Analysis of an Efficient Approach to Tolerate Soft Errors in Combinational Circuits

Manuj P.[1]****, K. B. Ramesh***[2]
[1]*UG student, Department of Electronics and Instrumentation Engineering,*
RV College of Engineering, Bengaluru, India
[2]*Associate Professor, Department of Electronics and Instrumentation Engineering,*
RV College of Engineering, Bengaluru,India
****Corresponding Author***
***E-Mail Id**: manujprakash2002@gmail.com*

ABSTRACT

In recent years, soft errors happen in the combinational logic circuits that genuinely impact the action of digital frameworks. Hence, the soft error has ended up a matter of great concern for unwavering quality issues at show. To extend the soft error tolerance, this paper proposes a modern method that will diminish the failure rate of the combinational circuits. A method has been introduced which is able discover out the foremost common minterms of Boolean polynomial math. And these minterms are secured by different cubes to maximize the logical veiling probability, which reduces the failure rate of a combinational circuit. The experimental study appears how the probability of circuit failure is diminished for a given combinational circuit.

Keywords: *Soft errors, combinational circuits, reliability, logical masking, failure probability*

INTRODUCTION

For the past few decades, a soft error has been a matter of great concern for combinational circuits as well as for computer frameworks. Day by day, the circuit is getting so smaller than the thickness it is expanding. As of now, the thickness has become around 1012 devices /cm2.

This advance is consistent with Moore's law. Innovation has presently reached the nanometer scale. The littler the circuit, the more sensitive it gets to be too diverse situations. In most cases, radiating particles combined with gadget bundling are causing soft errors.

In case these particles hit any crucial portion of a circuit, at that point the yield of the gadget may be off-base. For which, the entire circuit or gadget may work inaccurately.[1]

Research is always being wiped out distinctive ways for the reliability of the circuit and to moderate soft errors. To toleratesoft errors, diverse techniques have been introduced. Among these strategies, veiling impacts are pondered in 4, there estimation SER handle is appeared in 5, and implicitly don't care is utilized in an iterative method in 6. Other than these, the combinational circuit's unwavering quality can be evaluated by the sub-circuits mistake probability 7.

A probabilistic gate model (PGM), which could be a computational method, is utilized to evaluate the circuit's unwavering quality 8. In 9, the proposed method endures soft errors at the gadget level and centre logic level. Another Error Location Instrument (EDM) is appeared in which primarily works with information and code to diminish soft errors.

In spite of the fact that diverse strategies exist within the writing, the existing methods are confronting time/ area overhead, tall power consumption, and intemperate taken a toll in common cases.[2]

Hence, researchers are focusing on alternate techniques for efficient. For a Boolean work, our proposed method finds the easiest conceivable solutions based on covering cubes or common minterms among cubes. A scientific condition is used to compute the failure rate of each solution. The minimal failure rate is picked over diverse options to return soft error tolerant solutions since the lower failure rate represents the conceivable higher soft error tolerant solution. We can illustrate the concept of the proposed method utilizing an example to form our thought less demanding to get it. In Fig. 1, we can watch that there are three ways to reach the objective from the source.

The primary course is 1.3 km with the probability of failure, 0.36, the moment course is 1 km with the probability of failure 0.28 third courses are 1.3 km with the probability of failure, 0.36. Among these three ways, the moment way is better in judging the probability of failure rate and the remove of the path.[3]

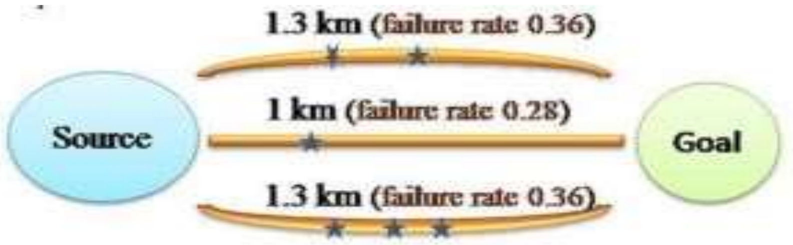


Fig. 1: *A Different Way from Source to the Destination.*

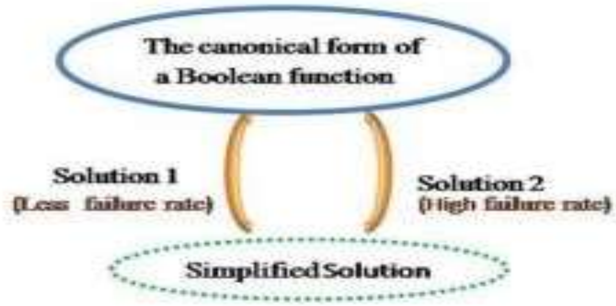


Fig. 2: *Different Simplified Solution Way for the Canonical form of a Boolean Function.*

Presently, within the same way, we will think approximately determining the solutions for the combinational circuits. The concept in this respect is outlined in Fig. 2. Assume the canonical shape of a Boolean work is

F(A, B,C,D) = ∑m (10,12,14) +∑d (4,11). It has two simplified solutions. Our primary point is to consider which solution isrearranged and features a higher soft error resilience capacity. In this respect, Solution 1 is more preferable to Solution 2.

The oddity of this paper can be summarized as follows. This paper returns the disentangled solution of a combinational circuit plan issue in lesser time and dodges 16% circuit failure which reduces the soft mistake failure rate compared to others. The method utilized in minimizing the soft error failure rate is novel as well and it spares a noteworthy amount of time concerning existing methods.[4,5]

The paper is organized as takes after. Segment II presents a study on existing research in this zone. The proposed method is discussed in area III. Area IV portrays the experimental analysis. As a last point, Area V contains the concluding statements.

RELATED WORK

There are several types of research, existing in the literature, to tolerate soft errors. A few of these researches are outlinedshortly as follows.

Triple Measured Repetition (TMR) is habitually utilized to enhance the execution of the circuit's soft mistake tolerance.

