EMPIRICAL TIMING ANALYSIS OF CPUS AND DELAY FAULT TOLERANT DESIGN USING PARTIAL REDUNDANCY

A Dissertation

by

SANGHOAN CHANG

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2007

Major Subject: Computer Engineering

EMPIRICAL TIMING ANALYSIS OF CPUS AND DELAY FAULT TOLERANT DESIGN USING PARTIAL REDUNDANCY

A Dissertation

by

SANGHOAN CHANG

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Gwan S. Choi
Committee Members,	A. L. N. Reddy
	J. Silva-Martinez
	P. Li
	H. Walker
Head of Department,	C. Georghiades

May 2007

Major Subject: Computer Engineering

ABSTRACT

Empirical Timing Analysis of CPUs and Delay Fault Tolerant
Design Using Partial Redundancy. (May 2007)
Sanghoan Chang, B.S., Seoul National University;
M.S., Seoul National University
Chair of Advisory Committee: Dr. Gwan S. Choi

The operating clock frequency is determined by the longest signal propagation delay, setup/hold time, and timing margin. These are becoming less predictable with the increasing design complexity and process miniaturization. The difficult challenge is then to ensure that a device operating at its clock frequency is error-free with quantifiable assurance. Effort at device-level engineering will not suffice for these circuits exhibiting wide process variation and heightened sensitivities to operating condition stress. Logic-level redress of this issue is a necessity and we propose a design-level remedy for this timing-uncertainty problem.

The aim of the design and analysis approaches presented in this dissertation is to provide framework, SABRE, wherein an increased operating clock frequency can be achieved. The approach is a combination of analytical modeling, experimental analysis, hardware /time-redundancy design, exception handling and recovery techniques. Our proposed design replicates only a necessary part of the original circuit to avoid high hardware overhead as in triple-modular-redundancy (TMR). The timing-critical combinational circuit is path-wise partitioned into two sections. The combinational circuits associated with long paths are laid out without any intrusion except for the fan-out connections from the first section of the circuit to a replicated second section of the combinational circuit. Thus only the second section of the circuit is replicated. The signals fanning out from the first section are latches, and thus are far shorter than

the paths spanning the entire combinational circuit. The replicated circuit is timed at a subsequent clock cycle to ascertain relaxed timing paths. This insures that the likelihood of mistiming due to stress or process variation is eliminated. During the subsequent clock cycle, the outcome of the two logically identical, yet time-interleaved, circuit outputs are compared to detect faults. When a fault is detected, the retry signal is triggered and the dynamic frequency-step-down takes place before a pipe flush, and retry is issued. The significant timing overhead associated with the retry is offset by the rarity of the timing violation events. Simulation results on ISCAS Benchmark circuits show that 10% of clock frequency gain is possible with 10 to 20 % of hardware overhead of replicated timing-critical circuit.

To My family, Haejin, Seungwon, Hiewon, and Parents

ACKNOWLEDGMENTS

First, I'd like to thank my advisor, Dr. Gwan S. Choi, for his kind advice and helpful guidance throughout my research. Without him, this dissertation would not be possible. I also thank my committee members, Drs. P. Li, H. Walker, A. Reddy, and J. Silva-Martinez.

I'd like to thank Dr. Li for the discussions on empirical delay failure rate estimation. The discussions helped me a lot to prepare for my final exam and dissertation. I'd also like to thank Dr. Walker for his suggestions and invaluable comments in the prelim and final exam. He provided me with various ways of looking at my research topics. I also thank Dr. Reddy for reminding me of the importance of discussion and presentation skills. I know I have a lot to improve. Dr. Silva-Martinez's comments were also useful because they were from a different viewpoint on my research topics.

I also thank Euncheol, Kiran, Pankaj, and Rohit for their lively discussions and invaluable input. I wish them good luck in the future.

Last, but not least, I thank my family for being with me through good times and hard times. Their understanding and encouragement have done half of my work.

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

FIGURE Page

FIGURE Page

CHAPTER I

INTRODUCTION

A. Motivation of research

Market's demands for better performance integrated circuits have driven the continuing improvements and innovations of VLSI technology. As characterized by Moore's law [1][2], the integration density of IC has increased exponentially and the operating clock frequency has reached at multi-GHz area owing to miniaturization of feature size and introduction of new materials and techniques in fabrication process[3][4].

Operating clock frequency is one of the key parameters representing IC's performance and determined by the signal delay through the longest path which is called the critical path. Hence, inaccurate timing evaluation in design stage may result in production of devices with too much timing margin (overdesign) or ones failing to satisfy the specification required by market. Both the cases are losses in manufacturing economics because the former means too much resources are spent and the latter means low yield. Similar cases can happen in speed binning and post-fabrication test where the timing characteristics of fabricated devices are evaluated and the operating clock frequency is marked. If it is too conservative, the gap between the operation performance and realizable performance will be big (underperformance) and in the other case, it may not guarantee the time-to-failure specification.

The accurate timing estimation is hindered by uncertainties in every stage of device manufacturing processes. First, in design stage, the design complexity resulting from exponentially increasing integration density prohibits the complete design

---

The journal model is *IEEE Transactions on Automatic Control.*

evaluation coverage. Even the number of the critical paths is expected to increase exponentially as the feature size is reduced [5]. Sometimes the experiences from the current technology can not be applicable to develop the next generation technology [6]. Second, in fabrication process, the electrical parameter fluctuations resulting from the intra-die process variations is not ignorable anymore in addition to those from the traditional, die-to-die process variations [7]. Finally, the timing of ICs are becoming more and more vulnerable to operating noises like temperature variations and power supply voltage fluctuations [8][9]. During operation, a signal propagating along a path may have interferences from the surrounding circuits, which are crosstalks. As the vertical dimension of devices has not been scaled as much as the horizontal dimension of them in IC fabrication process, the effects of crosstalk on signal delay has increased with miniaturization of feature size [10].

The timing estimation of a fabricated IC based on experiments is important in the senses that 1) it can provide timing characteristics information of device to be feed backed to design stage for realistic timing margin requirements, 2) it can play a role as a post-production test coverage evaluator, and 3) it can provide data to direct possible timing fault tolerant design. However, direct measurement of timing related quantities, for example, mean time to failure, is not possible in normal operating conditions because of the timing margin added in speed binning. To invoke measurable delay failures, the experiments need to be carried out in stress conditions. In the next sections, previous works on delay test, process variations, and stress test will be briefly described.

B. Previous work

1. Delay model and test

Common approach for addressing the timing abnormalities is delay modeling and test. A delay test is conducted by propagating a transition signal from input, through the target path, to output. Hence this requires two-test vector sets: one to stabilize the path and the other to initiate the signal transition in the target path.

