Susceptible Workload driven Selective Fault Tolerance using a Probabilistic Fault Model

Mauricio D. Gutierrez, Vasileios Tenentes, Tom J. Kazmierski

Electronics and Computer Science, University of Southampton, Southampton, United Kingdom

Email: {mdga1g11,V.Tenentes,tjk}@soton.ac.uk

Abstract—In this paper, we present a novel fault tolerance design technique, which is applicable at the register transfer level, based on protecting the functionality of logic circuits using a probabilistic fault model. The proposed technique selects the most susceptible workload of combinational circuits to protect against probabilistic faults. The workload susceptibility is ranked as the likelihood of any fault to bypass the inherent logical masking of the circuit and propagate an erroneous response to its outputs, when that workload is executed. The workload protection is achieved through a Triple Modular Redundancy (TMR) scheme by using the patterns that have been evaluated as most susceptible. We apply the proposed technique on LGSynth91 and ISCAS85 benchmarks and evaluate its fault tolerance capabilities against errors induced by permanent faults and soft errors. We show that the proposed technique, when it is applied to protect only the 32 most susceptible patterns, achieves on average of all the examined benchmarks, an error coverage improvement of 98% and 94% against errors induced by single stuck-at faults (permanent faults) and soft errors (transient faults), respectively, compared to a reduced TMR scheme that protects the same number of susceptible patterns without ranking them.

Index Terms—fault tolerance, susceptible workload, TMR, output deviations, error detection, permanent & transient faults

I. INTRODUCTION

Despite the advantages provided by technology scaling, hardware reliability has been significantly affected. Devices manufactured using 32nm technologies and below are more prone to all sources of instability and noise [1] due to the elevated cost for mitigating process variability [2]. Device characteristics are also degrading in time due to the escalation of aging mechanisms [3]. As a result, transient and permanent faults of Integrated Circuits (ICs) can appear in-the-field, and techniques to detect them are required.

Fault tolerant IC design techniques such as hardware redundancy [4], and error correcting codes, appeared as a solution for enhancing circuit reliability. Hardware redundancy consists of the complete or partial replication of a circuit in order to ensure correct functionality. By replicating the circuit, the reliability is increased as it is highly unlikely that an error would occur on every replica at the same time. Triple Modular Redundancy (TMR) utilizes two replicas of the original circuit, whose outputs are passed on to a majority voter [4]. TMR has been most widely used for safety-critical applications where robustness and data integrity are the top priority. Although TMR achieves a high level of reliability, it imposes a high area and power overhead, 200% of the original circuit plus the voter circuits, and is therefore non-viable for low power

mainstream applications. Selective hardening and Selective Fault Tolerance (SFT) have been proposed as a means of reducing area overhead and power consumption, while enhancing the reliability of circuits, when TMR is deemed too costly. Selective hardening aims to protect the most vulnerable parts of a circuit against soft errors and has been achieved in microprocessors [5], [6] through the architecture vulnerability factor of state elements. Selective hardening has also been achieved by propagating signal probabilities at the RT-Level to estimate the likelihood of an erroneous system output caused by soft errors, to select vulnerable logic blocks or nodes to protect [7], [8]. On the other hand, Selective fault tolerance (**SFT**) as introduced by [9], [10], ensures functional protection using a reduced TMR system, when is stimulated by pre-selected input patterns. Previous works on SFT rely on randomly selecting input vectors of a combinational circuit and guaranteeing protection only for these input patterns.

However, in the presence of a fault, not all input patterns are equally susceptible to it. Some patterns are less protected by the inherent logical masking of a circuit. When such patterns are executed in the presence of faults, the probability that the logical masking of circuits will be bypassed and an erroneous response will be generated is higher. Such patterns are defined as the susceptible workload.

Probabilistic fault models were developed for ranking test patterns according to their ability to sensitize the logic cones of a circuit that are more susceptible to propagate an erroneous response. Probabilistic fault models are known for enriching both the modelled and the un-modelled defect coverage of tests, without being biased towards any particular fault model. Output deviations (**OD**) [11], [12], is an RT-Level fault model calibrated through technology failure information that stems from technology reliability characterization, such as inductive fault analysis [13]. It is utilized for selecting the input patterns that maximize the probability of propagating an erroneous response to the primary outputs. The input patterns with the highest output deviations have a greater ability to penetrate the inherent logical masking of the circuit. [14] shows that selecting input patterns with high output deviations tends to provide more effective error detection capabilities than traditional fault models. In [12], a test set enrichment technique for the selection of test patterns is proposed. Output deviations have also been used for enriching the unmodelled defect coverage of tests during x-filling [15], linear [16], [17], [18] and statistical [19] compression.

