Extending Aging Monitors for Early Life and Wear-out Failure Prevention

Liu, Chang; Schneider, Eric; Kampmann, Matthias;
Hellebrand, Sybille; Wunderlich, Hans-Joachim

Proceedings of the 27th IEEE Asian Test Symposium (ATS'18), Hefei, Anhui, China,
15-18 October 2018, pp. 92-97

doi: https://doi.org/10.1109/ATS.2018.00028

Abstract: Aging monitors can indicate the wear-out phase of a semi-conductor device before it will actually fail, and allow the use of integrated circuits in applications with high safety and reliability demands. In the early phase of the lifecycle of integrated systems, small delay faults may indicate reliability problems and early life failures, even if they are smaller than the slack of any path and neither alter the functional behavior of a system nor violate any aging guardband. One option to detect this type of hidden delay faults (HDFs) is the application of a faster-than-at-speed-test (FAST). This paper shows that aging monitors can be extended at low cost to achieve high HDF test coverage with a reduction in test time during FAST. The result is a unified strategy to improve the reliability in both early and late phases of the system lifecycle.

Preprint

Extending Aging Monitors for Early Life and Wear-out Failure Prevention

Chang Liu*, Eric Schneider*, Matthias Kampmann†, Sybille Hellebrand† and Hans-Joachim Wunderlich*
*Institute of Computer Architecture and Computer Engineering, University of Stuttgart, Germany
†Computer Engineering, University of Paderborn, Germany

***Abstract*—Aging monitors can indicate the wear-out phase of a semi-conductor device before it will actually fail, and allow the use of integrated circuits in applications with high safety and reliability demands. In the early phase of the lifecycle of integrated systems, small delay faults may indicate reliability problems and early life failures, even if they are smaller than the slack of any path and neither alter the functional behavior of a system nor violate any aging guardband. One option to detect this type of hidden delay faults (HDFs) is the application of a faster-than-at-speed-test (FAST). This paper shows that aging monitors can be extended at low cost to achieve high HDF test coverage with a reduction in test time during FAST. The result is a unified strategy to improve the reliability in both early and late phases of the system lifecycle.**

***Keywords*-Faster-than-at-speed test, small delay faults, aging monitors**

I. Introduction

The reliability of integrated circuits is usually modeled by the so-called *bath tub curve* as shown in Fig. 1, indicating an increased failure rate at the end of the lifetime and also at the beginning with the useful phase in-between [1]. Aging mechanisms such as *Bias Temperature Instability* (BTI), *Hot-Carrier Injection* (HCI), or *electro migration* (EM) shift parameters in transistors and interconnects and cause gradual delay degradation over the device lifetime [2]. In critical applications, it is advantageous not to wait until a circuit is failing in the wear-out phase, but to warn that the end of life of the device will be reached soon. Aging monitors are a common technique for raising alerts, when the performance degradation due to aging increases the circuit delay, such that certain timing margins are violated [3–5].

In a similar way, we are interested whether a circuit will fail after manufacturing in its early life phases, even if it is fully functional during manufacturing test. Small delay faults are a good indicator of such a reliability problem, since they tend to increase soon during operation [6]. However, if their fault size is smaller than the minimum slack of all the paths through the associated locations, they are not detectable even by timing-aware test patterns, since they do not change the functionality. We call this type of small delay faults *"hidden delay fault"* (HDF), which may impose a severe reliability threat in the *early life* phase. As HDFs degrade under stress, they can be detected by a burn-in test, which stresses the hardware by increased voltage or temperature, to accelerate defect degradation and detect imperfections and "infant mortality". However, the reliability of the devices can be negatively affected by burn-in [7] and additionally, burn-in is a rather expensive process because of the demanded equipment and high test time.

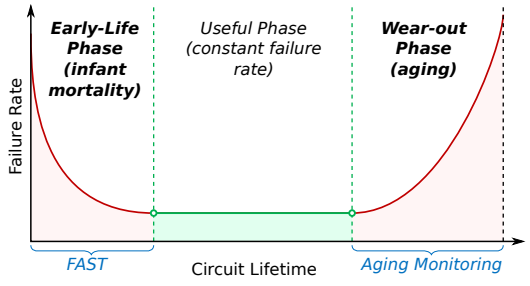


Fig. 1. Traditional *"bath tub"* curve with early-life and wear-out phases.

An alternative method for detecting hidden delay faults is to apply the test above the specified system frequency, called *faster-than-at-speed-test* (FAST) [8, 9]. However, the *maximum FAST frequency* f_{max} cannot be set merely according to the fault coverage requirement and is rather often limited by physical properties of the design (e.g., parasitic capacitance or IR-drop [10]). Therefore f_{max} is typically set to three times the nominal clock [11–13]. As we will see below, this limitation may reduce the efficacy of FAST.

