.	.	.	.	.			.	.		.
.	.	.	.	.			.	.		.
1	1	1	1	1	…	0	1	1	…	1
0	0	0	0	0	…	0	1	1	…	1

ALGORITHM (REFERENCE AND PROPOSED STRUCTURES)

Step 1: The circuits as in Fig.1 (Reference network) and Fig. 2 (Proposed network) were set up..

Step 2: The control lines C_1 to C_4 (Reference) and C_1 to C_3 (Proposed) as already explained were connected.

Step 3: The test vectors as given in Table 1 (Reference) and Table 2 (Proposed), were applied one by one.

Step 4: For each test vector, the fault free outputs F, O_1 and O_2 (Reference) and F and O (Proposed) were observed.

Step 5: The decimal equivalents of each of the above binary output sets were determined taking the outputs for the first vector as LSBs.

Step 6: The Single stuck-at faults at the control inputs, data inputs and intermediate gate outputs were simulated

and the corresponding decimal outputs were determined.

Step 7: The set of outputs were compared with the predetermined fault-free condition outputs. If the two output sets matched exactly, the corresponding fault was considered as not identifiable or detectable. Besides this condition, if the output sets were the same but different from fault-free sets, then they were considered to be indistinguishable.

Step 8: The identifiability and distinguishability factors were calculated with reference to the total number of fault combinations..

Step 9: Steps 3 to 8 were repeated for Double stuck-at, AND-bridging and OR-bridging faults for all possible combination pairs of control inputs, data inputs and intermediate gate outputs in the network.

RESULTS AND DISCUSSIONS

The following ten random functions were considered and Single stuck-at, Double stuck-at, AND-bridging and OR-bridging faults are simulated using MATLAB coding and the Consolidated results of both the Reference network structure and the Proposed network structure are tabulated in Tables 5 to 8 respectively.

$$F_1 = x_1 \oplus x_2x_3 \oplus x_1'x_2x_3$$

$$F_2 = x_1x_2 \oplus x_2'x_3 \oplus x_3'x_4 \oplus x_1x_2x_3$$

$$F_3 = x_1' \oplus x_2x_3'x_4 \oplus x_3x_4' \oplus x_2'x_3 \oplus x_1x_4x_5$$

$$F_4 = x_1x_2' \oplus x_2x_3x_4' \oplus x_4x_5'x_6 \oplus x_2x_5 \oplus x_2'x_5' \oplus x_3'x_2x_1 \oplus x_4x_6$$

$$F_5 = x_1'x_2x_3 \oplus x_4x_5x_6 \oplus x_4'x_6'x_7 \oplus x_3x_5x_7$$

$$F_6 = x_1x_2'x_3 \oplus x_4'x_5x_6' \oplus x_7x_8' \oplus x_2x_6x_7' \oplus x_1'x_6 \oplus x_3'x_4 \oplus x_1x_5 \oplus x_4x_5' \oplus x_5x_7 \oplus x_8x_3x_1 \oplus x_3x_5'x_1$$

$$F_7 = x_1x_2'x_3' \oplus x_4x_5'x_6 \oplus x_7'x_8x_9 \oplus x_1'x_4'x_9' \oplus x_2x_5' \oplus x_3x_5$$

$$F_8 = x_1'x_2x_3' \oplus x_4'x_5'x_6 \oplus x_7x_8'x_9' \oplus x_{10} \oplus x_6'x_7 \oplus x_8x_{10}$$

$$F_9 = x_1 \oplus x_2'x_3x_4' \oplus x_5'x_6x_7' \oplus x_8x_9x_{10} \oplus x_{10}'x_{11} \oplus x_1x_3x_9$$

$$F_{10} = x_1'x_2 \oplus x_3x_4'x_5 \oplus x_6x_7'x_8x_9 \oplus x_{10}x_{11}'x_{12} \oplus x_1x_2x_3' \oplus x_4'x_7$$

NUMERICAL ILLUSTRATION:

The fault-free output sets using Reference and Proposed networks for the illustrative function $F_1 = x_1 \oplus x_2x_3 \oplus x_1'x_2x_3$ were respectively found to be $\{F, O_1, O_2\} = \{126, 112, 127\}$ and $\{F, O\} = \{126, 7\}$.