Originally the delay fault models were developed for studying the faults residing in single gate's inputs and/or outputs (transition fault model [11]) or anywhere through a signal propagation path (path delay fault model [12]). In transition fault model, two kinds of fault, slow-to-rise and slow-to-fall, are aimed at and, because they can be treated as stuck-at-faults when you restrict clock period, the test vectors for stuck-at-faults can be used. Although it may be easy to cover the faults by applying the techniques used for stuck-at-faults model, it may not be possible to detect delay faults scattered across the whole path. Hence its usefulness is limited to the defects where their delay time is long enough to cause a logical failure for any signal passing the gate. This limitation of transition fault model can be overcome by path delay fault model. Path delay model treats the aggregated delay through a whole path and it has been extensively studied. In [12], a method based on 6-valued logic is used to determine whether a path delay fault is detected by a given input vector pair. First, vector V_1 is applied to the inputs and it propagates to the outputs. Then, vector V_2 is applied and, according to the 6-valued logic, the signal propagates to the output. By tracing back to the inputs, we know that which path is covered or not. The difficulty in applying this model to tests of integrated circuits is in its complexity.

As the number of paths grows exponentially as the number of gates increases, the main focus of path delay model is to cover paths of the longest path delays

and to find robust test which detects delay faults regardless of all other delays in the circuits. Another approach for quantitative delay fault model (gate delay fault model) is proposed by Carter et al.[13]. In the model, it is assumed that the delay time, size and location of the faults are known with some precision, which is not always possible. A fault is an added delay of certain size(time) in the propagation of a rising or falling transition from the gate input to output and for the given fault, and test vector is applied to detect the delay fault of assumed size. So, a little differences in the size of delay fault may result in different test vectors. Most of researches on the model have focused on the determination of the minimum fault size detected by given test [14][15].

Recently, two more delay fault models are added, line delay fault model[16] and segment delay fault model[17]. In line delay fault model, two tests for each line, rising delay test and falling delay test, test the delay through the longest sensitizable path passing the target line with rising or falling signal transition. One clear benefit of this model is that the maximum number of faults in the circuit is the twice of the number of lines. But the robust test is not always possible for the longest path passing the target line and it is necessary to test all non-robustly testable paths with longer delay than that of the longest robustly testable path. Another limitation of this model is that a test that targets a line with the longest path passing it may fail to detect faults distributed across a shorter path which includes the line. To deal with this situation, we need to test more paths and it can result in large number of paths to be tested. A delay fault model which compasses from transition fault to path delay fault is proposed by Heragu et al [17]. The segment length L is defined as 1 for transition fault and the maximum logic depth for path delay fault and can be chosen from available statistics on the types of manufacturing defects.

In proposing a delay fault tolerant design for operating clock frequency gain,

which is the main goal of this thesis, one of the issues to be considered is the mean-time-to-failure(MTTF) or failure rate at increased operating clock frequency. If it is too short, the performance gain in clock frequency can be reduced by the time budget for fault recovery and if it is too long, possible performance gain is wasted. The unfortunate limitation of the delay test is that test coverage is not directly related to the MTTF or delay failure rate. To solve this problem, delay failure rate of CPUs are estimated by combining model, analysis, and experiments. The experiments are conducted in realistic operating conditions using practical applications. They will be explained in chapter II.

2. Inter-die and intra-die process variation

Variations in process parameters can cause fluctuations in within-die (WID), within-wafer (WIW), wafer-to-wafer (W2W), and lot-to-lot (L2L) device's electrical characteristics. In [18], the possibilities for critical dimension (CD)'s variation is studied for lithography and etch processes. The sources of the CD variations are categorized according to the scale of variations from L2L to across the field. Here the field means the area of the silicon wafer which is exposed at the same time. Another example is where the inter-metal dielectric thickness after wafer polishing shows W2W and L2L variations [19]. In modeling process variations' effects, L2L, W2W, and WIW fluctuations comprise the die-to-die (D2D) fluctuation.

In the past, D2D fluctuations dominated the variations in ICs performance and WID fluctuations were neglected [7]. But as the minimum feature size of technology has decreased below the wavelength of light source used in stepper of lithography process, the WID fluctuations in defined pattern are comparable to the D2D fluctuations. In [20], the intra-die, uncorrelated parameter variations' effect on path delay is studied and the delay variations in carry select adder circuits from 0.5 μm technology

and 0.18 μm technology are compared. It shows that as the minimum feature size decreases, the influence from intra-die process parameter fluctuation increases. The intra-die process variations effects on path delay is shown to be more severe to low voltage operation circuits and to circuits with a large number of critical paths and low logic depth. For WID fluctuations, there are random ones and systematic ones. The doping concentration in device channel is one example of random fluctuations where device-to-device correlation is zero even they are adjacent to each other [21]. CD variation on a die is systematic one because it varies around a principal value resulting from the different focus of stepper lens in die-to-die process [18].

K. Bowman et al [5] modeled the maximum operating clock frequency (FMAX) distribution of CPU starting from number of critical paths in 0.25 μm technology. Using test vehicle and statistical simulation, the inter-die and intra-die variations are extracted and used to calculate FMAX distribution of CPU. It shows that the critical path delay distribution has 9 % and 3% of standard deviation/average ratio from die-to-die and intra-die process variation, respectively. Other factors that can affect FMAX distribution are operating condition's variation or operating noises like variations in temperature and power supply voltage, and crosstalk. In chapter III, the estimated operating noises' effect will be considered to modify FMAX distribution.

3. Stress test

The failure rate or the time-to-failure(TTF) is a measurable quantity in experiments for timing characteristics estimation. Because of the timing margin added in conservative speed binning, the TTF at normal operating conditions is expected to be tremendously long. Hence stress techniques in operating clock frequency and temperature should be applied to experiments.

Stress tests are widely used in industry to remove manufacturing defect to prevent

early failures in field (infant failure). An example is where voltage stress has been used to detect gate oxide defects like pinhole or excessively thin oxide. In [22], the time-dependent-dielectric-breakdown(TDDB) characteristics are studied to detect the oxide defects for the different shapes of gate oxides and pinhole is explained to be created from etch damages. High temperature also used in stress test to accelerate device degradation. For example, data retention is one of key reliability characteristics of non-volatile memories like Flash EEPROM. To estimate the time to data loss, which is usually guaranteed to be 10 years, the hot temperature stress is used to accelerate the charge escaping from the insulated floating gate [23]. To detect any defect in passivation layer, high humidity in test environment is also used because the hydrogen from the water can easily penetrate the passivation layer and cause electrical failures in operations of chips. In burn-in test, where every devices are under stress to remove the product of manufacturing defect, highly accelerated temperature/humidity test (HAST) is usually employed [24]. Another application of stress test for device life time extraction is explained in chapter II, section B. In chapter II, the experiments are carried out in higher temperature than that of normal operating condition and the results will be used to extrapolate the result of normal operating condition.