Since both aging and delay faults change the delay of circuit paths, in-situ delay monitors for aging prediction can be reused to detect small delay faults [14]. This approach is compliant with *automatic test equipment* (ATE) or *built-in self test* (BIST) infrastructures for manufacturing tests or periodic self-testing. As the reuse method directly utilizes the aging alert signal of monitors to indicate the delay test results, extra hardware for test evaluation, e.g., an ATE, MISR or X-tolerant compactors [15, 16] can be evaded. However, the pre-selected fixed frequencies and maximum test speed limit the reachable fault coverage of such a method.

The work extends the structure of standard aging monitors to allow for synchronizing monitors with the inverted clock signal. This way the doubled signal sampling speed can be realized for achieving a higher HDF coverage. In addition, the fault coverage is further improved by monitoring internal circuit nodes and not just the terminals of long paths. The contributions of the paper are:

- The structure of aging monitors is extended to increase the fault coverage of hidden delay faults and reduce the number of test frequencies for FAST concurrently.
- The observability provided by monitoring internal circuit nodes for aging is also used for HDF detection.
- To minimize the test time without affecting the fault coverage, heuristics are developed to select test frequencies and to remove irrelevant patterns.

The remainder of the paper is organized as follows: Section II gives background information on small delay fault testing as well as the aging monitor structure and placement. Then Section III provides an overview of the proposed monitor extension and placement. In Section IV, a heuristic for frequency and pattern selection is presented to minimize test time. Finally in Section V, the experimental setup and results are discussed.

II. Background

A. Detection Range of Hidden Delay Faults

Delay faults are characterized by fault location and size. Throughout the paper, the *fault size* is set to 6σ, where σ is the standard deviation of the normal distribution of the process variation and set to 0.2 of the nominal gate delay. Extensions for different variations or for variable fault sizes are possible, but would complicate the presentation of this paper. To motivate the need of high test frequencies for achieving significant HDF coverage, a topological delay analysis is performed for an industrial circuit. First, the length of the longest path through each fault site is calculated using a topological analysis, and all slow-to-rise and slow-to-fall small gate delay faults at these fault sites are considered. A fault is possibly detected at a certain frequency, if a transition propagated along the longest topological path would lead to a timing violation. Due to the small fault magnitude, very few faults are possibly detected using test clock periods close to the critical path length (CPL). However, more faults become possibly detected when shorter test clock periods are chosen. Fig. 2 presents a histogram with the solid blue line representing the number of possibly detectable small delay faults for each of the shown test clock periods. As indicated by the dashed orange line, a significant number of faults are located on short paths and require high test frequencies for detection. When the shortest test period is set to one-third of CPL (i.e., $f_{max} = 3 \cdot f_{nom}$), only 21% or less faults can be detected due to larger path slacks. If the maximum test frequency rises to six times the nominal speed (i.e., $f_{max} = 6 \cdot f_{nom}$), the fault coverage is more than doubled and now reaches up to 57%. In practice, not only the topology but also the available test set determines whether a fault remains hidden or not.

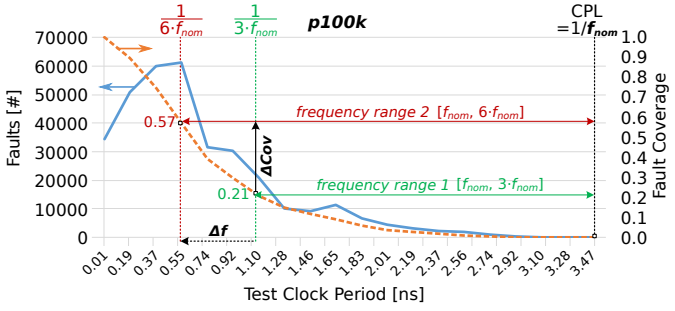


Fig. 2. Distribution of detectable hidden delay faults (left, blue) and optimistic fault coverage (right, orange) for selected test frequency ranges.

Definition 1: If a small delay fault φ is *not* detected by a test set P at a given test frequency f, then φ is considered as *hidden delay fault* with respect to P and f. For a set of frequencies F, the set of all hidden delay faults with respect to P and F is denoted as $\Phi_{HDF}(P, F)$.

For FAST, all observation times t at outputs are selected from within the interval (t_{min}, t_{nom}) whose boundaries $t_{min} = 1/f_{max}$ and $t_{nom} = 1/f_{nom}$ are defined by the maximum (f_{max}) and nominal frequency f_{nom}, respectively.

Definition 2: Let $t \in (t_{min}, t_{nom})$ be an observation time. For a small delay fault φ and test pattern set P, t is called a *detecting observation time* (DOT), if there is a pattern $p \in P$, such that φ is detected by capturing the responses at time t. The set of *all* detecting observation times is called the *detection range* $I(\varphi, P)$ *of* φ with respect to P.

In general the detection range $I(\varphi, P)$ is not a contiguous set, but it is the union of several intervals. To measure the size of a detection range, the lengths of all contained intervals are accumulated.

Definition 3: Let $I(\varphi, P)$ be the detection range of a fault φ consisting of N intervals given by $I_i = (t_i, t_{i+1})$ for $i = 1, \ldots, N$. Then the length $L(I(\varphi, P))$ is defined as $L(I(\varphi, P)) := \sum_{i=1}^{N} (t_{i+1} - t_i)$.