TMR comprises three for all intents and purposes vague adapts of the beginning circuit that energizes two out of three majority voters. TMR causes tall overhead concerning range and power (over 200%). Regular TMR is as it were workable for the sequential circuit but not workable for combinational circuits Space-Time TMR (ST-TMR) and Improved ST-TMR (ESTTMR) strategies are illustrated which are compelling for both combinational and consecutive circuits.

Be that as it may, TMR increases the estimate of capacity cells which are for the most part undesired in a few frameworks particularly within the case of implanted systems. In, soft mistake resistance with negligible locale overhead has been appeared. Be that as it may, these approaches consider as it were size based methods and ignored other procedures in this regard.

The objective of the synthesis-based approach is to maximize the veiling substance. In, a method that can re-contract the substantial logic has appeared. Two strategies are utilized to recoup unwavering quality: do not care-based re-synthesis and local rewriting. Be that as it may, in this approach, huge ranges are utilized to update littler sub-circuits to create advance. Additionally, weak scalability and longer runtime prerequisite debase the performance of this method.

A solution of wiring associations and evacuations are performed by finding futile wires within the circuit in. This technique offers a monotonous structure to keep wires and gates with ensuing secret impacts and tries to oust wires and entryways with higher misalignment.

Kwon *et al.* presented a mistake examination procedure for both combinational and successive logic circuits. It takes all possible ways from one gate to another and computes the logical probability. In this approach, for each entryway, a look-up table has been considered to assess the transitory beats at the starting yields.

Michels *et al.* have come up with a thought to tolerate soft errors which are inalienably passable for the circuit. The approach uses analog majority gates. However, it cannotcorrect all possible soft errors.

PROPOSED METHODOLOGY

In this segment, we have displayed a novel method for enhancing the soft mistake resistance of combinational circuits. This procedure decides the leading cube set to wrap all minterms to maximize the probability of logical concealing and return the error rate of combinational circuits.

A Boolean function with minterms and do not care conditions is provided in this approach. At that point the work is passed into K-map or tabulation strategies to return conceivable rearranged solutions. In the K-map or solution method, minterms that are common in more than one cube are created.

For each era, the error rate is calculated which demonstrates the soft error tolerance of that solution. A lower mistake rate gives higher soft error tolerance and bad habit versa.

Thus, the cubes returned by the proposed method will lead to a solution with the next soft error resistance rate among all conceivable disentangled solutions generated by the K-Map or Organization method. Fig. 3, illustrates the stream chart of the proposed method.

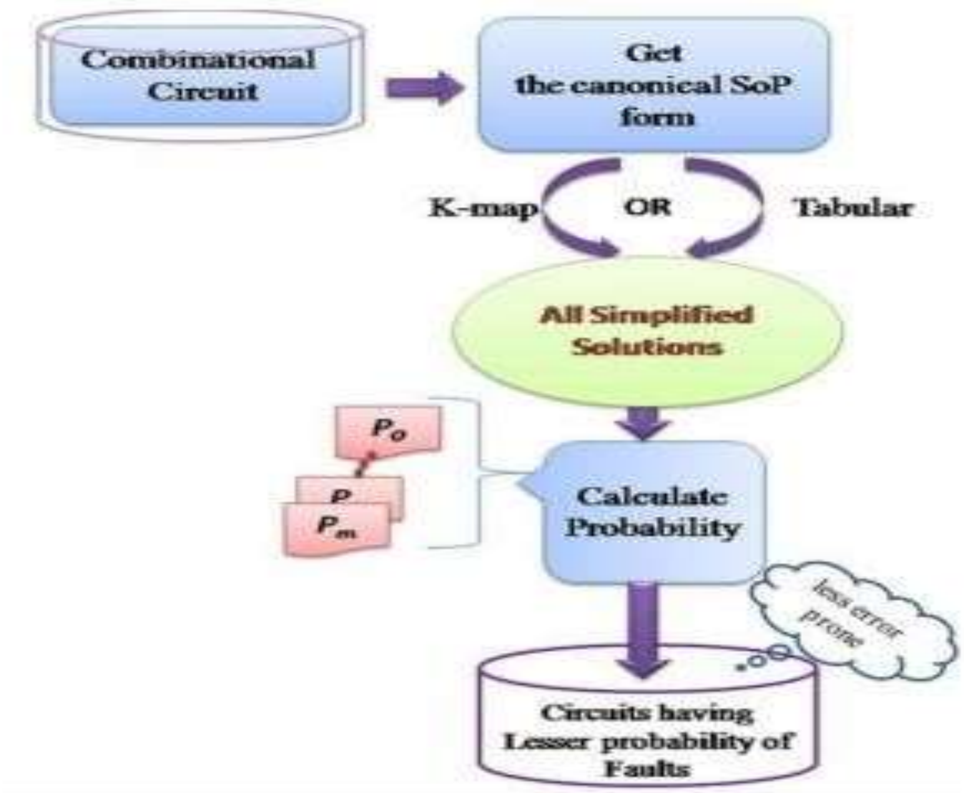


Fig. 3: *Flow Chart of the Proposed Method.*

Finding Probabilities of Minterms
A Boolean function

F(x_0, x_1, x_2, … . , x_n) has the features: P(x_i=1) = P($\boldsymbol{x_i}$ = 0) = 1/2and P(x_i.) != P(x_i) . P(x_j) for i≠ j, the output

probability is equal to the number of the minterms of the corresponding Boolean function divided by 2^n , where n is the number of variables .

Let us consider that the Boolean function has k minterms denoted by

$m_0, m_1, m_2, \ldots . . m_{k-1}$with n variables. Then probability of each minterm will be P(m_0) = P(m_1) =…….= P(m_{k-1}) =$^1/_{2n}$.

Failure Rate Calculation
The probability of each simplified solution can be estimated by using (1).

$$\text{Estimated Failure Rate} = \frac{\Sigma(P_m x F_m)}{\Sigma(Failure\ Rate\ of\ Individual\ Gate)}. \quad \ldots\ldots..(1)$$

Where $P_m$is the probability of a minterm, and $F_m$is the probability of common minterms. On the off chance that the Boolean work has n factors at that point the probability of each minterm is Pm = $\frac{1}{2}$.