C. Research direction

The final goal of the research is to propose a delay fault tolerant design for possible performance gain. Introduction of fault tolerant design to integrated circuits accompany a penalty in complexity. Hence, we need to study the trade-off between performance gain and circuits' complexity and the operating clock frequency of circuits is the essential parameter of the performance.

In determining target operation clock frequency of proposed design, the first step

will be to estimate the delay failure rate at various clock frequencies. Chapter II of this thesis is dedicated to the topic of delay failure rate estimation. First, starting from individual path delay distribution, a delay failure rate distribution model of circuits in pipelined structure is developed. For delay failure rates data to extract distribution parameters, experiments are conducted on CPUs at various operating clock frequencies. By combining the model and experimental data, delay failure rate distributions of CPUs are estimated. Because the estimated delay failure rate distribution is mainly affected by operation related noises, in chapter III, the operation noises' effects on the maximum operating clock frequency (FMAX) distribution are studied.

The proposed delay fault tolerant design, SABRE, is described in chapter IV. The difference between traditional fault tolerant system and delay fault tolerant system for operating clock frequency gain is that just adding redundancy is not enough. For example, triple-modular-redundancy (TMR) for delay fault tolerance may not work if the systems operate at higher clock frequency than that of specification. To be used as a reference signal to detect and remove delay fault, it is most important to ensure enough timing slack in the reference signal's propagation path. In the research, we try to achieve the delay fault tolerance by duplicating partial signal paths from the original circuits and placing them in the next pipeline stage. By sharing part of the original circuits, the area penalty for the redundancy circuits is optimized. The path delay across the duplicated circuits part will be added to the timing slack for the reference signals and the signal outputs from the original circuits will be compared to the reference signals. If the delay fault is detected, it will be removed by pipeline flushing and retry in lower operating clock frequency. In chapter V, the proposed delay fault detection scheme is applied to implement a design for single-event-transient (SET) fault detection.

CHAPTER II

EMPIRICAL ESTIMATION OF DELAY FAILURE RATE OF CPUS *

A. Introduction

This chapter presents a methodology for estimating the delay failure rate of a device during nominal operation using combination of analyses, modeling, and experiments. Section B describes the research approach and Section C presents modeling of the delay failure rate distribution. The factors affecting delay, crosstalk, power voltage fluctuation, and temperature, are analyzed in Section D. The Gaussian delay failure rate model is validated using Monte-Carlo simulation in Section E. In Section F, the experimental setup is described and the results are shown in Section G. Conclusion and future work are in Section H.

B. Approach

The Figure 1 shows the overall proposed approach for estimating delay failure rates. Initially the sources of noises like crosstalk, power fluctuation, and temperature variation are analyzed. The delay model that considers the effects of these noises is constructed. First, the relationship between path delay distribution and delay failure rate is introduced. Next, the noises' effects on path delay distribution are addressed. To apply the model in estimating the delay failure rate through time-to-failure(TTF) measurements data, a Gaussian path delay distribution is assumed. Next, this assumption is validated using Monte-Carlo simulation. The parameters

---

*Based on "Timing Failure Analysis of Commercial CPUs Under Operating Stress," by Sanghoan Chang and Gwan Choi which appeared in 21st IEEE International Symposium on Defect and Fault Tolerance in VLSI Systems, Oct. 2006. ©[2006] IEEE.

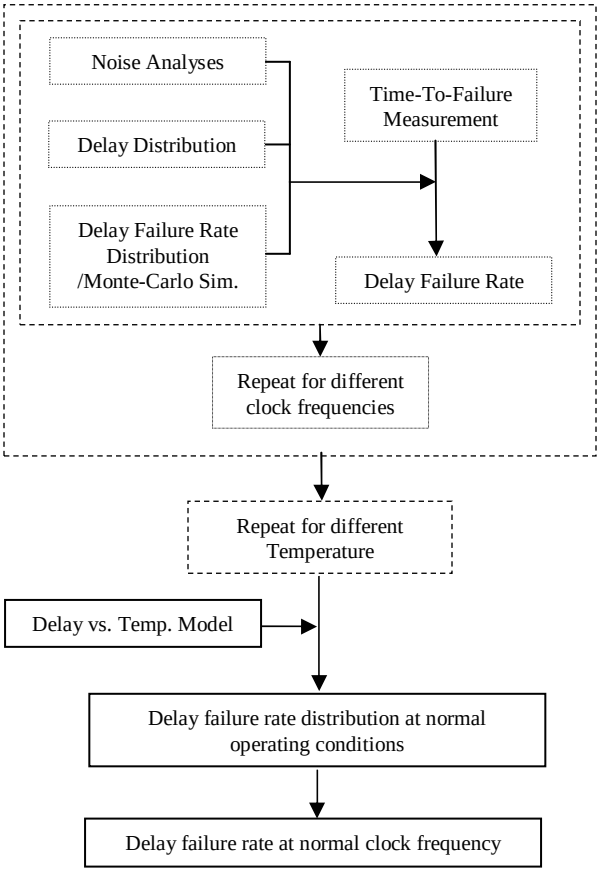


Fig. 1. Delay failure rate estimation flow

of the distribution are extracted from the experimental data and used to estimate the delay failure rate at normal operating clock frequency. In the experiments, the temperature and clock frequencies are set to higher values from those of nominal operating conditions to trigger the delay faults that have very small chance of activation in normal operating conditions. The results obtained from the experiments are time-to-failure distributions for several different stress conditions. These results are then used to project the delay failure rates at nominal operating conditions.

Stress test is a very common practice in estimating the lifetime of device [25][26][27][28][29]. For example, transistor degradations due to hot carrier effect are measured

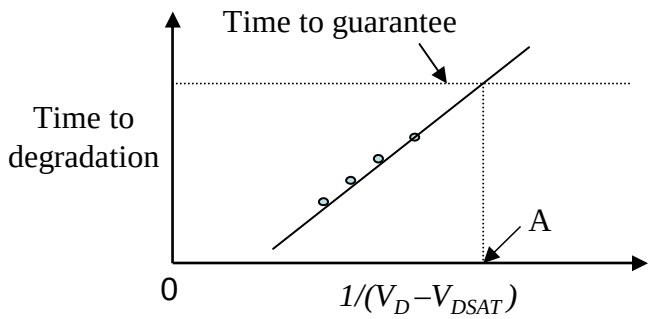


Fig. 2. Device lifetime extraction

at various (drain bias (V_D), gate bias (V_G)) combinations where V_G is chosen so that gate current (I_G) or substrate current (I_{SUB}) is maximized for the corresponding V_D. Then, measured stress times to predetermined current or transconductance degradation or threshold voltage shift are extrapolated to obtain that of normal operating V_D. Figure 2 shows an example of the plot. To guarantee the lifetime of the transistor, $1/(V_D - V_{DSAT})$ for normal operating condition should be placed at the right side of point A in the figure.