This work follows a pessimistic approach to reflect pulse filtering in CMOS technology. In case the length of an interval in the detection range is below a specified threshold, then the respective interval is assumed to be a *glitch* and is not added to the detection range $I(\varphi, P)$ of the fault. In the example of Fig. 3, the small glitch between the intervals I_1 and I_2 is shorter than the required threshold and hence is not included in the detection range. If a glitch masks a fault then the adjacent surrounding intervals (e.g., between I_2 and I_3) are kept pessimistically as disjoint intervals. All detection intervals outside of t_{min} and t_{max} are ignored.

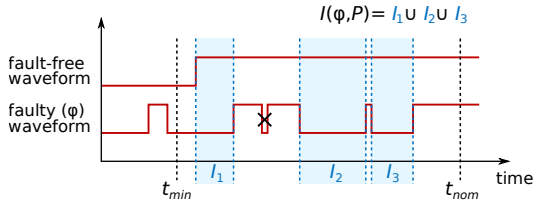
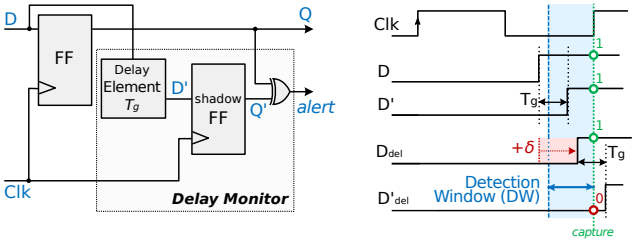


Fig. 3. *Detection range* $I(\varphi, P)$ of a fault φ for a pattern set P computed from *detection intervals* I_1, I_2 and I_3 after pulse filtering.

B. Delay Detecting Flip-Flops

Most in-situ aging monitors are based on the concept of delay detecting flip-flops which are standard flip-flops extended with a delay monitor [3]. The monitors sense the transitions during a predefined detection window and issue aging alerts for imminent timing failures. Aging monitors often contain a delay element (with delay T_g), a shadow flip-flop and an XOR gate (Fig. 4 (a)). Because of the delay element, the states of the standard (Q) and shadow flip-flop (Q') are identical only when D is stable T_g time units before the clock sampling edge. The XOR gate generates the signal *alert* by comparing the states of Q and Q'.

The *detection window* (DW) is the time period right before the clock sampling edge, i.e., the time during which the signal stability is checked. If an observed nominal signal D



(a) (b)

Fig. 4. Structure (a) and working principle (b) of a delay detecting flip-flop [3, 4] as aging monitor.

reaches its expected stable value before the detection window of monitors, then the path is functional and uncritical. On the contrary, if a signal D_{del} is delayed due to a degradation fault by some time δ, it can fall into the detection window T_g, such that it stabilizes at the shadow flip-flop input after the clock sampling edge (i.e., D'_{del} is D_{del} delayed by T_g). Since the shadow flip-flop samples an unstable signal value, different values will be compared by the XOR gate, thus generating an alert (Fig. 4 (b)).

C. Monitor Placement

Due to the high hardware cost, aging monitors cannot be integrated into all flip-flops of the design and are often placed at the end of the critical or long paths [17]. To further reduce the monitoring overhead, SlackProbe [18] proposes a monitor insertion method at intermediate circuit nets. In [19], monitors are placed at meticulously selected positions, called *observation points* inside the combinational logic. The selection of an *observation point* (OP) takes the path sensitization as well as the degradation status into account. It chooses the path prefix-segments with lengths close to half of the clock period for delay measurement. Therefore, different from monitors at path ends, aging monitors at observation points are synchronized with the inverted clock $\overline{Clk}$ (Fig. 5), and use observation times around half the clock period.

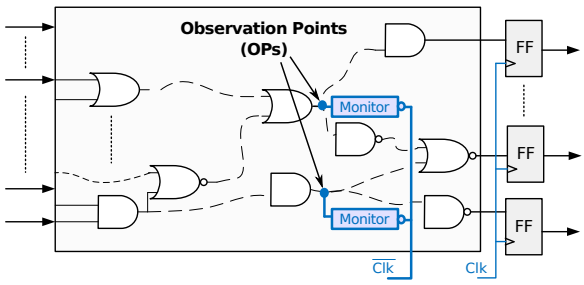
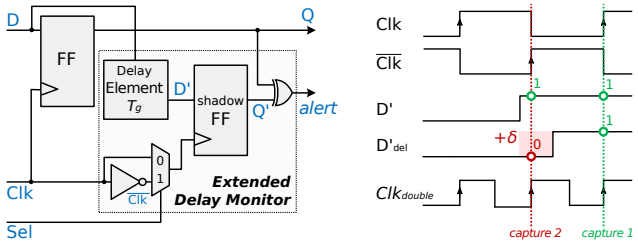


Fig. 5. Monitors placed at observation points are synchronized with the inverted clock.