The evaluated failure rate of a combinational circuit depends on the probability of a minterm and the probability of common minterms.

The input design probability is rise to the probability of minterms that's common in one or more groups. In the event that a minterm is common in two groups, at that point for that minterm the probability ought to be F_m= P_m/ 2 because it is common in two cubes.

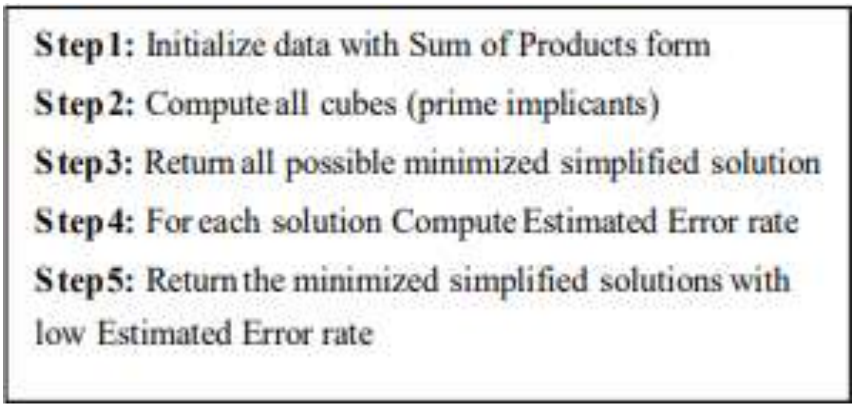
Essentially, in the event that it is 3 the at that point probability ought to be F_m= P_m/ 3, for 4 it would be F_m= P_m/ 4, and so on.

But, on the off chance that don't care minterm is common between two or more groupes, its probability will be comparative as Pm. For those minterms which can make as it were one group (i.e., exists in one group), in this case, F_m ought to be P_m

The Process of Estimating Error Rate

The method of assessing the error rate is depicted by using a calculation as appeared in Fig. 4. In Step 1, we have initialized information; ready to think of the input pattern as Ix. In Step 2, the cube is graphically coordinated with Gx.

At that point within the next step (Step 3), we are going discover all conceivable combinations cx of group Gx in such a way that all input Ix are secured. In Step 4, for each disentangled conceivable solutioncx, we ought to calculate the probability Pm and Fm for each related input term. After that, we have summarized the assessed error rate for every solution cx by utilizing (1). Step 5, will return solution cx with a low evaluated error rate.



Step1: Initialize data with Sum of Products form

Step2: Compute all cubes (prime implicants)

Step3: Return all possible minimized simplified solution

Step4: For each solution Compute Estimated Error rate

Step5: Return the minimized simplified solutions with low Estimated Error rate

Fig. 4: *The Process Step of Estimating the Error Rate.*

To demonstrate this handle, consider that we have some primary input designs: I_0, I_1, I_2, I_3, $I_4$and I_5; created two outputs are C_0, and C_1; four groupes of input designs are G_0 , G_1 , G_2 , and G_3 . We ought to keep up the component of the group in 2^norder, where n may be a positive numbers. Fig. 5, appears the graphical outline of the inputs and yields.

The four groups are shaped in such a way that all inputs are secured.

Group G_0 and G_2 are signified by a ruddy specked range and these two groups give an yield solution C_0 for their input patterns. Fig. 6, speaks to the yield solution C_0. In Fig. 7, another yield solution C_1 is appeared, which is gotten from three groups G_0, $G_1$and G_3.

These three groups are indicated by a sky blue dabbed region. From Fig. 7, we are able watch that I_0 is common in two groups G_0, and G_1. Other inputs have a place to different groups. Be that as it may, in Fig. 4, no input is common in the shown two groups. Subsequently, for input I_0, in case there's a soft error in G_0 at that point C_1 will return the yield for this but C_2 will not return the yield.

Expect a combinational logic circuit that can be generated by four factors A, B, C, D and it has six prime ensnares : - 100 (BC'D'), 000- (A'B'C'), 11-0 (ABD'), 010- (A'BC'), 1-10 (ACD'), and 101-(AB'C). A least of two implicants is required to cover ON-minterms.

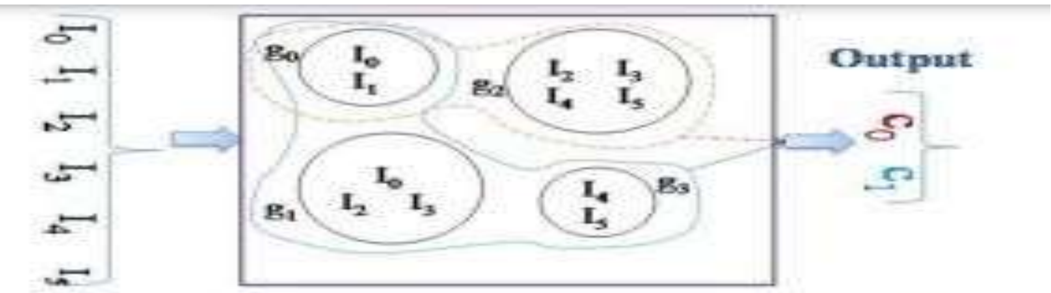


Fig. 5: *Graphical Illustration of the Inputs and Outputs.*

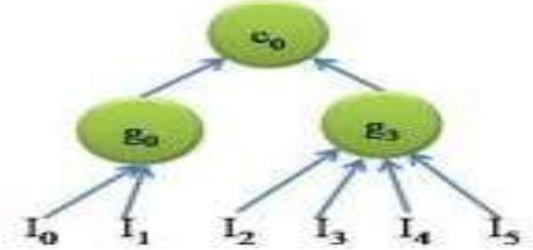


Fig. 6: *Output Solutions C_0.*

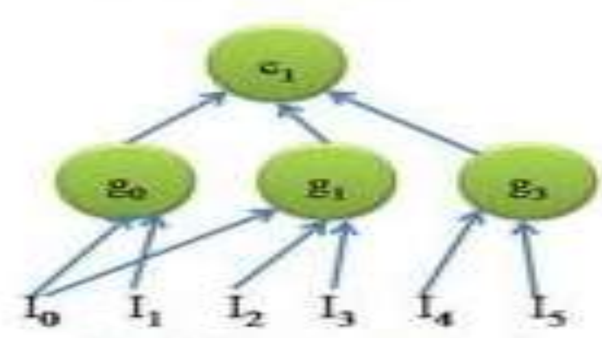


Fig. 7: *Output Solution C_1.*

For this example, three possible solutions will cover all ON minterms. Fig. 8, is the K-map representation of a given example of four input patterns. Don't care conditions are represented by X, while ON-minterms are represented by 1. Table

1, demonstrates the coverage of ON-minterms by arranging the cubes. The first row indicates that minterm 1100 is covered by two cubes 11-0 and -100. Similarly, the remaining rows illustrate the presence of other minterms in the shown five cubes. The probability of occurrence for each minterm in the four variables' K-map is 1/24 =0.0625.