Estimating the lifetime of integrated circuit under device wear-out is more complicated than that of a single transistor [30][31]. First, model parameter sets need to be extracted from the transistors at various levels of degradations. Second, the level of degradation should be correlated to the age of device in normal operating conditions. Third, the fact that each transistor in the same gate is under different operating stress should be considered in the circuit simulation. The result is still deterministic and the lifetime estimation based on it may be very conservative according to the allowed variation of model parameters and operating conditions in the simulation. The timing margin of device in operating conditions variations can be evaluated by Shmoo plot [32]. Still it is questionable whether it can cover all the possible combination of signals inside the chip, e.g. crosstalk and delay paths as a

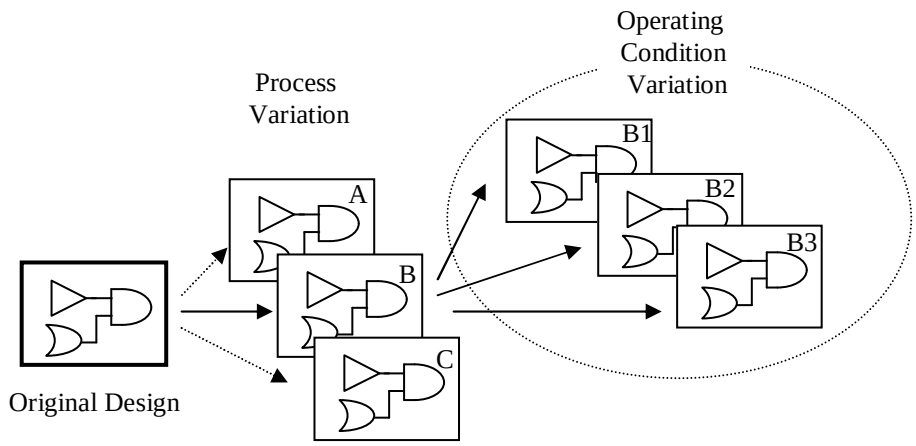


Fig. 3. Device performance variations

result of long sequence of input vectors.

The proposed approach tries to solve these problems by estimating the delay failure rate distribution based on path delay distribution model and experiments in realistic operating environments.

C. Delay failure rate model

In this Section, delay failure rate model is developed. The focus is on the delay variations resulting from operating noises as illustrated in Figure 3. Even devices are fabricated from a single design, they are differ in signal propagation delay because of the process variations (A, B, and C in the figure). For a specific path, the delay varies according to the operating conditions (B1, B2, and B3 in the figure).

In a circuit with m number of paths and l number of primary outputs (PO), the path delay distribution at k^{th} clock t_k of time period t_p can be represented by l-dimensional vector D_k=$\{d_1, d_2, ..., d_l\}$, each of l path delays having distribution, d_i=$g_i(t)$, attributed to the effect of variations in operating conditions and noises like

crosstalk. Then, the probability of delay fault is given by $P_d(k) = P\{max\{d_1, ..., d_l\} > t_p\}$ and it is defined over the path delay probability space as shown in equation (2.1). p is a probability that a path to PO has a delay fault and in the range of interests in this research, it is less than 10^{-10}. $F(d_1, ..., d_i, ..., d_l)$ is the joint probability density function over $\{d_1...d_l\}$ space.

$$\begin{aligned} P_d(k) &= 1 - \int_{d_1=-\infty}^{d_1=t_p} ... \int_{d_l=-\infty}^{d_l=t_p} F(d_1, ..., d_i, ..., d_l)\mathrm{d}d_1...\mathrm{d}d_l \\ &= \sum_{i=1}^{l} \int_{d_i=t_p}^{d_i=\infty} \mathrm{d}d_i \int_{d_1=-\infty}^{d_1=t_p} ... \int_{d_l=-\infty}^{d_l=t_p} F(d_1...d_l)\mathrm{d}d_1...\mathrm{d}d_l + O(p^2) \end{aligned} \tag{2.1}$$

The direct integration of $F(d_1, ..., di, ..., dl)$ is extremely difficult as l grows larger [6]. Even some paths are dependant on others, considering the fact that the chance of any path has a delay fault is very small, $P_d(k)$ can be approximated as shown in equation (2.2) when $\int_{t=0}^{t=t_p} g_i(t)\mathrm{d}t \approx 1$ for any i.

$$P_d(k) \approx \sum_{i=1}^{l} \int_{t=t_p}^{t=\infty} g_i(t)\mathrm{d}t \tag{2.2}$$

The equation (2.2) means that when $t \geq t_p$, for any path, the probability of delay fault is almost 0. If we take the average of $P_d(k)$ over time $T = M \cdot t_p$ ($M \gg 1$), it is given by equation (2.3).

$$P_d = \frac{1}{M} \sum_{k=1}^{M} P_d(k) = \sum_{i=1}^{m} \int_{t=t_p}^{\infty} f_i(t) \cdot p_{ex,i} dt \tag{2.3}$$

$f_i(t)$ stands for the delay distribution of each path in the circuit. $p_{ex,i}$ is the excitation probability of the path i. If we apply (2.3) to a circuit of pipeline depth n, where i^{th} pipeline combinational circuit (CC_i) has m_i paths, the P_d can be rewritten as the

following equation (2.4).

$$P_{d,i} = \sum_{j=1}^{m_i} \int_{t=t_p}^{\infty} f_{i,j}(t) \cdot p_{ex(i,j)} dt \tag{2.4}$$

In researches on path delay, the distribution of the longest path caused by process variation is the focus of analysis because it determines the performance and yield [33]. However, in this research, all path delays are considered since the delay failure rate resulting from operating conditions variations is statistical in nature and it is assumed that any process defects rendering abnormally long path delay are rejected during post-fabrication tests.