III. Delay Monitor Reuse Overview

A. Extended Aging Monitor

We aim to cover the small delay faults on short paths with a low test cost, i.e., little hardware penalty, simple test configuration and low test frequencies. To achieve this goal, we extend the aging monitor structure at the path ends and reuse the internal element, i.e., shadow register for small delay fault testing. Due to the reuse, the only hardware penalty induced is the inverter and MUX of the clock selection logic before the clock pin of the shadow register (Fig. 6 (a)). The Sel signal is used to select one of two clock signals and switch the monitor between an *aging monitoring mode* and a *small delay testing mode*. In the monitoring mode ($Sel = 0$), the signal D' is sampled by the shadow flip-flop according to the regular clock Clk. In case the $alert$ signal is raised (logic '1'), an imminent failure due to degradation will be indicated. When the inverted clock $\overline{Clk}$ is selected ($Sel = 1$), the monitor works in the small delay testing mode. Assuming the sum of the delays of the inverter and MUX is in the same magnitude as the delay element (T_g), signal D is sampled after about half of the clock period by the shadow flip-flop. Hence, the test responses can be sampled and collected by the register at an observation time which corresponds to double the clock frequency (Clk_{double}), even though the rest of the circuit remains operating at nominal speed. As shown in the example of Fig. 6 (b), the faulty value of the delayed signal D_{del} (e.g., D delayed by some fault δ) cannot be observed by the standard flip-flop, since it is captured following the regular clock Clk at the capture point $capture1$. However, the hidden delay fault can indeed be captured by the shadow flip-flop with $\overline{Clk}$ that triggers at capture point $capture2$. Due to the large sampling time difference between the standard and shadow flip-flop, the expected states of these registers may not be identical anymore. Thus, the $alert$ signal cannot be directly used for fault detection and is ignored in the delay testing mode. All shadow flip-flops are organized in scan-chains. Their values as well as the values of the standard flip-flops can later be shifted out and compared with their expected test responses respectively.



(a) (b)

Fig. 6. Extended structure of the aging monitor (a) and working principle with logic values sampled by the inverted clock (b).

This extended structure is applied to all monitors at the end of circuit paths. With this novel structure, it is possible to cover more hidden delay faults which are previously undetectable due to the limitations of the maximum test frequency in conventional FAST and detect those former detectable faults at a lower test speed.

B. Observation Point Reuse for Hidden Delay Fault Testing

For an accurate and efficient aging prediction, a standard flip-flop and delay monitor (Fig. 4 (a)) are inserted at each observation point. As flip-flops and monitors placed at observation points are clocked with $\overline{Clk}$ (Fig. 5), logic values

in standard flip-flops are sampled at the half clock period, i.e., doubled frequency. As a result, the traditional monitoring structure can be directly utilized for delay testing, and no extended monitors are required. Note that the inserted aging monitors allow to cover additional shorter paths of path lengths down to $\frac{1}{2} \cdot t_{min}$ due to the inverted clock. Therefore, selected observation points in the combinational logic will provide additional observabilities for hidden delay fault detection.

Monitors placed at intermediate circuit nodes with other insertion schemes such as SlackProbe [18] can also be used for small delay fault testing, by adjusting their clock phases appropriately with delay elements.

IV. Test Frequency and Pattern Selection

Fig. 7 provides an overview of the implemented HDF test flow. First, a topological analysis of the circuit is performed (1) using timing information from *standard delay format* (SDF) files. During the process, at-speed detectable faults (with minimum slack smaller than the fault magnitude) and timing-redundant HDF locations (observable only with frequencies higher than f_{max}) are identified and removed from the initial fault list. The remaining locations are investigated by explicit timing-accurate simulation (2) using a fast and efficient GPU-accelerated small delay fault simulator [20]. The fault simulation compares the faulty waveforms to a good value simulation to determine the *detection ranges* (DR) of each fault (3) and hence provides the ranges of relevant test frequencies for detection. The detection ranges are analyzed, and the set of *target HDFs* is determined (4) based on the information of the monitor placement, i.e., positions of the internal OPs and monitors at path ends (M). We define the *target HDF* as the HDFs the fault effects (i.e. the detection ranges) of which are generated by a given test pattern set P and observable at any of the monitors, outputs or flip-flops during the valid FAST frequency range (f_{nom}, f_{max}). Finally, a selection heuristic identifies a set of relevant test frequencies $F \subseteq (f_{nom}, f_{max})$ (5) in combination with selected pattern pairs from P (6) in order to generate an efficient *test schedule* for detecting the targeted HDFs with reduced test time. A test schedule $S \subseteq F \times P$ is a set of frequency-pattern combinations $(f, p) \in F \times P$, each of which indicates the required test frequency f for the application of a test pattern p.

A. Aligning Detection Ranges

We intend to maximize the small delay fault coverage with the extra observability provided by aging monitors and concurrently minimize the test time. Aging monitors enlarge the detection range of small delay faults and enable the measurement of the signal behavior at a higher speed region, which is previously unreachable due to the constraints of the maximum test frequency. Since the signal sampling time of aging monitors is half of the provided test period, we need to map the detection range observed by monitors to the time region of the required clock.