In this paper we propose a novel selective fault tolerance design technique based on the probabilistic fault model of output deviations. Contrary to previous SFT techniques, which protect unranked susceptible input combinations, the proposed probabilistic selective fault tolerance design technique (PSFT), protects the most susceptible input patterns of combinational circuits, which are selected based on a ranking metric according to maximizing the circuit output deviations. This paper is organized as follows. Section II presents an overview of previous works on Selective Fault Tolerance and reviews the probabilistic fault model of output deviations. Section III describes the PSFT design technique. Simulation results of applying this technique to a set of the LGSynth91 and ISCAS85 benchmarks are shown in Section IV, where it is shown that when the most susceptible workload is protected, then the protection of any workload of the circuit is also enriched against permanent and transient faults. Simulation results of applying this technique to a set of the LGSynth91 benchmarks are shown in Section IV. Finally, the concluding remarks are presented in Section V.

II. MOTIVATION

In this section, the concept of Selective Fault Tolerance and the probabilistic fault model of output deviations (OD) are reviewed. The effectiveness of OD in detecting errors compared to the random selection of input patterns is presented as a motivational experiment.

A. Selective Fault Tolerance

Selective Fault Tolerance (**SFT**) was proposed as a modification of TMR. SFT reduces TMR cost by protecting the functionality of the circuit for only a subset of input patterns [9]. This input pattern subset X_1 is selected randomly by the system designer. The input patterns within the subset are ensured to be protected with the same level of reliability of TMR, while the rest are not guaranteed protection.

Figure 1 depicts the existing technique of SFT. For a combinational circuit $S_1=S$ to be protected using SFT, two smaller circuits s_2 and s_3 need to be generated. The behaviour of circuits s_2 and s_3 with a protected set of X_1 is described as follows:

$$S(x) = S_1(x) = s_2(x) = s_3(x), \qquad x \in X_1 \tag{1}$$

$$(S_1(x) = s_2(x)) \vee (S_1(x) = s_3(x)), \ x \notin X_1 \tag{2}$$

To determine if the input x falls within the protected set X_1, the characteristic function $\chi(x)$ must be defined [10]. The
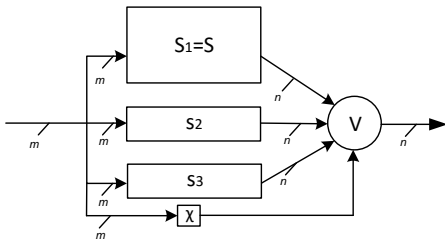
Fig. 1. Previous Selective Fault Tolerance architecture

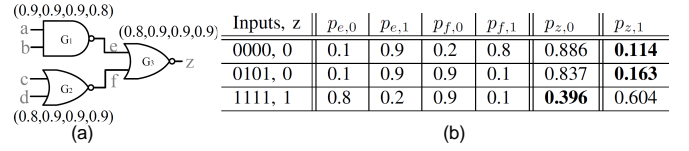| Inputs, z	$p_{e,0}$	$p_{e,1}$	$p_{f,0}$	$p_{f,1}$	$p_{z,0}$	$p_{z,1}$
0000, 0	0.1	0.9	0.2	0.8	0.886	**0.114**
0101, 0	0.1	0.9	0.9	0.1	0.837	**0.163**
1111, 1	0.8	0.2	0.9	0.1	**0.396**	0.604

Fig. 2. Output deviations example [11]: (a) simple circuit with confidence level vectors and (b) propagated output deviations

output of this function is passed on to a modified majority voter as shown in Figure 1, where the outputs of s_2 and s_3 are considered if and only if $\chi(x)=1$, otherwise the output of the TMR system is the output of circuit S_1.

$$\chi(x) = \begin{cases} 1 & x \in X_1 \\ 0 & x \notin X_1 \end{cases}$$

A simplified method for SFT was proposed in [20], where the circuits s_2 and s_3 are replaced by identical circuits. These circuits are the minimal combinational circuits that for an input pattern within the protected set X_1, exhibit the same output as the original circuit S_1. The system can be described in the following form: (*not care* indicates undefined values).

$$S(x) = S_1(x) = s_2(x) = s_3(x), \qquad x \in X_1 \tag{3}$$

$$(s_2(x) = not\ care) \wedge (s_3(x) = not\ care),\ x \notin X_1 \tag{4}$$

B. Probabilistic Fault Model: Output Deviations

The selection of input patterns with high output deviations tends to provide more effective error detection capabilities than traditional fault models [14]. The Output deviations are used to rank patterns according to their likelihood of propagating an logic error. There are a few requirements to compute the output deviations of an input pattern. First, a confidence level vector is assigned to each gate in the circuit. The confidence level R_k of a gate G_k with N inputs and one output is a vector with 2^N elements such as: $R_k = (r_k^{00...00}, r_k^{00...01}, r_k^{00...10}, \cdots, r_k^{11...11})$ where each $r_k^{xx...xx}$ denotes the probability that G_k's output is correct for the corresponding input pattern. The actual probability values can be generated from various sources, e.g., inductive fault analysis, layout information or transistor-level failure probabilities. In this paper, the probability values obtained by inductive fault analysis shown in [11] are used.