The observation of a delay failure at the output involves three probability values. First is the excitation probability ($p_{ex(i,j)}$) that the path becomes active. Second is the probability that the path has a longer delay than clock period, t_p. This we refer to as ($p_{d(i,j)}(t_p)$). The last is the propagation probability of delay fault to output without becoming masked (p_{pp}). $p_{ex(i,j)}$ is specific to each path and p_{pp} is a constant for all paths in a combinational circuit of a pipeline stage. $P_{d,i}(t)$ denotes the probability of all the excited delay faults at time t during the clock period t_p in CC_i. When there is a delay fault in CC_i, it will propagate to PO or be masked out over the next pipeline stages. The probabilities for all possible events originating from CC_i can be expressed by $H_i(t)$ in the following equation:

$$\begin{aligned} H_i(t) = P_{n,i}(t) \quad &+ \quad P_{d,i}(t) \cdot [p_{m,i+1}(t+1) \\ &+ \quad p_{pp,i+1}(t+1) \cdot \{p_{m,i+2}(t+2) + p_{pp,i+2}(t+2)...\}] \end{aligned} \tag{2.5}$$

where $P_{d,i}(t)$ is the probability of excited fault at time t, $P_{n,i}(t)$ is $1 - P_{d,i}(t)$, $p_{m,j}(t)$ is the masking probability of CC_j at time t, and $p_{pp,j}(t)$ is the fault propagation probability of CC_j at time t. We state this general event expression a place holder

for the failure event probabilities that we empirically derived from the subsequent analysis. In statistical analysis, all the probabilities are averaged over time and it is expected that the values are constant as long as the operating conditions remain fixed. Hence, the time parameter 't's can now be omitted without losing generality. Then the equation (2.5) can be rewritten as:

$$H_i(t) = P_{n,i} + P_{d,i} \cdot [p_{m,i+1} + p_{pp,i+1} \cdot (p_{m,i+2} + p_{pp,i+2} ...)] \tag{2.6}$$

If there is a failure at the primary output at time t, the probabilities for all these events can be obtained by:

$$H(t) = H_1(t-n) \cdot H_2(t-n+1) ... H_N(t-1) \tag{2.7}$$

$H(t)$ is now expanded to obtain the expression for delay failure probability. The following assumption is made to simplify the expression.

Assumption: Only a single-point-fault/single-failure is considered. The value of $P_{d,i}$ is extremely small and the contributions from the multiple fault driven delay failures(near-coincidental faults) are expected to be insignificant.

Note that the above assumption is applicable while $P_{d,i} \gg P_{d,j} \cdot P_{d,k}$ for any i, j, and k. With the assumption, the expression for the delay failure probability (P_{df}) is simplified to

$$\begin{aligned} P_{df} &\approx \sum_{k=1}^{n} [P_{d,k} \cdot \prod_{i=1, i \neq k}^{n} P_{n,i} \cdot \prod_{j=k+1}^{n} p_{pp,j}] \\ &\approx \sum_{k=1}^{n} [P_{d,k} \cdot \prod_{j=k+1}^{n} p_{pp,j}] = \sum_{k=1}^{n} [P_{d,k} \cdot P_{pp}(k)] \end{aligned} \tag{2.8}$$

where $P_{pp}(k) = \prod_{j=k+1}^{n} p_{pp,j}$.

Finally the delay failure probability is formulated in (2.9).

$$P_{df}(t_p) = \int_{t'=t_p}^{t'=\infty} \sum_{i=1}^{n} [\sum_{j=1}^{m_j} \{f_{i,j}(t') \cdot p_{ex}(i,j)\} \cdot P_{pp}(i)]dt' \equiv \int_{t'=t_p}^{t'=\infty} S(t')dt' \tag{2.9}$$

where $f_{i,j}$ is path delay distribution of jth path in i^{th} pipelined structure, p_{ex} is the excitation probability, and $P_{pp}(i)$ is the propagation probability to output through CC_{i+1} to CC_n. To calculate $P_{df}(t)$, it is necessary to characterize $S(t)$. As is described in the next section, the effects of operating noises on path delay make the individual path delay distribution, $f_{i,j}$, a Gaussian distribution. Then, $S(t)$ is a summation of Gaussian distributions. In Section E, using the results of Monte-Carlo simulations, it is shown that the summation of Gaussian distribution can be approximated to a normal distribution for the ranges of interests in this research.

In operating circuit, operation for each clock can be considered to a Bernoulli trial of failure probability of P_{df}. Then, the probability, $P(n)$, that it survives $(n-1)$ clocks and fails at n^{th} clock is given (2.10).

$$P(n) = P_{df} \times (1 - P_{df})^{n-1} \tag{2.10}$$

And the cumulative failure rate, $P_c(n)$, is given by (2.11) if $n \gg N$ (pipeline depth).

$$P_c(n) = \sum_{i=1}^{n} P(i) = 1 - (1 - P_{df})^n \tag{2.11}$$

In (2.11), P_{df} is given as a distribution of operating clock period. At a fixed clock frequency, the cumulative failure rate, $P_c(n)$, corresponds to the cumulative TTF distribution in experiments. Hence, by fitting the cumulative TFF distributions measured in the experiments to (2.11), the value of P_{df} can now be calculated. With P_{df}'s at multiple points, (2.9) can be used to estimate the delay failure rate distribution.

D. Modeling of operation noises

There are several reported analyses of delay distribution of a chip. In [34], the delay distribution of a chip is assumed to be Gaussian by the Central Limit Theorem. Even the sum of correlated random variables, the assumption of Gaussian distribution is applicable for most practical models of correlation [35]. To address the characteristics of the path delay distribution, it is necessary to consider the factors affecting it. From the design of circuits to its usage in the field, two factors can affect the continuous path delay distribution. The first results from the variations of manufacturing process. The variations in the processes parameters like the gate oxide thickness, gate width, wire-trace width and thickness contribute to the path delay variation. These form the continuous path delay distribution. As the result of these variations, the devices fabricated from a single design can vary in the signal transition delay and, hence, the performance.

The other attribute to the timing variation is of operating condition in origin and it is specific to the parameters that vary during field use. Such variations include sources like power-supply noise, temperature, capacitive couplings/crosstalk among other environmental factors. The following subsections build modeling components associated with several known noise sources.

1. Crosstalk vs. delay

When there are multiple signal propagations across a chip, each signal is under influence of other signals voltage level changes. It may boost the signal propagation or defer it, which is called crosstalk. As the feature size of technology scales, the shrink rate of the vertical dimension can not follow that of the horizontal dimension. The result is that "edge capacitance" from adjacent wires plays a dominant role in

wire's capacitive coupling [36]. Past research shows that the capacitive crosstalk can cause up to 15 % variations in delay [37]. The crosstalk affecting the timing of victim wire-trace from the surrounding circuits is studied for ASIC circuits [34]. Using known crosstalk analysis method, the signal transition window technique, the aggressor wire's effect on the victim's delay is expressed in (2.12). In [38], the N aggressors are classified into two groups according to their signal transition time comparing with that of victim's, where both transition times are similar (N_1) and where the aggressor's transition time is shorter than that of victim's (N_2) as below .

$$\Delta t_{pd} = \sum_{i=1}^{N} \Delta t_{pdi} \propto \sum_{i=1}^{N} C_{pri} \cdot \Delta_i = \sum_{i=1}^{N_1} C_{pri} \cdot \Delta_{1i} + \sum_{i=1}^{N_2} C_{pri} \cdot \Delta_{2i} \tag{2.12}$$

where C_{pri} is the coupling ratio between the victim and ith aggressor, Δ_i is its contribution to delay which is determined by its transition timing with respect to that of the victim's. For a victim path consisting of M victim segments, the delay is given by the summation of aggressors' crosstalk over each segment as in (2.13).

$$\begin{aligned} \Delta t &= \sum_{j=1}^{M} \Delta t_{pd,j} \qquad (2.13) \\ &= \sum_{j=1}^{M} \left[\sum_{i=1}^{N_j} \Delta t_{pdi} \right] \propto \sum_{j=1}^{M} \left[\sum_{i=1}^{N_j} C_{pri} \cdot \Delta i \right] = \sum_{j=1}^{M} \left[\sum_{i=1}^{N_{1j}} C_{pri} \cdot \Delta_{1i} + \sum_{i=1}^{N_{2j}} C_{pri} \cdot \Delta_{2i} \right] \end{aligned}$$

In [38], for a long signal wire, the probability for capacitive coupling ratio from the aggressors and helpers is shown to have a normal distribution. As the number of coupling contributors to signal path delay in (2.13) increases, by the central limit theorem [39], the delay distribution approximates to a Gaussian distribution.