Based on a particular pattern set P, we perform the good-machine time simulation and fault simulation [20] for all target

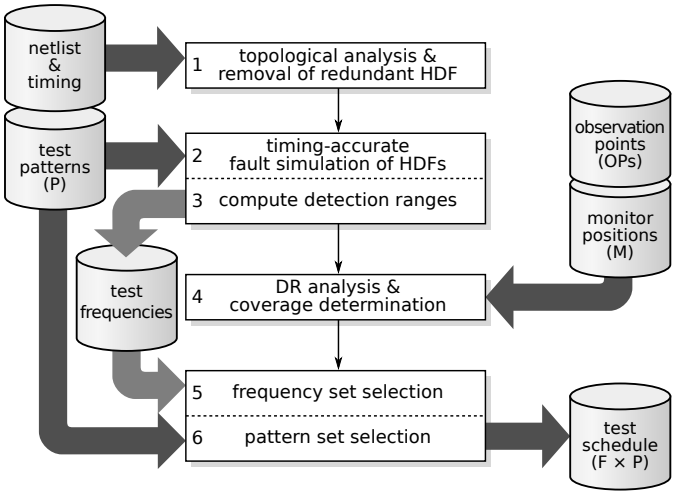


Fig. 7. Overview of the proposed HDF test flow.

faults φ, then collect the waveforms at all primary, pseudo-primary and observation points. The fault-free and corresponding faulty waveforms are compared for the identification of detection ranges $I(\varphi, P)$ which is the union of the detection ranges observed at standard flip-flops, shadow registers in monitors at path ends and observation points (Fig. 7 step 2 and 3).

$$I(\varphi, P) = I_{\text{FF}}(\varphi, P) \cup I_M(\varphi, P) \cup I_{\text{OP}}(\varphi, P) \quad (1)$$

As mentioned above, we assume that the standard flip-flops in the circuit can sample the waveforms with frequencies in the range from f_{nom} to $f_{max} = 3 \cdot f_{nom}$ during FAST resulting in the sampling interval $I_{\text{FF}}(\varphi, P) \subseteq (t_{min}, t_{nom}) = (\frac{1}{3} \cdot t_{nom}, t_{nom})$ (cf. Sec. II). The detection range with respect to standard flip-flops is the union of time intervals during which SDFs can be observed. The inverted clock utilized by the shadow registers in aging monitors at path ends, as well as the flip-flops at observation points leads to halved observation times, hence resulting in a *shifted* detection range $I_M(\varphi, P) \cup I_{\text{OP}}(\varphi, P) \subseteq (\frac{1}{2} \cdot t_{min}, \frac{1}{2} \cdot t_{nom}) = (\frac{1}{6} \cdot t_{nom}, \frac{1}{2} \cdot t_{nom})$. In order to simplify the test frequency selection problem in the next section, the detection ranges with respect to the monitors are therefore transformed linearly: Detection ranges of monitors at path ends and observation points $I_M(\varphi, P) \cup I_{\text{OP}}(\varphi, P) \subseteq (\frac{1}{2} \cdot t_{min}, \frac{1}{2} \cdot t_{nom})$ are mapped by

$$map : (\tfrac{1}{2} \cdot t_{min}, \tfrac{1}{2} \cdot t_{nom}) \mapsto (t_{min}, t_{nom}), t \mapsto 2 \cdot t, \quad (2)$$

to a corresponding range $I_{2f}(\varphi, P)$ in the regular sample interval domain.

The *combined* detection range $I'(\varphi, P)$ to be utilized for the test frequency and pattern selection is then computed as

$$I'(\varphi, P) := I_{\text{FF}}(\varphi, P) \cup I_{2f}(\varphi, P). \quad (3)$$

B. Selection Heuristic

Clock generators are commonly based on the *Phase Locked Loop* (PLL) structure. If the switching of frequencies requires re-locking the PLL during FAST, usually it takes tens or hundreds of microseconds, which corresponds to a loss of

thousands or tens of thousands of instruction cycles [21, 22]. As a result, the number of frequencies needed for FAST often has a larger impact on test time than the number of patterns. Therefore in our test time optimization process, we first aim to pick a minimum number of test frequencies and then optimize the pattern set for each selected test speed.

To cover all target faults Φ with a minimum number of test frequencies is a Minimum Hitting Set problem. This NP-hard problem can be solved by a hypergraph-based method [23]. For a low computational effort, a heuristic selection algorithm is implemented in this work.

The following steps are performed to select appropriate observation times for obtaining relevant FAST frequencies:

1) Select the hard-to-detect fault with the minimal length of the detection range: Select φ_h, if $L(\varphi_h) = \underset{\varphi \in \Phi}{min}\{L(\varphi)\}$.
2) Intersect $I(\varphi_h, P)$ of the selected fault φ_h with the detection ranges of other faults. This results in a reduced detection range $I_{red}(\varphi_h, P) := (\cup_{\varphi \in (\Phi \setminus \varphi_h)}\{I(\varphi, P)\}) \cap I(\varphi_h, P)$.
3) Select an observation time $t_s \in I_{red}(\varphi_h, P)$ which covers a maximum number of faults Φ_{t_s}.
4) Remove the detected faults from the target list: $\Phi = \Phi \setminus \Phi_{t_s}$. If $\Phi \neq \varnothing$, repeat from step 1).