Single Stuck-at faults:

The stuck-at-0 and stuck-at-1 faults have been simulated for the given function at lines C_1, C_2, C_3, C_4, x_1, x_2, x_3, zl_1, za_1, za_2, za_3, zx_1 and zx_2 and the results are shown in the Tables 3 and 4. Here C_1, C_2, C_3 and C_4 are the control lines; x_1, x_2, x_3 are the three input lines; zl_1 is the complementary output of x_1; za_1, za_2, za_3 are the three output lines of the AND gates while zx_1 and zx_2 are the two Ex-OR tree gate outputs.

Table 3. Stuck-at-0 outputs for function F_1

Fault line	C_1	C_2	C_3	C_4	x_1	x_2	x_3	zl_1	za_1	za_2	za_3	zx_1	zx_2
F	126	126	120	6	120	86	86	46	40	6	46	80	0
O_1	112	112	112	112	0	0	0	0	112	112	112	112	112
O2	126	127	127	127	127	127	127	126	127	127	127	127	127

Table 4. Stuck-at-1 outputs for function F_1

Fault line	C_1	C_2	C_3	C_4	x_1	x_2	x_3	zl_1	za_1	za_2	za_3	zx_1	zx_2
F	38	126	126	126	126	126	126	118	215	249	209	175	255
O_1	0	112	112	112	120	116	114	112	112	112	112	112	112
O2	255	127	127	127	255	255	255	255	127	127	127	127	127

For the stuck-at-0 fault at C_2 as well as for the stuck at-1 fault at C_2, C_3 and C_4, the output sets obtained are the same as that of the fault-free set. Hence, these faults are unidentifiable. The identifiability factor is (26-4)/26 *100 = 84.62%.

Also, in stuck-at-0 fault, the output set {6, 112, 127} is repeated two times for C_4 and za_2 and {86, 0, 127} is repeated two times for x_2 and x_3 but different from fault free values. These faults are detectable but indistinguishable within the same subsets. Thus the overall distinguishability factor for this function is (26-4) / 26 *100= 84.62%. However, if only one output set namely {6,112,127} is considered, then the distinguishability factor for this set is (26-2)/26 *100 = 92.31%, which is higher than the overall factor.

The consolidated simulation results of Single-stuck-at faults for all the ten random functions F_1 to F_{10} with the Reference and Proposed network structures are tabulated in Table 5. It shows that the distinguishability factor had improved by 10% using the proposed structure. It can be further inferred that Identifiability and Distinguishability factors are better when the number of variables are more. The observability is better using a single auxiliary output O using XOR gate for the proposed structure than the reference structure which uses two auxiliary outputs O_1 and O_2 using AND and OR gates. Further, the location of fault can also be easily diagnosed from the output set. For instance if the output set is {6,112,127} then the fault condition would be only one of the two cases involving C_4 or za_2 as given in Table 3 and hence those lines only need to be checked.

Table 5. Consolidated Simulation Results for Single Stuck-at Faults

S.No.	Function	No. of data Inputs	Total Possible Faults	Reference Structure		Proposed Structure	
				% ID factor	% Dist. factor	% ID factor	% Dist. factor
1	F_1	3	26	84.62	84.62	83.33	91.67
2	F_2	4	34	97.06	82.35	96.88	87.50
3	F_3	5	44	95.45	95.45	95.24	95.24
4	F_4	6	54	96.30	92.59	96.15	96.15
5	F_5	7	42	97.62	73.81	97.50	90.00
6	F_6	8	82	96.34	97.56	96.25	97.50
7	F_7	9	62	98.39	83.87	98.33	96.67
8	F_8	10	64	98.44	78.13	98.39	96.77
9	F_9	11	62	98.39	79.03	98.33	96.67
10	F_{10}	12	64	98.44	78.13	98.39	96.77
Average				96.11	84.55	95.88	94.49

Double Stuck-at faults:

Double Stuck-at faults can also occur quite frequently. In this case, exactly two lines are faulted, though the two lines can be any of the input/output or intermediate lines. The network structure and test vectors are the same as those for the single stuck-at fault. However, in the test procedure, two lines at a time are considered and made to be stuck-at-0 or stuck-at-1 and simulated. Since two lines are involved, four combinations, viz. (0,0), (0,1), (1,0) and (1,1) are possible for each pair of lines. For instance, four fault combinations exist for the pair of lines {c1,c2} as {0,0}, {0,1}, {1,0} 1nd {1,1}. Hence, the total number of fault combinations are much higher than the single fault case.

The consolidated simulation results for double stuck-at faults for all the ten random functions considered are given in Table 6.