2. Power noise vs. delay

The delay variation resulting from switching noise has been found to be mainly dependent on the first ground bounce peak [40] and the ground bounce peak is shown

to be proportional to the number of simultaneously switching gate [41]. Hence, the delay distribution follows the shape of the distribution of the number of gates which switch at the same time. This delay distribution too can be approximated to a Gaussian distribution. Power supply noise is also shown to impact the delay for device dominated paths [42]:

$$D = D_0 \cdot [1 - f(N_w) \cdot N_a - g(N_a) \cdot N_w - \varphi(N_a, N_w)] \tag{2.14}$$

where f and g are the path-dependent noise sensitivity factors and $\varphi(N_a, N_w)$ is for the higher order dependencies on N_a and N_w which are the noise pulse amplitude and width, respectively. Considering the fact that the delay is practically linear with respect to both the amplitude and the width of the noise pulse, the above equation can be approximated to

$$D \approx D_0 \cdot [1 - (f_1 + g_1) \cdot N_a \cdot N_w] \tag{2.15}$$

where f_1 and g_1 are the first order Taylor expansion coefficients.

With multiple power noises, the delay is expressed by (2.16).

$$D = D_0 \cdot [1 - (f_1 + g_1) \cdot \sum_i (N_{a,i} \cdot N_{w,i})] \tag{2.16}$$

D has a Gaussian distribution by the central limit theorem.

3. Temperature variation vs. delay

When a signal propagates, the transistors which pass the signal radiate thermal flux mostly from the junctions. The interconnects that carry the signal also radiate heat because of Joule heating. The path delay dependence on the temperature also needs to be considered because high temperature can play significant role in the delay error generation. Deep-submicron technologies enable higher packing circuit

density and that results in a higher heat generation per unit area. Let's consider a small area S on the device with N transistors and M interconnects. In every clock period, the transistors and interconnects that pass the signal emit heat and its conduction/dissipation to adjacent area determines the device temperature. For transistor i $(1 \leq i \leq N)$, the thermal emission during clock period t_k is $R_{tr,i}(t_k)$. For the interconnects, it can be expressed in similar fashion. $R_{in,j}(t_k)(1 \leq j \leq M)$ is the thermal emission during the kth clock period, or at time t_k. The temperature of the S is thus expressed by (2.17)

$$\frac{dT_s(t_k)}{dt} = \frac{\sum_{i=1}^{N}[R_{tr,i}(t_k)] + \sum_{j=1}^{M}[R_{in,j}(t_k)] - C[T_s(t_k) - T_e(t_k)]}{Q} \tag{2.17}$$

where C is a constant for thermal conduction to adjacent area, T_s is the temperature of area S, T_e the temperature of adjacent area, and Q is heat capacity of S.

Summing up the right side of (2.17) from t_1 to t_k gives the temperature of area S, T_s, at time t_k. It is assumed that the temperature variation on the chip is gradual or the temperature difference $T_s - T_e$ is small and constant

$$T_s(t_k) = \frac{\sum_{l=1}^{k}[\sum_{i=1}^{N} R_{tr,i}(t_l) + \sum_{j=1}^{M} R_{in,j}(t_l) - C']}{Q} \tag{2.18}$$

In (2.18), using the Central Limit Theorem, we can observe that the temperature T_s at time t_k has a normal distribution, $f_{T_s}(t_k)$, because summation of each $R_{tr,i}(t_l)$ and $R_{in,j}(t_l)$ is assumed to be random. The number of clocks in consideration is order of 10^9 because the device is clocked at about GHz range. And one additional assumption about the cooling system is made. If the cooling system is efficient enough to ensure that the probability of S having temperature higher than arbitrary temperature T, is same all through the time t, then it can be assumed that $f_{T_s}(t_k)$ is approximately equal to $f_{T_s}(t)$ for arbitrary time t. To find the effect of temperature

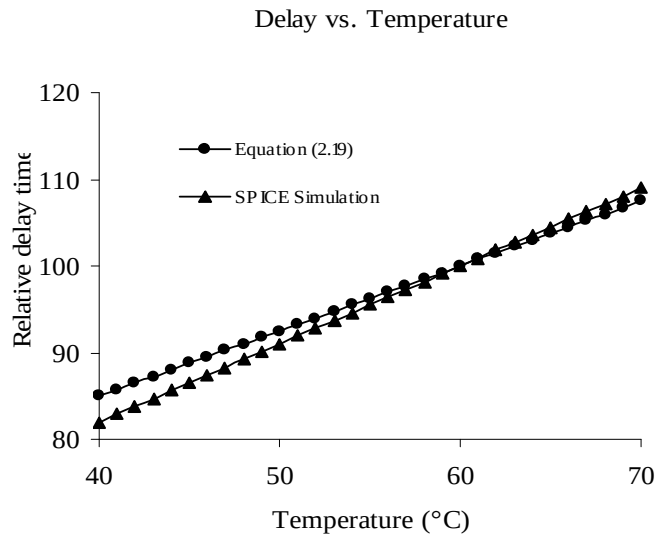


Fig. 4. Delay vs. temperature

distribution on the delay, a simple model using the propagation delay time in (2.19) is used, and SPICE simulation using TSMC 0.18 μm CMOS technology file is conducted. The results show that the aforementioned assumptions that the temperature is linearly related to delay within the temperature range of interest. In the model, the combination circuit (CC_i) is replaced by an array of invertors. The propagation delay times for low-to-high and high-to-low are calculated in the following equation,

$$\int dt = -C_{load} \int \frac{1}{I_{D_n,D_p}} dV_{out} \tag{2.19}$$

where $I_{D,n}$ is for the calculation of high to low delay time and $I_{D,p}$ for low to high delay time. The power supply voltage is set to 1.6 V and the widths of the transistors are adjusted to yield the same low-to-high and high-to-low transition time. The calculation result is shown in Figure 4.