After the relevant observation times have been determined, the steps of the pattern selection are performed for each of the selected test times $t_s \in C_s$:

1) Select the pattern pair detecting the maximum number faults: Select p_s, if $|\Phi_{p_s}| = \underset{p \in P}{max}\{|\Phi_{p_i}|\}$.
2) Remove the detected faults from the FAST group: $\Phi_{t_s} = \Phi_{t_s} \setminus \Phi_{p_s}$. If $\Phi_{t_s} \neq \varnothing$, repeat from step 1).

V. Evaluation

In the experiments, benchmark designs from ISCAS'89 and industrial designs were synthesized using the NanGate 45nm open cell library [24] in order to obtain the netlist and the timing-information. During synthesis, monitors are assumed to be integrated at long path ends [17] and internal *observation points* (OPs). Monitors at long path ends cover 25% of the total pseudo-primary outputs. The OPs are selected with the same setup as in [19]. Compacted transition delay fault test sets with an average test coverage of over 99.9% are used for the evaluation, which were generated by a commercial ATPG tool. As initial fault set, fault locations are considered at each input and output pin of a gate in the circuits assuming two individual small delay faults for modeling *slow-to-rise* and *slow-to-fall* effects. The nominal clock period of each circuit is set to the critical path length from static timing analysis. All experiments were performed on a host system equipped with two Intel Xeon E5-2690 v3 processors clocked at 2.6GHz and 512GB of RAM.

A. Increased Hidden Delay Fault Coverage

Table I summarizes the basic circuit statistics and the targeted hidden delay faults (cf. Sec. IV). Column 2 and 3 report the size of the circuit in number of gates along with the number of flip-flops. The size of the ATPG-generated transition fault test pattern set is given in column 4. In column 5 and 6, the number of internal OPs as well as the number of monitors at path ends ($|M|$) is shown. Column 7 through 9 then summarize the number of targeted hidden delay faults for traditional FAST *without* (*old*) and the novel proposed approach *with* the use of extended monitors (*prop.*) as well as the relative gain in HDF detection shown in the last column.

As shown, the monitor extension allows for an increase in the number of targeted hidden delay faults ranging from 3.2% (circuit p78k) up to 90.5% (p89k) with an average of over 36.8% for all of the designs investigated. This way, more HDFs can be detected allowing for a broader surveillance of both early life and wear-out failures within the circuit.

B. Frequency and Pattern Selection Results

Table II summarizes the required test frequencies and test pattern pairs in the test schedule resulting from the selection heuristic to achieve a given coverage of targeted hidden delay faults (cf. Table I). For a given coverage FC of target HDFs the required number of test frequencies $|F_{FC}|$, the overall number of pattern test $|P_{FC}|$ in the generated test schedule as well as the resulting reduction ($\Delta\%$) compared to a naïve test approach with $|F_{FC}| \times |P|$ pattern tests is reported. The respective results of each coverage target are given in consecutive columns.

As shown for $FC = 0.90$, only few test frequencies are required to achieve the coverage of targeted HDFs with the provided pattern set. Trivially, the higher the coverage target is, the more test frequencies and more pattern tests are required. Especially from $FC = 0.99$ to $FC = 1.00$, the number of required test frequencies is doubled for most of the cases. However, compared to the naïve test approach with all frequencies and all patterns, the number of pattern tests (and hence the reduction in test time) reduces significantly for all circuits by an average of 81.0% up to 96.2% (p45k). For certain cases the heuristic could not achieve any reduction (e.g., circuit p78k with $FC = 0.90$). Here, all patterns need to be applied for each of the computed frequencies as the removal of any pattern test will result in a reduction in coverage.

VI. Conclusion

Even though hidden delay faults (HDFs) do not violate the functional timing at the time of manufacturing, they are

TABLE I. Circuit statistics and targeted hidden delay faults (HDF).

Circuit[1]	Gates[2]	FFs[3]	$\|P\|$[4]	OPs[5]	$\|M\|$[6]	Target HDFs $\Phi_{HDF}(P,F)$		
						old[7]	*prop.*[8]	$\Delta\%$[9]
s9234	1766	228	155	13	63	5946	6349	(+6.7%)
s13207	2867	669	195	16	198	3938	6645	(+68.7%)
s15850	3324	597	134	7	169	4198	7285	(+73.5%)
s35932	11168	1728	39	184	513	30854	33134	(+7.3%)
s38417	9796	1636	128	28	435	26849	30797	(+14.7%)
s38584	12213	1450	160	11	426	22823	29410	(+28.8%)
p35k	23294	2173	1512	2	558	42071	56735	(+34.8%)
p45k	25406	2331	2719	4	638	61876	79510	(+28.4%)
p78k	70495	2977	70	306	872	305863	315886	(+3.2%)
p89k	58726	4301	993	16	1140	63032	120114	(+90.5%)
p100k	60767	5735	2631	36	1458	118675	184544	(+55.5%)
p141k	107655	10501	842	8	2626	213798	276433	(+29.2%)

TABLE II. Number of test frequencies $|F_{FC}|$ and pattern tests $|P_{FC}|$ of the generated test schedules S_{FC} for different coverages FC of targeted hidden delay faults (delay size $\delta = 6\sigma$). The reduction in test time $\Delta\%$ by the test schedules is given as relative reduction in pattern tests $(1 - |S_{FC}|/|F_{FC} \times P|)$.