Table 6. Consolidated Simulation Results for Double Stuck-at Faults

S.No.	Function	No. of data Inputs	Total Possible Faults	Reference Structure		Proposed Structure	
				% ID factor	% Dist. factor	% ID factor	% Dist. factor
1	F_1	3	312	98.40	30.13	98.11	32.20
2	F_2	4	544	100	31.25	100	42.29
3	F_3	5	924	99.89	33.98	99.88	55.12
4	F_4	6	1404	99.93	33.26	99.92	52.23
5	F_5	7	840	100	29.64	100	55.79
6	F_6	8	3280	100	32.50	99.90	59.46
7	F_7	9	1860	100	32.80	100	67.30
8	F_8	10	1984	100	31.50	100	69.14
9	F_9	11	1860	100	33.12	100	68.45
10	F_{10}	12	1984	100	31.10	100	69.14
Average				99.82	31.93	99.78	57.11

AND-Bridging faults

The bridging faults are considered as a special case of multiple faults. In an AND-bridging fault, all the lines involved in the fault have the same logic value equal to the logical AND of their pre-fault values. In this paper only two lines are assumed to be bridged at a time. The total number of faults correspond to the number of two line combinations out of the total input and output lines.

The simulation results for the AND-bridging type of faults for all the ten random functions considered are given in Table 7.

Table 7. Consolidated Simulation Results for AND-Bridging Faults

S.No.	Function	No. of Data Inputs	Total Possible Faults	Reference Structure		Proposed Structure	
				% ID factor	% Dist. factor	% ID factor	% Dist. Factor
1	F_1	3	78	85.90	47.44	87.88	57.58
2	F_2	4	136	95.59	36.76	94.17	76.67
3	F_3	5	231	89.18	47.62	95.71	76.67
4	F_4	6	351	90.88	52.42	97.54	78.77
5	F_5	7	210	86.19	48.10	93.16	81.58
6	F_6	8	820	91.59	58.66	97.69	85.51

7	F_7	9	465	91.40	47.74	97.01	83.22
8	F_8	10	496	90.52	33.06	97.20	83.87
9	F_9	11	465	90.75	38.49	96.09	86.90
10	F_{10}	12	496	89.11	47.98	95.48	86.88
Average				90.11	45.83	95.19	79.77

OR-Bridging faults

The OR-bridging fault is similar to the AND-bridging type. The difference is that, the post-fault values of all the lines involved would be equal to the logical-OR value of the pre-fault values. Only two lines were considered to be faulted at a time.

The results of the simulation for the OR-bridging type of faults for all the ten random functions considered are given in Table 8.

Table 8. Consolidated Simulation Results for OR-Bridging Faults

S.No.	Function	No. of Data Inputs	Total Possible Faults	Reference Structure		Proposed Structure	
				% ID factor	% Dist. factor	% ID factor	% Dist. factor
1	F_1	3	78	84.62	58.97	84.85	59.09
2	F_2	4	136	98.53	52.21	100	78.33
3	F_3	5	231	96.54	59.74	98.10	64.29
4	F_4	6	351	96.58	67.24	99.08	70.46
5	F_5	7	210	98.57	85.71	100	78.42
6	F_6	8	820	98.29	70.12	99.62	74.49
7	F_7	9	465	98.28	79.14	100	76.55
8	F_8	10	496	98.59	85.89	100	86.67
9	F_9	11	465	98.71	86.88	99.77	88.74
10	F_{10}	12	496	98.79	80.85	100	80.43
Average				96.75	72.68	98.14	75.75

CONCLUSION

A proposed test set scheme for detection of Single stuck-at, Double stuck-at, AND-bridging and OR-bridging faults for ESOP RMC logic functions have been detailed and the simulation results are shown in comparison with the reference method. The results conclude that n+5 test vectors are sufficient to detect the four different types of faults in digital circuits. Further, the location of fault can also be diagnosed through the output sets. The analysis and diagnosis have been done through compact tabulation and two quantification indices. All possible combinations of the data lines, control lines and all intermediate gate outputs line pairs have been considered. The overall identifiability factor for all the four types of faults was above 95% with a single network structure. It was also observed that the overall distinguishabililty factor has improved in the range of 57-94%. The individual set distinguishability factor was more

than 95% and much more than the overall distinguishability factor as already explained. With the proposed structure even the overall distinguishability factor has improved by 25% when compared to the reference structure.