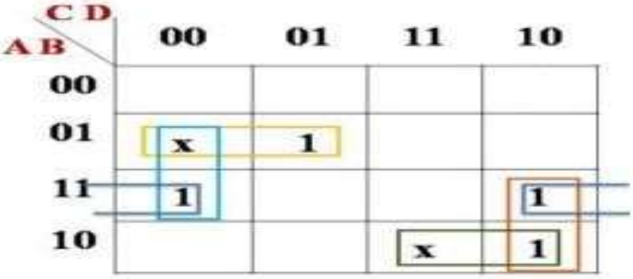
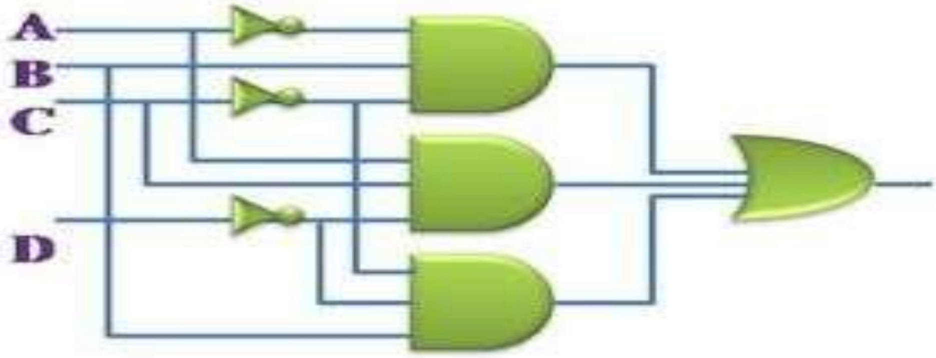


Fig. 8: *K-Map Representation for the Given Example.*

Minterm	*010-*	*1-10*	*11-0*	*-100*	*101-*
1100			1	1	
0101	1				
1110		1	1		
1010		1			1

Table 1: *Cube Covering Example.*

Indeed in spite of the fact that there are three potential solutions, there's a distinction between them. Considering the primary utilization with three cubes-100, 1-10, and 010-, each minterm is gotten by just one shape. As a result, for each minterm, in case a mistake occurs at the logic gate for a minterm, no other cube will cover it, and the error will amplify to the yield. The moment one employments three implicants (cubes) 11-0, 1-10, and 010-. Here the minterm 1100, and 1010 are gotten by one and another minterm 1110 is encased in two cubes. In case 1110 hits an error by reasoning, another will cover it within the third usage with three cubes010-, 11-0, and 101-, which is comparative to the primary one.



(a)

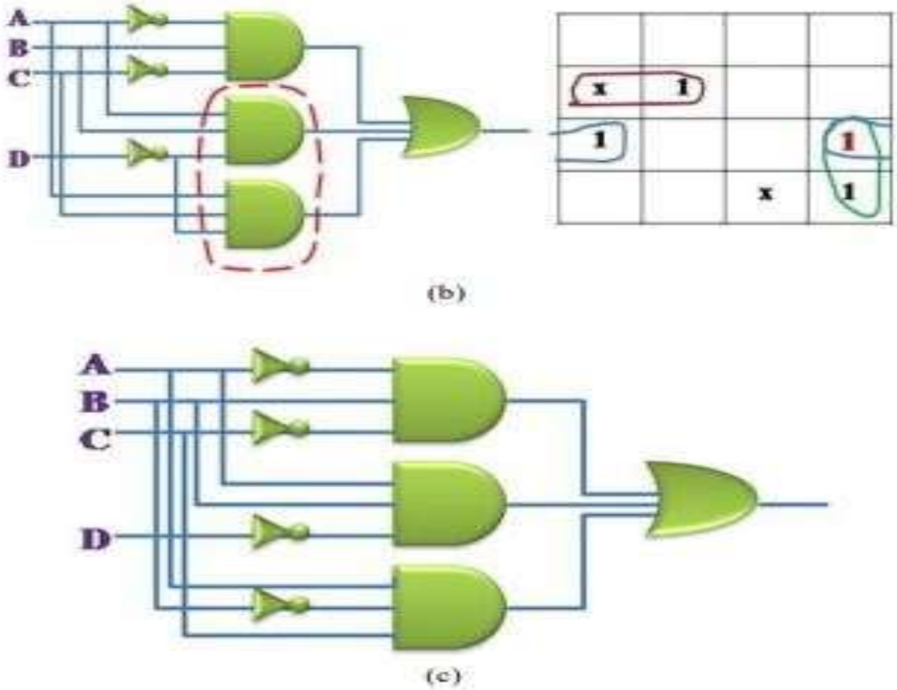


***Fig. 9:** Different Types of Implementations for the given Example: (a) Implementation q: Logical Circuit of Cubes -100, 1-10, and 010- is called, (b) Implementation 2: On the Left Side Logical Circuit Diagram of Cubes 11-0, 1-10, and 010-, On the right Side we are presenting the K-Map Representation, (c) Implementation 3: Logical Circuit Diagram of Cubes 010-, 11-0, and 101-.*

In Fig. 9(a), 9(b) and 9(c) demonstrate the logic circuits for implementation 1, 2, and 3 respectively. Implementation 1 in Fig. 9(a), is the logical circuit diagram of cubes -100, 1-10, and 010-, where every minterm is enclosed to one implicant (cube). Because of this, if there is an error in the logic gate it will propagate to the output. No other cube will mask this. Using (1), the estimated failure rate for a logic circuit (a) is 0.268.