Propagation of signal probabilities in the circuit follows the principle shown in [21], with no consideration for signal correlation to reduce computation complexity. The signal probabilities $p_{k,0}$ and $p_{k,1}$ are associated to each net k in the circuit. By unitarity: $p_{k,0} + p_{k,1}=1$. In the case of a NAND gate G_k with inputs a, b and output z the propagation of signal probabilities with the confidence level vector r is as follows:

$$p_{z,0} = p_{a,0}p_{b,0}\left(1 - r_k^{00}\right) + p_{a,0}p_{b,1}\left(1 - r_k^{01}\right) + p_{a,1}p_{b,0}\left(1 - r_k^{10}\right) + p_{a,1}p_{b,1}r_k^{11} \tag{5}$$

$$p_{z,1} = p_{a,0}p_{b,0}r_k^{00} + p_{a,0}p_{b,1}r_k^{01} + p_{a,1}p_{b,0}r_k^{10} + p_{a,1}p_{b,1}\left(1 - r_k^{11}\right) \tag{6}$$

The same principle is applied to compute the signal probabilities for all gate types. For any gate G, let its fault-free output value for any input pattern t_j be d, $d \in (0, 1)$. The output deviation Δ_{Gj} of G for input pattern t_j is defined as $p_{G\bar{d}}$, where $\bar{d}$ is the complement of d ($\bar{d} = 1 - d$). In other words, the output deviation of an input pattern is a measure of

patterns	Fault Coverage (%)		Impr. (%)	Suscept. to bit flips (%)		Impr. (%)
	rp	pp		rp	pp	
8	4.9	14.2	189.4	0.4	1.9	375
16	5.3	19.6	268.6	1.6	3.3	106.3
32	6.5	26.6	311.1	2.6	4.8	84.7
64	9.7	33.7	247.9	5.1	9.4	84.3
128	21.6	42.1	94.7	10.1	16.3	61.4
256	22.4	49.8	122.1	20.8	29.6	42.3
512	37.0	56.2	51.9	45.3	57.3	26.5
1024	39.0	61.8	58.5	88.3	93.5	5.9

the likelihood that the gate output is incorrect due to a fault in the circuit, but unbiased towards any fault model [11].

Example: Figure 2 shows a circuit with a confidence level vector associated with each gate. The table contains three input patterns and their output deviations. The first column contains the input pattern (a, b, c, d), along with the expected fault-free output value z. The next columns show the signal probabilities for both logic '0' and '1' of the two internal nets and the primary output (e, f, z). The output deviation of a pattern is the likelihood that an incorrect value is observed at the output z. Therefore the output deviations (the erroneous behaviour in G_3) for the presented input patterns are: $\Delta_{3,0000} = p_{z,1}$, $\Delta_{3,0101} = p_{z,1}$, $\Delta_{3,1111} = p_{z,0}$. In this example, the input pattern 1111 has the greatest output deviation, with a probability of observing a 0 (the erroneous value) at z of $p_{z,0}$=0.396, thus offering the highest likelihood of detecting an error. ●

Table I presents the results of a motivational experiment to show that different patterns exhibit different susceptibility to permanent and transient faults. We select two sets of patterns. The first is random patterns (rp) and the second one is selected based on the probabilistic fault model of output deviations (pp) [11] for the combinational circuit *pdc* from the LGSynth91 benchmarks. For this experiment we generate rp and pp sets by gradually increasing the size of the sets from 8 to 1024 patterns. The values presented of the random patterns, under (*rp*) columns), are obtained using the average results of 30 different sets. The first column shows the size of the rp and pp sets. The next columns show the fault coverage of the rp and the pp, respectively, against permanent faults, by deploying fault simulation to compute the fault coverage of the sets against stuck-at faults. Thus, these results represent the ratio of faults which affect the operation of the circuit for the examined input patterns. Note that the pp set was obtained using random patterns and ranking [11] them without considering initial patterns that are generated for stuck-at faults, thus they are not biased towards stuck-at faults. Column 'Impr.', shows the higher susceptibility of pp against rp for permanent faults in the range [51.9% 311.1%], as expected. The next column presents the susceptibility of the sets of patterns against transient faults. This evaluation is conducted by flipping a single bit (yet examining all the possible bit flips) at every input pattern and observing, if this error is propagated at the output of the benchmark or it is masked by the inherent
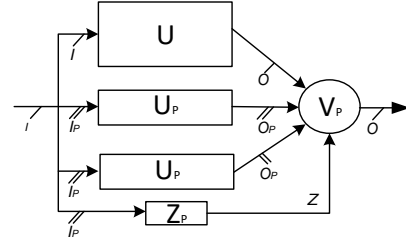