With the temperature ranging between 40 ^0C and 70 ^0C, the delay appears to be linear and it grades at 0.75 $\%/^0C$. The result of SPICE simulation with TSMC 0.18 μm CMOS Technology model parameters also shows that, for those temperature

ranges, it is linear with the rate of 0.9 $\%/^0C$.

If only the combinational circuit is considered, these simulation results show clearly the linear relationship between the propagation delay and temperature for the range of our interest (realistic device temperature ranges). The simulation results support the linear extrapolation in estimating delay of different temperature from those of experiments.

In this Section, the effects of noise factors on delay are analyzed to be a Gaussian. But as the number of individual delay contributors increases, regardless of the underlying distributions, the law of large number [43] and the central limit theorem [39] enable the assumption of Gaussian delay distribution for each path.

E. Simulation

By integrating the factors affecting timing during actual operation into the delay model, the delay distribution of each path can be expressed as a Gaussian distribution as described in the previous Sections. Thus, (2.9) is now expressed as:

$$P_{df}(t_p) = \int_{t'=t_p}^{t'=\infty} \sum_{i=1}^{n} [\sum_{j=1}^{m_j} \{G_{i,j}(t') \cdot p_{ex}(i,j)\} \cdot P_{pp}(i)] dt' \tag{2.20}$$

where $G_{i,j}$ is a Gaussian delay distribution for each path.

To calculate $P_{df}(t_p)$, the parameters of all these Gaussian distributions and probabilities for each path need to be obtained. This is impractical, considering the complexity of a circuit. In delay failure rate estimation, the focus is on the tail part of distribution in (2.20) which corresponds to extremely low delay failure probability. Hence, to empirically estimate the delay failure rate, an additional assumption is made on the (2.20), that is, the summation of the Gaussian distributions can be

approximated to single normal distribution.

$$\sum_{i=1}^{n}[\sum_{j=1}^{m_j}\{G_{i,j}(t')\cdot p_{ex}(i,j)\}\cdot P_{pp}(i)] \approx N(m,\sigma:T,V) \tag{2.21}$$

To validate this assumption, Monte-Carlo simulations are carried out and the results are in the Figure 5. For the case of n=5 stages of pipelined circuit, m_j=100 means to form the Gaussian distributions per stage are generated. Importance Sampling is used to select simulation runs. Five runs are made from randomly generated means: random1 to random5 in the figure. To cover the case of circuit optimization where the delay of circuits is clustered around the longest delay, 50 means are sampled from 90% to 100% range of the maximum path delay and 50 means from less than 90% of the maximum path delay (50_50_1 to 50_50_5 in the figure). The standard deviation for each distribution is selected randomly but to be proportional to the mean in the ranges of 5% to 15% of the mean. From the summation of the Gaussian distributions, the mean and standard deviation of the assumed normal distribution is extracted numerically at 7 standard deviations and at 7.1 standard deviations because these numbers correspond to the clock periods of the experiments. The extracted means and standard deviations are used to calculate the probabilities for 8, 8.5, and 9 standard deviations because these points correspond to the nominal operating clock periods of the CPU chips used in the experiments. In the graph, approximate values and summation values are defined, respectively, as Zs are 8, 8.5, or 9 in (2.22). Most of the cases the errors are less than 7 %.

$$\begin{aligned} Summation: & \int_{t'=m+Z\cdot\sigma}^{t'=\infty}\sum_{i=1}^{n}[\sum_{j=1}^{m_j}\{G_{i,j}(t')\cdot P_{ex}(i,j)\}\cdot P_{pp}(i)]dt' \\ Approximation: & \int_{t'=m+Z\cdot\sigma}^{t'=\infty} N(m,\sigma:T,V)dt' \end{aligned} \tag{2.22}$$

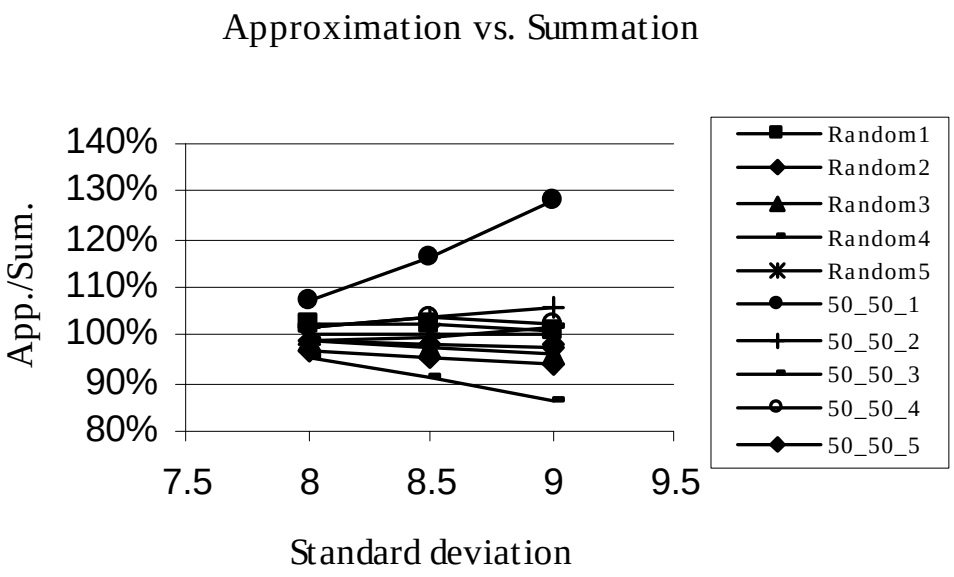


Fig. 5. Comparison of approximation and summation

F. Experiment

We consider only single-point fault because near-coincidental multiple delay faults resulting in a failure is unlikely. This is a resonable assumption because the probability of a sigle fault/failure is extremely small. A signal propagation path is considered to be a chain of combinational circuits and registers, which is a pipeline. Pipeline is of our primary interest because, typically, the timing of a processor is limited by the delay across the combinational circuit between latches. Figure 6 shows three possible scenarios of delay fault propagation.