Usage 2 is in Fig. 9(b), on the cleared out side logical circuit chart of cubes11-0, 1-10, and 010-. On the right side, we are showing the K-Map representation. In implementation 2, minterm 1100 and 1010 are encased in one implicant called a cube, another minterm 1110 is enclosed in two cubes. In Fig. 9 (b), ON-minterms are shown by red colour in K-map and related circuits that encase those ON minterms are spoken to by the ruddy specked zone of the circuit. In this circuit, on the off chance that there's a mistake in any entryway for the minterm 1110, another gate will veil it. The evaluated failure rate for the logic circuit (b) is 0.231, which is calculated by (1).

In Fig. 9(c), Usage 3 speaks to the consistent circuit diagram of cubes 010-, 11-0, and 101-. In usage 3, every minterm is encased in one implicant (cube). Hence, logic mistakes will not be conceal in this execution as it does for usage 2. The assessed failure rate for the logic circuit (c) is 0.268, which is calculated by (1).

For the sake of understanding, we can say that the canonical form of a Boolean function is F(A, B, C, D) = ∑m(5, 10,12,14) +∑d (4,11) has three simplified solutions. It means that for this function, we can generate three combinational logic circuits. The Illustrated form of this case is shown in Fig. 10. Among these three solutions, the error rate of A ' BC ' +ABD

'+AC D ' (Solution 1) is lower. This error rate is calculated using (1). The estimated failure rate for A ' BC ' +ABD ' +AC D ' is 0.0049 and this failure rate for other two solutions A'BC'+ACD'+BC'D' and A'BC'+ABD'+AB'C are 0.0059.

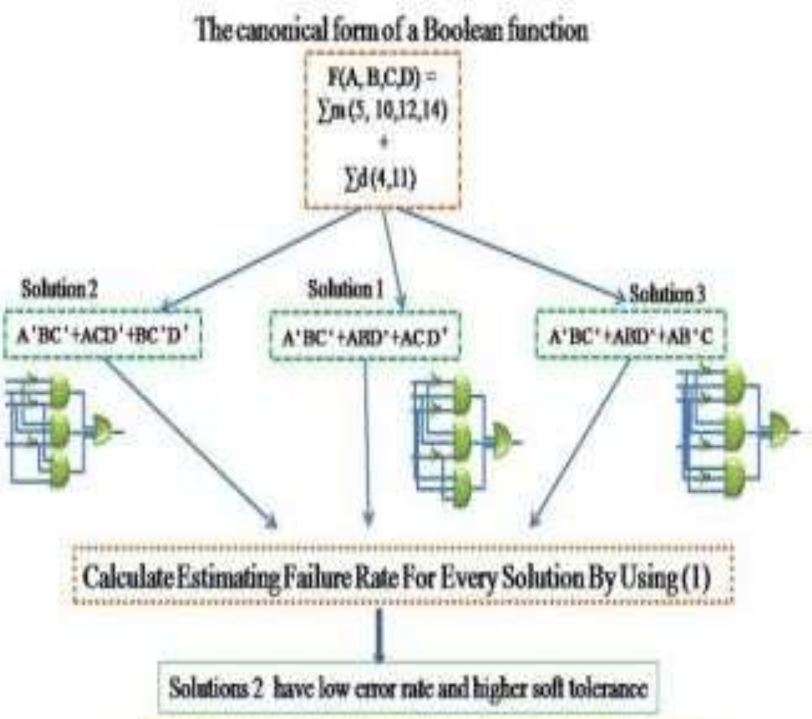

The canonical form of a Boolean function
F(A, B,C,D) =
∑m (5, 10,12,14)
+
∑d (4,11)
Solution 2
A'BC'+ACD'+BC'D'
Solution 1
A'BC'+ABD'+ACD'
Solution 3
A'BC'+ABD'+AB'C
Calculate Estimating Failure Rate For Every Solution By Using (1)
Solutions 2 have low error rate and higher soft tolerance

Fig. 10: *An Illustrated form of the Given Example.*

EXPERIMANTAL ANALYSIS

In this area, we have taken diverse Boolean functions and at last calculated the failure rate of each implementation. Then, we made a comparison among them and draw a final recommendation to return soft error tolerant solutions for a Boolean work.

Experimental Setup

We have used the following setup to implement our logic step to calculate the approximate error rate of a given Boolean function.

- Processor: Intel CORETMi3-7100M CPU @ 3.90 GHz
- Operating System: Windows 8
- RAM:4GB
- System Type: 64-bit Operating System

Experimental Results

The mistake rates for distinctive usage of a Boolean function are analysed and a comparison among them is drawn. In Fig. 11, we have taken six distinctive Boolean capacities to show the adequacy of the proposed method.

A canonical form of a combinational circuit is F(1) =∑m (5, 12, 14, 10) + ∑ d (4, 11) which has three simplified solutions. F(1) is already outlined in Fig. 10. The three solutions of F (1) are: A'BC'+ABD'+ACD' (Solution 1), A'BC'+ACD'+BC'D' (Solution 2) and A'BC' +ABD' +AB'C (Solution 3). F(1) has five scope cube. It may be a 4 factors work. The probability of each minterm is P_m= 1/24=0.0625. Solution 1 has a common minterm 1110, which is covered by two cubes: 1-10 and 11-0. The probability of minterm 1110 is F14 = 0.0625/2 = 0.0312. There is no common minterm for Solution 2 and Solution 3.Hence,

]

$$\text{Estimated Failure Rate for Solution 1} = \frac{0.06252 + 0.06252 + 0.06252 + 0.06252 + 0.0625 * \frac{0.0625}{2}}{4} = 0.0049$$

$$\text{Estimated Failure Rate for Solution 2} = \frac{0.06252 + 0.06252 + 0.06252 + 0.06252 + 0.06252}{4} = 0.0059$$

$$\text{Estimated Failure Rate for Solution 3} = \frac{0.06252 + 0.06252 + 0.06252 + 0.06252 + 0.06252}{4} = 0.0059$$

Looking at the over comes about, it can be watched that the Estimated Failure Rate for Solution 1 is lower than others.