Fig. 3. Proposed Probabilistic Selective Fault Tolerance design

logic masking of the circuit. The values depicted are the percentage of bit flips at the primary inputs that successfully bypassed circuit's logical masking. Note that even for transient faults, pp set exhibit higher sensitivity [5.9% 375%], which is shown under column 'Impr.', against the rp set.

III. PROPOSED PSFT DESIGN TECHNIQUE

This section describes the proposed Probabilistic Selective Fault Tolerance (PSFT) design technique.

A. Concept

The PSFT design is presented in Figure 3. Similar to previous SFT techniques, the design consists of a reduced TMR system with the original circuit U, two smaller redundant circuits U_P, and a Z_P characteristic function that validates when the inputs of the U_P units belong to the protected input set. The original circuit U receives all the inputs, while the U_P and Z_P unit inputs are those that affect the selected logic cones determined by the desired minimum protected susceptibility γ. A majority voter V_P is used at the outputs of the circuit the output of which is selected only when the Z_P=1.

B. Proposed PSFT design flow

Figure 4 presents the flow diagram for the proposed PSFT design technique. The technique requires the number of patterns to protect (N) and minimum protected susceptibility (γ) constraints. N is defined as the number of the most susceptible patterns to protect and γ as the minimum protected susceptibility. γ ensures that the most vulnerable logic cones are protected in descending order and enables a trade-off between area overhead and fault tolerance, which is explored in Section IV. The proposed technique consists of three processes. First, the maximum output deviations of all cones are computed. The logic cone selection normalises the maximum output deviations and returns those with their normalised maximum output deviations $>= \gamma$. The pattern ranking computes the output deviations of a large number of patterns and ranks them accordingly. The patterns whose output deviations are closest to the maximum are the highest ranked. The top N ranked patterns are selected for protection and presented in PLA form for synthesis using ABC [22].

1) Process 1 (Compute maximum output deviations): The PSFT technique reads the RTL netlist and computes the maximum output deviations per logic cone. This is achieved by computing the output deviations of a large number of random patterns (100K) and marking the maximum observed output deviations per cone for every observable fault-free logic value.
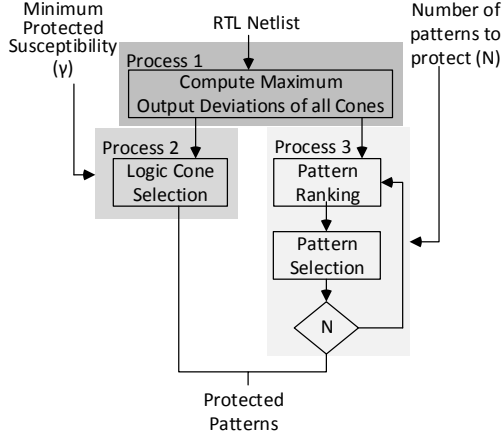
Fig. 4.   Proposed design technique flow diagram

Fig. 6.   Simulation Setup

Benchmark	I/O	Gates	Benchmark	I/O	Gates
pdc	16/40	806	t481	16/1	389
table3	14/14	445	c880	60/26	383
ex5	8/63	256	c3540	50/22	1699

2) Process 2 (Logic cone selection): Computing the output deviations allows for the identification and selection of the logic cones that provide the highest likelihood of detecting logic errors. The logic cones are selected according to their maximum output deviations. Those cones whose normalised maximum output deviations (*NMOD*) is equal or greater than the minimum protected susceptibility γ will be selected. That is *NMOD>=γ*. The *NMOD* for each cone is defined as follows:

$$NMODk = \frac{maximumOD_k}{\max_{i=1}^{n} maximumOD_i} \qquad (7)$$

Example: Figure 5 depicts the logic cone selection according to their NMOD. The 4-input 3-output circuit is divided into three overlapping sections. Each section depicts a logic cone from inputs to outputs. The red section corresponds to the logic cone of output Z_1 with *NMOD* of 1.0. blue to Z_2 with *NMOD* of 0.8 and green to Z_3 with *NMOD* of 0.2. If the desired γ is 1 then only the logic cone of output Z_1 would be selected. With γ=0.7, the leftmost two cones are to be selected. Finally, with γ=0.1 all three cones would be selected. •

The logic cone selection based on *NMOD* is dependant on the constraint γ. As γ gets closer to 1.0, the amount of selected cones is reduced discarding those with a smaller NMOD, that is, the less vulnerable cones. A trade-off analysis of area overhead vs γ is presented in Section IV.