Circuit[1]	Targeted Hidden Delay Fault Coverage (FC)														
	$FC \geq 0.90$			$FC \geq 0.95$			$FC \geq 0.98$			$FC \geq 0.99$			$FC = 1.00$		
	$\|F_{90}\|^{(2)}$	$\|P_{90}\|^{(3)}$	$\Delta\%^{(4)}$	$\|F_{95}\|^{(5)}$	$\|P_{95}\|^{(6)}$	$\Delta\%^{(7)}$	$\|F_{98}\|^{(8)}$	$\|P_{98}\|^{(9)}$	$\Delta\%^{(10)}$	$\|F_{99}\|^{(11)}$	$\|P_{99}\|^{(12)}$	$\Delta\%^{(13)}$	$\|F_{100}\|^{(14)}$	$\|P_{100}\|^{(15)}$	$\Delta\%^{(16)}$
s9234	8	408	-67.1%	10	459	-70.4%	14	508	-76.6%	16	525	-78.8%	24	550	-85.2%
s13207	5	382	-60.8%	7	465	-65.9%	11	522	-75.7%	13	537	-78.8%	23	577	-87.1%
s15850	6	329	-59.1%	9	427	-64.6%	13	478	-72.6%	15	496	-75.3%	29	528	-86.4%
s35932	4	149	-4.5%	5	183	-6.2%	7	238	-12.8%	8	259	-17.0%	16	333	-46.6%
s38417	8	741	-27.6%	11	917	-34.9%	17	1111	-48.9%	21	1191	-55.7%	39	1313	-73.7%
s38584	6	633	-34.1%	8	762	-40.5%	13	967	-53.5%	16	1055	-58.8%	34	1177	-78.4%
p35k	19	4380	-84.8%	28	4956	-88.3%	40	5422	-91.0%	49	5592	-92.5%	90	5827	-95.7%
p45k	5	3088	-77.3%	8	3421	-84.3%	11	3707	-87.6%	15	3933	-90.4%	41	4208	-96.2%
p78k	3	210	0.0%	6	420	0.0%	9	621	-1.4%	13	870	-4.4%	52	1808	-50.3%
p89k	12	4741	-60.2%	16	5508	-65.3%	24	6322	-73.5%	30	6642	-77.7%	69	7119	-89.6%
p100k	5	4988	-62.1%	9	6294	-73.4%	16	7690	-81.7%	21	8285	-85.0%	67	9073	-94.9%
p141k	9	5334	-28.1%	18	7806	-47.4%	30	9469	-61.7%	40	10230	-69.0%	93	11248	-85.3%

an essential indicator of hardware marginalities. However, detection of HDFs is often costly, and the maximum FAST frequency constrains the fault coverage. This work extends the in-situ delay monitors for aging monitoring to a test structure for efficient small delay fault detection at lower test frequencies. By inverting the clock signal of shadow registers in aging monitors the data signals can be sampled at observation times corresponding to doubling the applied test frequency. As a result, hidden delay faults can be detected by using the novel structure, which were previously undetectable. Additionally, already formerly detectable faults may now be detected at even lower test frequencies. For covering all target faults with a minimized test time, we present a heuristic selection algorithm. Experimental results showed a substantial increase in the coverage of HDFs by up to 90.5% through reuse of the shadow registers. The presented heuristic covers all targeted HDFs with a reduction in test time of up to 96.2% over conventional FAST.

ACKNOWLEDGEMENTS

Parts of this work have been supported by the German Research Foundation (DFG) under grants WU 245/19-1 and HE1686/4-1, FAST.

REFERENCES

[1] G. A. Klutke, P. C. Kiessler, and M. A. Wortman, "A Critical Look at the Bathtub Curve," *IEEE Trans. on Reliability (TR)*, vol. 52, no. 1, pp. 125–129, Mar. 2003.

[2] X. Li, J. Qin, and J. B. Bernstein, "Compact Modeling of MOSFET Wearout Mechanisms for Circuit-Reliability Simulation," *IEEE Trans. on Device and Materials Reliability (TDMR)*, vol. 8, no. 1, pp. 98–121, Mar. 2008.

[3] M. Agarwal, B. C. Paul *et al.*, "Circuit Failure Prediction and Its Application to Transistor Aging," in *Proc. IEEE 25th VLSI Test Symp. (VTS)*, May 2007, pp. 277–286.