For this reason, we have chosen Solution1 concerning soft error tolerance. In this way, we have delivered solutions or the rest of the capacities.

Solution 1 incorporates a lower mistake rate in comparison to both solution 2 and solution 3 and it encompasses a higher soft error tolerance than the other two solutions. F(6) has three simplified solutions where solution 1 appears the best performance with a moo assessed error rate. We have used randomly diverse input capacities.

F(2) = ∑ m (4, , 7, 8, 10, 12, 14, 15) + ∑ d (5) ,
F (3) = ∑ m (4, ,7,8,12,14, 15) + ∑ d (5),
F(4) = ∑m(2,3,7,9,11,13) + ∑d(8),
F(5) = ∑m (0, 2, 5, 6, 7, 8, 10, 12, 3, 14, 15) + ∑ d (3),F(6) = ∑ m (2,3,7, ,11)+ ∑d(8).

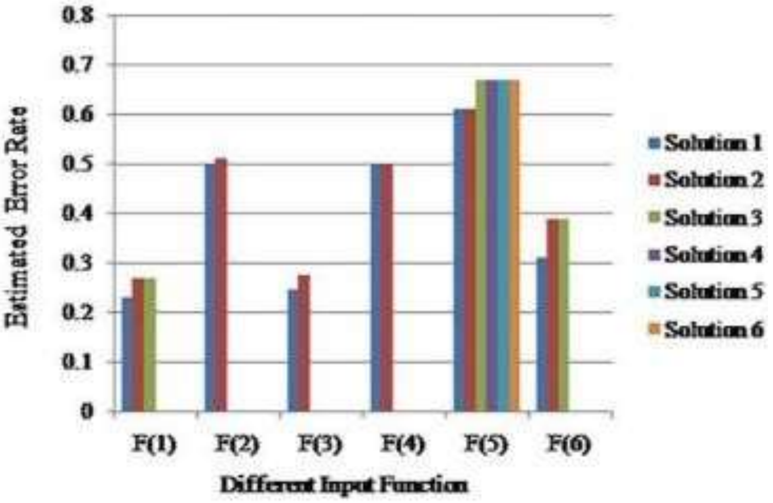


Fig. 11: *Estimated Error Rates for the Solutions of Different Boolean Functions.*

Each of F(2), F(3), F(4) have two diverse solutions. The inexact mistake rate of the returned solutions for F(2) and F(3) have distinctive values, though theinexact error rate of the rearranged solutions for F(4) have the same values. F(5) has six distinctive rearranged solutions.

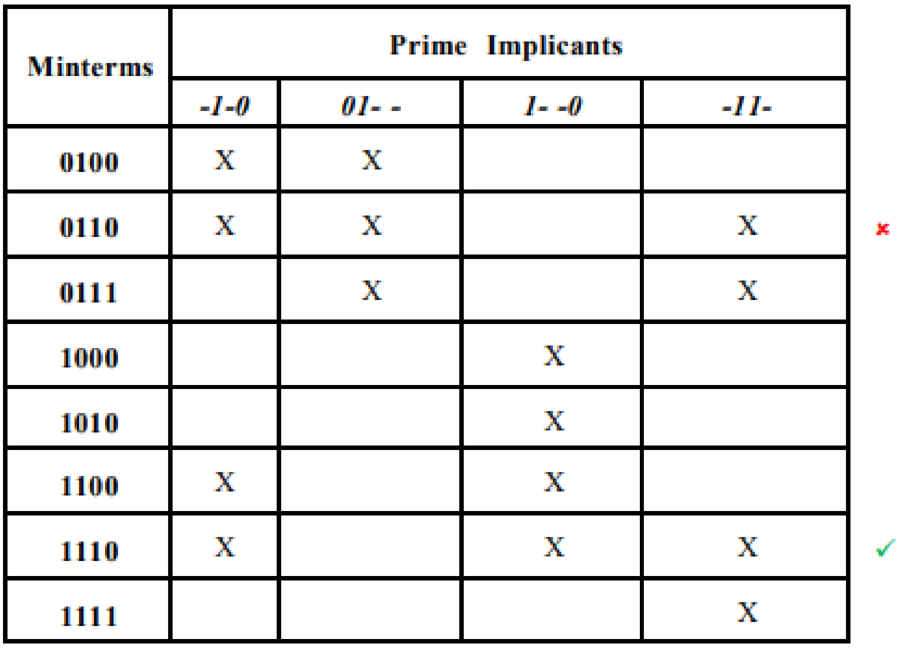
Solution 1 and Solution 2 have a rise to inexact mistake rateand these are lower than the other four break even with inexact error rates. F(6) has three distinctive rearranged solution Solution 1 has a lower error rate in comparison to both solution 2 and solution 3.

Let us examine the out comes in detail within the case of four and five variables' functions. Quick, we clarify the four variable's Boolean work (in canonical frame) $FE1$(A, B,C,D) = ∑m (4,6,7,8,10,12,14,15) +∑d (5). Our to begin with step is to discover out the prime implicants for the given work. Within the case of the given work, we have gotten four prime implicants and these are -1-0, 01--, 1--0, -11-.

We utilize all of these prime implicants to form a scope table (Table 2) for the minterms recorded in $FE1$ (A, B, C, D). Do not care terms are not considered to create the scope table. The primary column shows the minterms of $FE1$ (A, B, C, D) and the second column shows the prime implicants. The prime implicant -1-0 of function $FE1$ (A, B, C, D) make a cube with minterms 0100,0110, 1100, 1110.

Basically we will say that -1-0 may be a set of 0100,0110, 1100, 1110. Essentially, 01--, 1--0, -11- are corresponding prime implicants of (0100, 0101,0110, 0111), (1110,1100, 1010, 1000) and (1111, 1110, 0111, 0110) respectively.