3) Process 3 (Pattern ranking and pattern selection): The output deviations of a large number of patterns are computed and those with their output deviation at the protected cones the closest to the maximum output deviation are ranked higher.
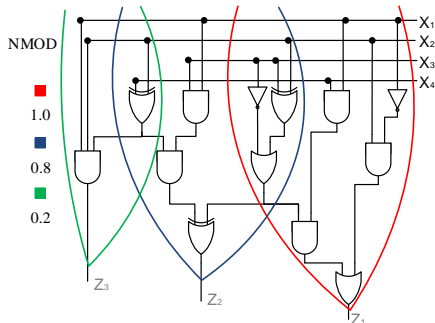

Fig. 5.   Example of logic cone selection with different NMOD
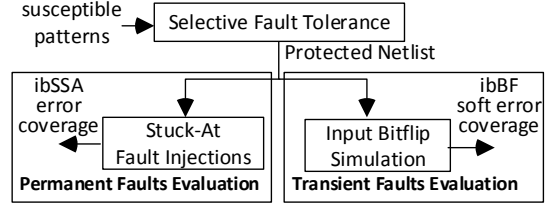
The highest ranked patterns are selected. This process is repeated until the desired number of patterns (N) have been selected, producing the protected input set.

IV.   SIMULATION RESULTS

This section presents the simulation results of applying this technique on a subset of combinational circuits of the LGSynth91 and ISCAS85 benchmark suites (Table II). The simulation setup is described and a comparison of the error coverage of random vs probabilistic patterns is shown. The area overhead of the PSFT technique and a trade-off analysis for a different number of patterns (N) and minimum protected susceptibility (γ) are presented.

Figure 6 shows the simulation setup deployed to evaluate the error coverage of the proposed technique against permanent and transient faults. The most susceptible patterns are synthesized using the ABC [22] synthesis tool into the proposed reduced TMR scheme (Figure 3), the protected netlist by the proposed technique is then evaluated using single Stuck-At fault injections to obtain the *induced by Single Stuck-At error coverage* (**ibSSA EC**) of the proposed technique, which is shown in Figure 6. Transient fault analysis is also conducted, by injecting soft errors induced by bit flips (**ibBF**) at the inputs of the circuit. The soft error coverage (SEC) is computed by injecting 50k such random upsets. The transient fault simulation consists on finding 50K combinations of random input patterns and single bit flips such that the output of the unprotected circuit (U) is affected by the bit flip at the primary input. The SEC is the percentage of these upsets that are not affecting the protected circuit.

Table III presents the comparison of the ibSSA EC in block U of the random patterns ($U_{(rp)}$) and the ranked probabilistic patterns ($U_{(pp)}$) obtained for the selected benchmark circuits. Data in $U_{(rp)}$ column is the average of 30 different random patterns selections. The $U_{(pp)}$ column shows the ibSSA error coverage of the ranked of the ranked by the proposed technique patterns. The Impr (%) column shows the improvement of the $U_{(pp)}$ over the $U_{(rp)}$ calculated as: $Impr=(U_{pp} - U_{rp})/U_{rp}$. Note that the $U_{(pp)}$ consistently exhibit a higher ibSSA EC than the $U_{(rp)}$. This improvement saturates while the number of patterns N that is protected increases. This is attributed to the increased probability that the random patterns $U_{(rp)}$

contain highly susceptible patterns. However, the saturation point depends on the size of the design, and for larger designs occurs at higher N values, which should be avoided, due to the high area cost associated with protecting large workloads using TMR.

Figure 7 presents the resulting ibSSA error coverage (EC) and area overhead of the PSFT design for the circuit *c880*. The results for a different number of protected patterns (N) are shown for minimum protected susceptibility $\gamma=1$. This indicates that the selected logic cones will be those with a normalised maximum output deviations = 1. The left axis corresponds to the ibSSA error coverage and the right axis to the area overhead of the proposed PSFT design. The area overhead of the proposed PSFT design is the sum of the area costs of the three blocks (U, U_P & Z_P) depicted in Figure 3 divided by the size of the original circuit: $(2 \cdot U_P + Z_P)/U$. For the scope of this paper the cost of the voters will be ignored. Similar to the results shown in Table III, the ibSSA EC of the pp is consistently higher than that the one of the rp for all examined N values.