REFERENCES:

[1] Reddy S.M., "Easily Testable Realizations for Logic Functions", IEEE Trans. on Computers, Vol. 21, No. 11, pp. 1183- 1188, Nov. 1972.
[2] Pradhan D.K., "Universal Test Sets for Multiple Fault Detection in AND-EXOR Arrays", IEEE Trans. on Computers, Vol. 27, No. 2, pp. 181-187, Feb. 1978.
[3] Bhattacharya B.B., Gupta B., Sarkar S and Choudhury A.K. "Testable design of RMC networks with Universal Tests for Detecting Stuck-at and Bridging Faults", IEE Proc. Computers and Digital Techniques, Vol.132, Part E., No.3, pp. 155-162, May 1985.
[4] Sasao, T., "Easily Testable Realization for Reed-Muller Expressions", IEEE Trans. on Computers, Vol. 46, No. 6, pp. 709-716, Jun. 1997.
[5] Kalay U, Hall D.V. and Petrowski M.A., "A Minimal Universal Test Set for Self-Test of EXOR-Sum-of-Products Circuits", IEEE Trans. on Computers, Vol. 49, No. 3, pp. 267-276, Mar. 2000.
[6] Zhongliang P., "Testable Realizations of ESOP Expressions of Logic Functions", Proc. of 11th Asian Test Symposium (ATS"02), IEEE Computer Society, pp 140-144, 2002.
[7] Zhongliang P.,"Bridging Fault Detections for Testable Realizations of Logic Functions", Proc. of 16th International Conference on VLSI Design, pp. 423-427, Jan. 2003..
[8] Hafizur Rahaman, Debesh K.Dass, "Bridging Fault Detection in Double Fixed-Polarity Reed-Muller (DFPRM) PLA", IEEE Proceedings Asia and South Pacific Design Automation Conference (ASP-DAC'05), pp 172-177, 2005.
[9] Hafizur Rahaman, Debesh K.Dass, "Universal Test Set for Detecting Stuck-at and Bridging Faults in Double Fixed-Polarity Reed-Muller Programmable Logic Arrays", IEE Proc. Comput. Digit. Tech., Vol. 153, No. 2, pp. 109-116, March 2006.
[10] Neelakantan P.N. and Ebenezer Jeyakumar A., "Stuck-at Fault Test Vectors for Exclusive-OR Sum Reed-Muller Canonical Boolean Functions", GESTS International Transactions on Computer Science and Engineering, Vol. 27, No. 1, pp. 139-147, Jan. 2006.
[11] Neelakantan P.N. and Jeyakumar A. E., "Single Stuck-At Fault Diagnosing Circuit of Reed-Muller Canonical Exclusive-Or Sum of Product Boolean Expressions", Journal of Computer Science, USA,Vol. 2,Issue 7, pp. 595-599, July 2006.
[12] Pan Zhongliang and Chen Guangju, "Fault Detection Test Set for Testable Realizations of Logic Functions with ESOP Expressions", Journal of Electronics (China), Vol. 24(2), pp. 238-244, 2007.
[13] Ling Chen, and Zhong Liang Pan, "Fault Detection of Bridging Faults in Digital Circuits by Shared Binary Decision Diagram", Key Engineering Materials, Tans Tech Publications (Volumes 439 - 440), pp. 1235-1240, June 2010.
[14] Wenjin Wu, Pengjun Wang, Xiaoying Zhang, Lingli Wang and Dai Jing "Search for the Best Polarity of Multi-Output RM Circuits Base on QGA", Second International Symposium on Intelligent Information Technology Application, Vol. 3, pp. 279 – 282, 2008.
[15] Zhong Liang Pan, Ling Chen and Guang Zhao Zhang, "Test Pattern Generation of VLSI Circuits Using Hopfield Neural Networks", Applied Mechanics and Materials, Trans Tech Publications (Volumes 29 - 32), pp. 1034-1039, August, 2010.
[16] Zhao Ying and Li Yanjuan, "A Multiple Faults Test Generation Algorithm Based on Neural Networks and Chaotic Searching For Digital Circuits", International Conference on Computational Intelligence and Software Engineering (CISE), December 2010.
[17] Geetha V., Devarajan N. and Neelakantan P. N., "OR-Bridging Fault Identification and Diagnosis for Exclusive-OR Sum of Products Reed-Muller Canonical Circuits", Journal of Computer Science, USA, Issue 7(5), pp. 744-748, July 2011.
[18] Geetha V., Devarajan N. and Neelakantan P. N., "OR-Bridging Fault Analysis and Diagnosis for Exclusive-OR Sum of Products Reed-Muller Canonical Circuits", European Journal of Scientific Research, USA, Issue 71, No. 4, pp. 482-489, March 2012.