Table 2: *Minterms Coverage Table- Example F_{EI} (A,B,C,D).*



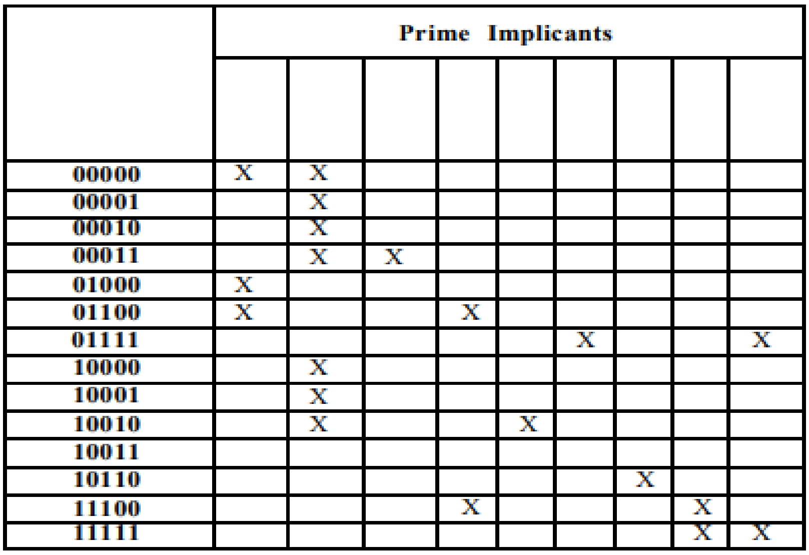
Minterms	Prime Implicants				
	-1-0	01- -	1- -0	-11-	
0100	X	X			
0110	X	X		X	✗
0111		X		X	
1000			X		
1010			X		
1100	X		X		
1110	X		X	X	✓
1111				X	

We get two rearranged solutions for FE1(A, B, C, D). These are A'B + AD' + BC and BD' + AD' + BC. The logical solution BD' + AD' + BCA'B + AD' + BC comes from 01--, 1--0, -11- and BD' + AD' + BC comes from -1-0, 1--0 +, -11-.

The minterm1110 is display three times within the solution BD' + AD' + BC. Another minterm 0110 exists three times but we have no rearranged solution that incorporates 0110 more or equal to three times. In Table 2, the push that contains minterm 0110 is stamped by a red cross. Subsequently, the simplified solution BD '+ AD' + BC features a lower probability of having errors than others. The push that contains minterm 1110 in Table 2, is stamped by the ‘ Green Check’ image.

Within the comparative way, for the Boolean work FE2(A, B, C,D) = ∑ m(0,1,2,3,8,12,15,1 ,17,18,1 ,22,2+8,31) +∑ d(4,11,29,30) we get another minterms’ scope table (Table 3). We have gotten five rearranged solutions for this problem but the solution AB'DE' + BCDE + BCD'E' + B'C' + A'D'E' has lesser probability of having mistaken than other solutions.

Table 3: *MI Interms Coverage Table- Example F_{EI} (A,B,C,D).*



	Prime Implicants								
00000	X	X							
00001		X							
00010		X							
00011		X	X						
01000	X								
01100	X			X					
01111						X			X
10000		X							
10001		X							
10010		X			X				
10011									
10110							X		
11100				X				X	
11111								X	X

We have calculated the time for distinctive Boolean functions (up to six factors) by utilizing the proposed method and the method proposed by Maleh *et al.* Table 4, shows the recorded time for both methods.

The time calculation for the proposed show is appeared as takes after. We have calculated the time required to urge yield from an input. This handle can be outlined by Fig. 12. After taking a work as input, we start the work to number time. The time tallying is finished after inferring the yields of the input work.

We have calculated the time for each work and spare it in our data file. In Table 4, the calculated time for six such capacities is shown.

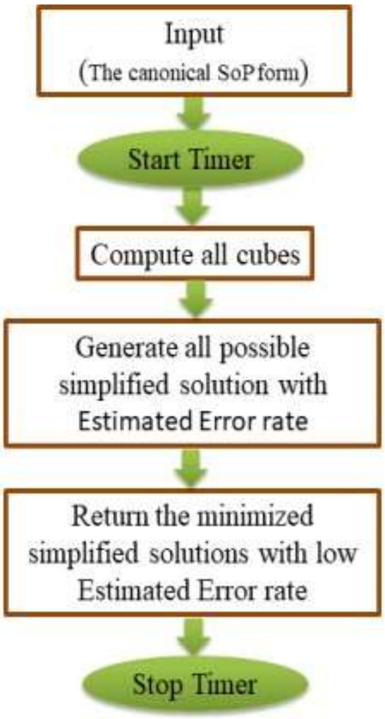
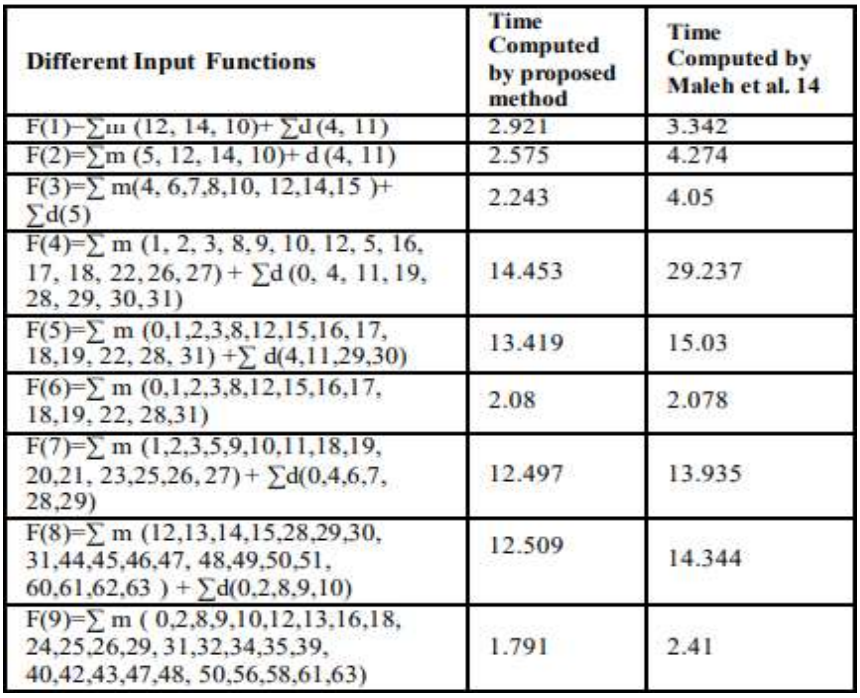