The computation of the ibSSA EC of the PSFT design is calculated by adding the coverage in each of the blocks of the design. The EC in the original circuit U is obtained by the protected patterns ($U_{(pp)}$). The coverage of the U_P and Z_P blocks is 100%, as the protected patterns sensitize them fully. The ibSSA EC of the PSFT design is computed as: PSFT = $(100 \cdot (2 \cdot |U_P| + |Z_P|) + U_{(pp)} \cdot |U|)/(2 \cdot |U_P| + |Z_P| + |U|)$, where $|U|$, $|U_p|$, and $|Z_P|$ are the sizes of the blocks depicted in Figure 3.

Table IV shows the SEC improvement of pp over rp, area overhead, ibSSA EC and TMR area improvement when only the most vulnerable logic cones are selected ($\gamma=1$) for 8 and 32 protected patterns. For the SEC results, only bit flips that affect the logic cones selected using a $\gamma=1$ are used to calculate the masking capabilities of this technique. The second column shows the SEC improvement calculated by the input bit flip
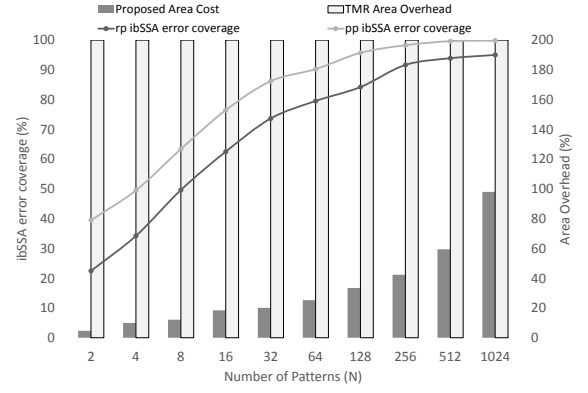

Fig. 7. Area overhead of Benchmark *c880* with minimum protected susceptibility $\gamma=1$

simulation ($[SEC_{pp}-SEC_{rp}]/SEC_{rp}$). The ibSSA EC $U_{(pp)}$ & $U_{(rp)}$ of the original circuit U and the improvement of $U_{(pp)}$ over $U_{(rp)}$ are presented in the next three columns. Followed by the total ibSSA EC of the whole PSFT design. The area overhead of each of the blocks U_P, Z_P (Figure 3) and of the full system are shown in the following three columns. The improvement in area overhead over TMR is presented in the last column, which is calculated as TMR_{impr} = 200 - $PSFT_{area}$. The table shows an average ibSSA EC of 63% with an average improvement of 98% of $U_{(pp)}$ over $U_{(rp)}$, and an average SEC improvement of 94%. When only 32 patterns are protected, we observe an area cost in the range of 18-103% for all circuits, which corresponds to a 97-181% reduction compared to TMR. Note that for circuit *c880*, using only the 32 patterns with the highest output deviations provides an ibSSA EC of 89% with an area overhead of only 20%. *c880* exhibited on average a SEC of 4.47% for rp and of 7.49% for pp, an improvement of 68%. The logic cones selected with $\gamma=1$ have an input space of 2^{10} (10 inputs), therefore, with just 32 out of 2^{10} patterns ($32/2^{10}$ = 3.13%), the proposed technique can cover 7.49% of bit flips at the inputs. Circuit *ex5* exhibit a large SEC improvement compared to the other benchmarks due the small input space (2^8), which allows for a simpler identification pp with higher coverage than rp.

Figure 8 presents the trade-off between area overhead of the

TABLE III
IMPROVEMENT OF ERROR COVERAGE (EC) INDUCED BY SINGLE
STUCK-AT FAULTS (IBSSA)

	ex5			pdc		
	ibSSA EC (%)		Impr (%)	ibSSA EC (%)		Impr (%)
patterns	$U_{(rp)}$	$U_{(pp)}$		$U_{(rp)}$	$U_{(pp)}$	
2	15.5	16.4	5.8	2.3	5.1	122.3
16	47.7	59.3	24.3	5.3	19.6	268.6
128	75.2	79.9	6.3	21.6	42.1	94.7
1024	84.0	91.0	8.4	39.0	61.8	58.5
	t481			table3		
	ibSSA EC (%)		Impr (%)	ibSSA EC (%)		Impr (%)
patterns	$U_{(rp)}$	$U_{(pp)}$		$U_{(rp)}$	$U_{(pp)}$	
2	12.4	12.5	0.8	2.6	6.1	134.4
16	23.8	28.1	18.0	11.2	28.5	155.5
128	45.2	47.0	4.0	33.0	51.5	56.3
1024	61.7	62.0	0.5	74.5	76.9	3.3
	c3540			c880		
	ibSSA EC (%)		Impr (%)	ibSSA EC (%)		Impr (%)
patterns	$U_{(rp)}$	$U_{(pp)}$		$U_{(rp)}$	$U_{(pp)}$	
2	16.1	18.8	17	22.5	39.5	75.8
16	45.1	50.1	11.6	62.5	76.5	22.3
128	71.5	79	11	84.2	95.7	13.6
1024	93.7	97.9	4	95.1	99.8	5.1