[4] M. Saliva, F. Cacho *et al.*, "Digital Circuits Reliability with In-Situ Monitors in 28nm Fully Depleted SOI," in *Proc. Conf. on Design, Automation Test in Europe (DATE)*, Mar. 2015, pp. 441–446.

[5] M. Omaña, D. Rossi *et al.*, "Low Cost NBTI Degradation Detection and Masking Approaches," *IEEE Transactions on Computers*, vol. 62, no. 3, pp. 496–509, March 2013.

[6] Y. M. Kim, T. W. Chen *et al.*, "Gate-Oxide Early-Life Failure Identification using Delay Shifts," in *Proc. 28th VLSI Test Symp. (VTS)*, April 2010, pp. 69–74.

[7] R. Vollertsen and R. Hijab, "Burn-In," in *IEEE Int'l Integrated Reliability Workshop Final Report*, Oct. 1999, pp. 132–133.

[8] W. W. Mao and M. D. Ciletti, "A Variable Observation Time Method for Testing Delay Faults," in *Proc. ACM/IEEE 27th Design Automation Conf. (DAC)*, Jun. 1990, pp. 728–731.

[9] H. Yan and A. D. Singh, "Experiments in Detecting Delay Faults using Multiple Higher Frequency Clocks and Results from Neighboring Die," in *Proc. IEEE Int'l Test Conf. (ITC)*, Sep. 2003, pp. 105–111.

[10] R. Tayade and J. A. Abraham, "On-chip Programmable Capture for Accurate Path Delay Test and Characterization," in *Proc. IEEE Int'l Test Conf. (ITC)*, Oct. 2008, pp. 1–10, Paper 6.2.

[11] M. Amodeo and B. Cory, "Defining faster-than-at-speed delay tests," *Cadence Nanometer Test Quarterly eNewsletter*, vol. 2, no. 2, May 2005.

[12] N. Ahmed, M. Tehranipoor, and V. Jayaram, "A Novel Framework for Faster-than-at-Speed Delay Test Considering IR-drop Effects," in *Proc. IEEE/ACM Int'l Conf. on Computer-Aided Design (ICCAD)*, Nov. 2006, pp. 198–203.

[13] J. Lee and E. J. McCluskey, "Failing Frequency Signature Analysis," in *Proc. IEEE Int'l Test Conf. (ITC)*, Oct. 2008, pp. 1–8, Paper 5.2.

[14] C. Liu, M. A. Kochte, and H.-J. Wunderlich, "Aging Monitor Reuse for Small Delay Fault Testing," in *Proc. 35th VLSI Test Symp. (VTS)*, Apr. 2017, pp. 1–6.

[15] A. Singh, C. Han, and X. Qian, "An Output Compression Scheme for Handling X-states from Over-Clocked Delay Tests," in *Proc. 28th VLSI Test Symp. (VTS)*, Apr. 2010, pp. 57–62.

[16] S. Hellebrand, T. Indlekofer *et al.*, "FAST-BIST: Faster-than-At-Speed BIST Targeting Hidden Delay Defects," in *Proc. IEEE Int'l Test Conf. (ITC)*, Oct. 2014, pp. 1–8, Paper 29.3.

[17] M. Agarwal, V. Balakrishnan *et al.*, "Optimized Circuit Failure Prediction for Aging: Practicality and Promise," in *Proc. IEEE Int'l Test Conf. (ITC)*, Oct. 2008, pp. 1–10, Paper 26.1.

[18] L. Lai, V. Chandra *et al.*, "SlackProbe: A Flexible and Efficient In Situ Timing Slack Monitoring Methodology," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, vol. 33, no. 8, pp. 1168–1179, Aug. 2014.

[19] C. Liu, M. A. Kochte, and H.-J. Wunderlich, "Efficient Observation Point Selection for Aging Monitoring," in *Proc. IEEE 21st Int'l On-Line Testing Symp. (IOLTS)*, Jul. 2015, pp. 176–181.

[20] E. Schneider, M. A. Kochte *et al.*, "GPU-Accelerated Simulation of Small Delay Faults," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, vol. 36, no. 5, pp. 829–841, May 2017.

[21] K. J. Nowka, G. D. Carpenter *et al.*, "A 32-bit PowerPC System-on-a-Chip With Support for Dynamic Voltage Scaling and Dynamic Frequency Scaling," *IEEE Journ. of Solid-State Circuits (JSSC)*, vol. 37, no. 11, pp. 1441–1447, Nov. 2002.

[22] Cypress Semiconductor Corp., *CY274X, 4-PLL Spread-Spectrum Clock Generator*, Apr. 2017. [Online]. Available: http://www.cypress.com/documentation/datasheets/cy27410-4-pll-spread-spectrum-clock-generator

[23] M. Kampmann, M. A. Kochte *et al.*, "Optimized Selection of Frequencies for Faster-Than-at-Speed Test," in *Proc. IEEE 24th Asian Test Symp. (ATS)*, Nov. 2015, pp. 109–114.

[24] Nangate Inc. (2018) NanGate 45nm Open Cell Library. [Online]. Available: http://www.nangate.com/