Input
(The canonical SoP form)
Start Timer
Compute all cubes
Generate all possible simplified solution with Estimated Error rate
Return the minimized simplified solutions with low Estimated Error rate
Stop Timer

Fig. 12: *The Estimated process of Computation Time.*

Table 4: *Time Computed for Dfferent Boolan Functions.*



Different Input Functions	Time Computed by proposed method	Time Computed by Maleh et al. 14
F(1)-∑m (12, 14, 10)+ ∑d (4, 11)	2.921	3.342
F(2)=∑m (5, 12, 14, 10)+ d (4, 11)	2.575	4.274
F(3)=∑ m(4, 6,7,8,10, 12,14,15)+ ∑d(5)	2.243	4.05
F(4)=∑ m (1, 2, 3, 8, 9, 10, 12, 5, 16, 17, 18, 22, 26, 27) + ∑d (0, 4, 11, 19, 28, 29, 30, 31)	14.453	29.237
F(5)=∑ m (0,1,2,3,8,12,15,16, 17, 18,19, 22, 28, 31) +∑ d(4,11,29,30)	13.419	15.03
F(6)=∑ m (0,1,2,3,8,12,15,16,17, 18,19, 22, 28,31)	2.08	2.078
F(7)=∑ m (1,2,3,5,9,10,11,18,19, 20,21, 23,25,26, 27) + ∑d(0,4,6,7, 28,29)	12.497	13.935
F(8)=∑ m (12,13,14,15,28,29,30, 31,44,45,46,47, 48,49,50,51, 60,61,62,63) + ∑d(0,2,8,9,10)	12.509	14.344
F(9)=∑ m (0,2,8,9,10,12,13,16,18, 24,25,26,29, 31,32,34,35,39, 40,42,43,47,48, 50,56,58,61,63)	1.791	2.41

Fig. 13, shows the comparison of time required between the proposed method and the method proposed by Maleh *et al.* 14. From this figure, it can be observed that the proposed method requires less time than another one in all examples.

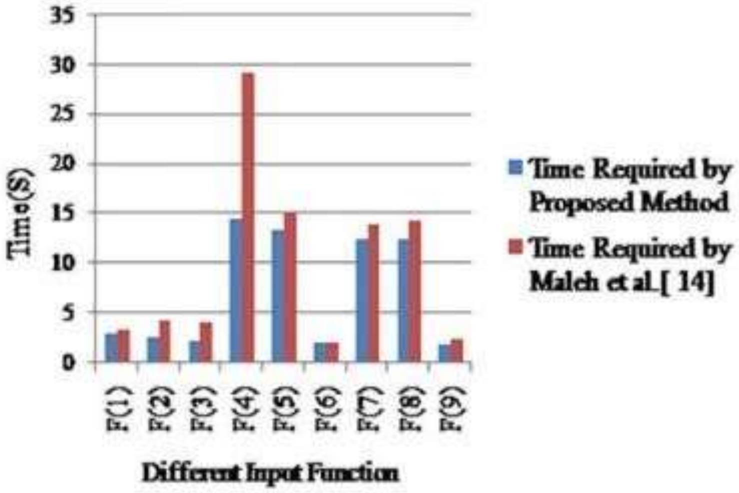


Fig. 13: *Comparison of Time Requirement between the Proposed Method and the Method Proposed by Maleh et al.*

CONCLUSIONS

In this paper, we have come up with an unused thought that upgrades the soft mistake resilience capacity of combinational circuits. The proposed method returns an productive solution among all conceivable disentangled whole of items. We have used K-map or Organization method to discover all conceivable resolutions, decided the evaluated mistake rate for all solutions, and after that hailed the solutions that have a lower failure rate. Our method is more compelling than other soft error resistance strategies portrayed within the writing. This paper returns the rearranged solution of a combinational circuit design issue and maintains a strategic distance from 16% circuit failure which reduces the soft error failure rate compared to others.

In addition, the proposed method spares a noteworthy sum of time to return simplified combination circuits with a lesser failure rate concerning existing methods.

This paper has not considered NOT gate to calculate circuit failure rate. And this paper works with the Boolean capacities having the most extreme of six factors.

Within the future, the research can be upgraded to work with the work having higher factors. In case there are factors in a work, at that point the number of Boolean capacities will be 2 2 n .Hence, all conceivable Boolean capacities can be generated at to begin with, and after that the rearranged solution with having lesser failure rates can be returned in future work.

REFERENCES

1. Xiao, J., Shi, Z. H., Jiang, J. H., Yang, X. H., Huang, Y. J., & Hu, H. G. (2019). A locating method for reliability-critical gates with a parallel-structured genetic algorithm. *Journal of Computer Science and Technology, 34*(5), 1136-1151.
2. Roy, T. R., & Sadi, M. S. (2021, May). An Efficient Approach to Tolerate Soft Errors in Combinational Circuits. In *2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)* (pp. 648-655). IEEE.
3. Tsukahara, A., & Kanasugi, A. (2018, September). Design of Approximate Arithmetic Circuits within Tolerance. In *2018 10th Computer Science and Electronic Engineering (CEEC)* (pp. 79-82). IEEE.
4. De, S., Huisken, J., & Corporaal, H. (2019, July). An automated approximation methodology for arithmetic circuits. In *2019 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)* (pp. 1-6). IEEE.
5. Jiang, H., Santiago, F. J. H., Mo, H., Liu, L., & Han, J. (2020). Approximate arithmetic circuits: A survey, characterization, and recent applications. *Proceedings of the IEEE, 108*(12), 2108-2135.