TABLE IV
PERMANENT AND SOFT ERROR COVERAGE, AREA OVERHEAD AND THE
IMPROVEMENT OVER TMR OF THE PROPOSED TECHNIQUE WITH $\gamma = 1.0$

| $\gamma = 1$, N=8	SEC	ibSSA EC (%)				Area overhead (%)			TMR
Benchmark	Impr.(%)	$U_{(pp)}$	$U_{(rp)}$	Impr.(%)	PSFT	U_P	Z_P	PSFT	Impr.(%)
pdc	71	14	4	189	20	2	3	7	193
table3	45	19	7	154	26	3	4	10	190
ex5	232	44	33	34	48	5	1	11	189
t481	50	21	18	15	32	6	4	16	184
c880	55	63	60	6	67	4	9	16	184
Average	**91**	**32**	**25**	**80**	**39**	**4**	**4**	**12**	**188**
$\gamma = 1$, N=32	SEC	ibSSA EC (%)				Area overhead (%)			TMR
Benchmark	Impr.(%)	$U_{(pp)}$	$U_{(rp)}$	Impr.(%)	PSFT	U_P	Z_P	PSFT	Impr.(%)
pdc	75	27	6	311	38	4	11	18	182
table3	25	37	15	140	48	8	9	25	175
ex5	267	71	61	17	76	10	4	24	176
t481	33	34	30	11	65	33	36	103	97
c880	68	86	78	10	89	8	4	20	180
Average	**94**	**51**	**38**	**98**	**63**	**13**	**13**	**38**	**162**
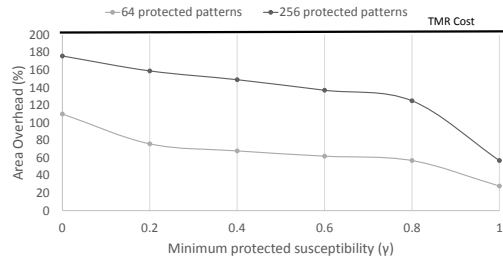
Fig. 8. Area overhead of different (γs) for benchmark *pdc*

PSFT design and different minimum protected susceptibilities (γ) values for the circuit *pdc*. The area overhead of the protected patterns N = 64 and N = 256 are shown for all γ values. With a γ=0, all the logic cones are selected, similarly when γ is 1, only the cones with NMOD = 1 are chosen. That is, only the cones that show the maximum output deviations of the circuit. When γ=0, the PSFT design is synthesized for all cones, which yields a high area overhead even for a small number of patterns. This is due to the intrinsic logic sharing present in most circuits which the synthesis tool is then unable to simply. It can be seen as expected, that for both N = 64 and N = 256, the area overhead decreases until reaching a γ=1. Note that for both 256 and 64 protected patterns, the area overhead for $\gamma = 0$ is 176% and 110% respectively, which decreases to 57% and 28% for γ=1.

V. Conclusions

We showed that not all input patterns of combinational circuits are equally susceptible to permanent or transient faults and that some patterns are less protected by the inherent logical masking of the circuit (Table I). By combining the technique of Selective Fault Tolerance (Fig. 1) and a probabilistic fault model based on the theory of output deviations (Fig. 2), we proposed a novel RTL-based selective fault tolerance design technique (Fig. 3 & 4). The proposed technique selects the most susceptible patterns and protects them using a reduced TMR scheme. We evaluated the fault tolerance capabilities of the proposed technique against errors induced by permanent faults (single stuck-at faults) and transient faults (bit flips) on a set of benchmarks (Table II). Trade-offs between achieved tolerance against permanent faults (Table III) and transient faults (Table IV) together with area overhead (Fig. 8) are also demonstrated. The fault tolerance evaluation of workload was conducted using random patterns that are not biased towards any fault model and are not explicitly protected by the proposed technique. Hence, we conclude that the protection of the most susceptible workload of combinational circuits through a probabilistic fault model that is not biased towards any type of faults, ensures that the fault tolerance against any fault model is also enriched.

References

[1] J. Srinivasan, S. V. Adve, P. Bose, and J. A. Rivers, "Lifetime reliability: toward an architectural solution," *IEEE Micro*, vol. 25, no. 3, pp. 70–80, May 2005.

[2] S. Bhunia, S. Mukhopadhyay, and K. Roy, "Process variations and process-tolerant design," in *20th International Conference on VLSI Design held jointly with 6th International Conference on Embedded Systems (VLSID'07)*, Jan 2007, pp. 699–704.

[3] M. Omaa, D. Rossi, N. Bosio, and C. Metra, "Low cost nbti degradation detection and masking approaches," *IEEE Transactions on Computers*, vol. 62, no. 3, pp. 496–509, March 2013.

[4] J. M. Cazeaux, D. Rossi, and C. Metra, "New high speed cmos self-checking voter," in *On-Line Testing Symposium, 2004. IOLTS 2004. Proceedings. 10th IEEE International*, July 2004, pp. 58–63.

[5] M. Maniatakos and Y. Makris, "Workload-driven selective hardening of control state elements in modern microprocessors," in *2010 28th VLSI Test Symposium (VTS)*, April 2010, pp. 159–164.

[6] M. Maniatakos, M. Michael, C. Tirumurti, and Y. Makris, "Revisiting vulnerability analysis in modern microprocessors," *IEEE Transactions on Computers*, vol. 64, no. 9, pp. 2664–2674, Sept 2015.

[7] S. N. Pagliarini, L. A. d. B. Naviner, and J. F. Naviner, "Selective hardening methodology for combinational logic," in *2012 13th Latin American Test Workshop (LATW)*, April 2012, pp. 1–6.

[8] C. G. Zoellin, H. J. Wunderlich, I. Polian, and B. Becker, "Selective hardening in early design steps," in *2008 13th European Test Symposium*, May 2008, pp. 185–190.

[9] M. Augustin, M. Goessel, and R. Kraemer, "Reducing the area overhead of tmr-systems by protecting specific signals," in *2010 IEEE 16th International On-Line Testing Symposium*, July 2010, pp. 268–273.

[10] ——, "Implementation of selective fault tolerance with conventional synthesis tools," in *Design and Diagnostics of Electronic Circuits Systems (DDECS), 2011 IEEE 14th International Symposium on*, April 2011, pp. 213–218.

[11] Z. Wang, K. Chakrabarty, and M. Goessel, "Test set enrichment using a probabilistic fault model and the theory of output deviations," in *Proceedings of the Design Automation Test in Europe Conference*, vol. 1, March 2006, pp. 1–6.

[12] Z. Wang and K. Chakrabarty, "An efficient test pattern selection method for improving defect coverage with reduced test data volume and test application time," in *2006 15th Asian Test Symposium*, Nov 2006.

[13] F. J. Ferguson and J. P. Shen, "A cmos fault extractor for inductive fault analysis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 7, no. 11, pp. 1181–1194, Nov 1988.

[14] Z. Wang and K. Chakrabarty, "Test-quality/cost optimization using output-deviation-based reordering of test patterns," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 2, pp. 352–365, Feb 2008.

[15] S. Balatsouka, V. Tenentes, X. Kavousianos, and K. Chakrabarty, "Defect aware x-filling for low-power scan testing," in *2010 Design, Automation Test in Europe Conference Exhibition (DATE 2010)*, March 2010.

[16] X. Kavousianos, K. Chakrabarty, E. Kalligeros, and V. Tenentes, "Defect coverage-driven window-based test compression," in *2010 19th IEEE Asian Test Symposium*, Dec 2010, pp. 141–146.

[17] X. Kavousianos, V. Tenentes, K. Chakrabarty, and E. Kalligeros, "Defect-oriented lfsr reseeding to target unmodeled defects using stuck-at test sets," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 12, pp. 2330–2335, Dec 2011.

[18] V. Tenentes and X. Kavousianos, "Low power test-compression for high test-quality and low test-data volume," in *2011 Asian Test Symposium*, Nov 2011, pp. 46–53.

[19] ——, "High-quality statistical test compression with narrow ate interface," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 32, no. 9, pp. 1369–1382, Sept 2013.

[20] L. Agnola, M. Vladutiu, M. Udrescu, and L. Prodan, "Simplified selective fault tolerance technique for protection of selected inputs via triple modular redundancy systems," in *Applied Computational Intelligence and Informatics (SACI), 2012 7th IEEE International Symposium on*, May 2012, pp. 267–272.

[21] K. P. Parker and E. J. McCluskey, "Probabilistic treatment of general combinational networks," *IEEE Transactions on Computers*, vol. C-24, no. 6, pp. 668–670, June 1975.

[22] "Abc: A system for sequential synthesis and verification." [Online]. Available: http://www.eecs.berkeley.edu/ alanmi/abc/