In the path III of Figure 6, a delay fault is masked from propagating and subsequently becoming a failure is prevented. Remaining two scenarios describe conditions prevailing when failure at the primary output is observed. Delay fault can propagate directly to the output (path II) or it may propagate and momentarily reside in cache/memory (path I) and result in latent fault. For exact delay fault estimation, both above cases need to be considered. There have been several researches on the

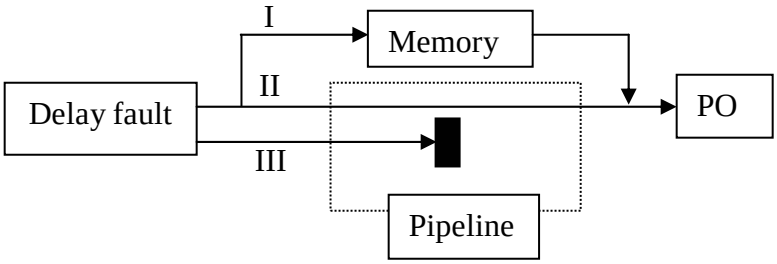


Fig. 6. Fault propagation

fault latency, the time between the occurrence of a physical fault and the subsequent corruption of data causing an error [44]. In our experiments only the paths II and III are considered. This is because the application execution time for each experimental run is in order of several seconds while the MTTF is in order of hundreds if not thousands of seconds. Thus errors that become latent beyond each experimental run are not observable and neglected. While this omission may yield slight underestimation of failure data, yet would not significantly change the overall distribution.

Hence the delay failure rate distribution is modeled to be Gaussian in the Section D and E. The parameters of the distribution can be extracted by the delay failure rates at different clock frequencies. The delay failure rate distribution parameters at different temperatures can also be combined to be extrapolated to estimate delay failure rate distribution of normal operating conditions.

In the experiment, the mean-times-to-delay-failures at various operating conditions are measured. The clock frequency and temperature are varied to permit the delay failures observable. For the experiment, the test system is configured as follows:
- CPU : AMD Duron 800 MHz. (FSB 100MHz x 8)/ 750 MHz. (FSB 100MHz x 7.5)
- Main Memory : DDR 2100 MHz 128 MB
- Graphic Card : ATI Raze Iic AGP

- Operating System : Windows 2000^{TM1}

The clock frequency is controlled by Fuzzylogic^{TM2}. software provided by the mainboard manufacturer. The temperature of CPU is controlled by the thermoelectric cooler. Other peripherals on the mainboard are isolated from the temperature changes by exposing them to the ambient air. The temperatures are measured using a sensor beneath the CPU. A probe is put beside the CPU and its thermal resistances are measured for reference. In the experiment, the application execution determines the excitation probability and is carefully constructed to maximize the error detection/observation probability. The test program consists of 3D graphic routine provided by the mainboard vendor, and it is used to stimulate the circuit to propagate the delay faults. The program is specifically designed to be a CPU-intensive. It is intended to check for CPU errors during increased-frequency testing. To prevent the operating system from interrupting and triggering the CPU into a standby state, another 3D graphic software is run in the background. The CPU usage during the experiments is monitored to be 100% by Windows Task Manager and Speedfan 4.2^{TM3}.

Two issues are carefully considered during the experiments: (1) Are the failures observed during the experiment actually resulting from delay faults? (2) If these are delay failures, what may be the component source of these failures? We conducted the experiments at a specific temperature. The only control variable is hence the clock frequency. If failures originate from fault sources other than delay faults, the TTF distribution would be the same as that generated from another experiment set at different clock frequency. The result is carefully analyzed and the failure distributions

[1]Windows 2000 is a registered trademark of Microsoft Corporation.

[2]Fuzzylogic is a registered trademark of Microstar International Corporation.

[3]Copyright 2000-2007 by Alfredo Milani Comparetti.

are obtained that follow the frequency dependency. Secondly, the experiment is designed to isolate only the CPU chip as target component that could have delay faults. We ensure this by conducting elevated frequency testing of all other components in the system except CPU. Then, during the experiments, we ran the rest of the system at or lower frequency to further reduce the chance of having faults originate from any components other than the CPU chip. This isolation test is necessary because, when the front-side bus frequency of the chip is adjusted during the experiment; other peripheral components may also be affected. Isolation test confirmed the fault silence for varying FSB frequencies; no fault was detected during isolation test. During the isolation test, the clock multiplier for the CPU unit is set to a low value. In the experiments, the effects of temperature also make it certain that the delay failures are of CPU origin because other devices on the mainboard are cooled to specified levels. The memory clock is controlled independently and is set at conservative level to make certain that no memory error would occur.

The test control flow is set up as Figure 7: First, the background program that preempts the standby-state interruption is spawned. This program is in continuous loop and generates graphical output in the background. Next comes the temperature setting. To achieve steady CPU temperature, the temperature is monitored for 10 minutes. The supply voltage/current to the thermoelectric cooler is determined by trial to the delay experiment clock frequency. The FSB clock frequency adjustment follows for the target frequency. The clock frequency adjustment affects the thermal generation and thus changes the temperature of CPU die. We introduce a control loop to change dynamically the heat dissipation capacity through the thermal electric cooler. This permits controlled temperature of the IC throughout the experiment period. The clock frequency used during the set-up process is chosen from ad-hoc experiments/trials to guarantee that the CPU is free-of-error during the speed ramp

```
Begin
Step1.
  - Run graphic program
  - Start Temp. monitor
Step 2.
  - TE Cooler V/I adjustment
  - Wait time Time1
Step 3.
  - if (Temperature =TTarget)
      {Goto Step 4.}
    else
      Goto Step 2.
Step 4.
  - Adjust Frequency
  - if(Any Error)
      {Record the time and error message;
       Stop;}
```

Fig. 7. Experiment flow

process. According to manufacturer's data sheet [45], the thermal dissipation variation by the clock frequency change is expected to be less than 1 % in case of 860 MHz for temperature set-up and 870 MHz for delay experiment. When a failure is observed, the log file generated by operating system is checked to get time to failure and error.

G. Experimental results

TTF distributions for two CPUs (A and B) are measured. With CPU A, the experiments are conducted at two different temperatures at two different frequencies to generate data on the delay-temperature dependencies. With CPU B, the experiments are conducted at three different frequencies to validate the Gaussian-distribution assumption associated with the delay failure rate distribution. The number of measure-

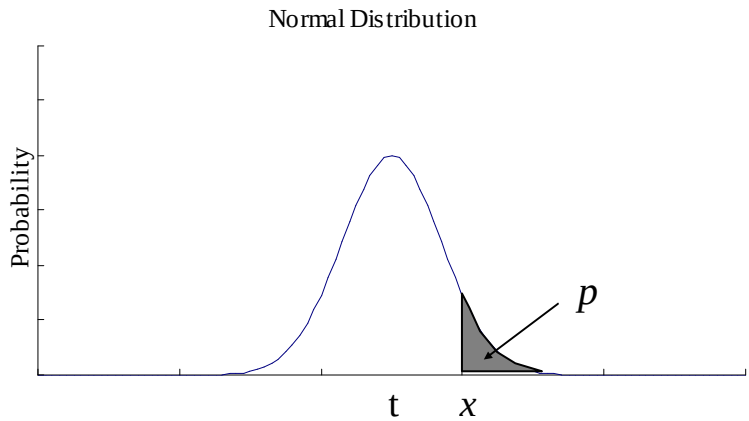


Fig. 8. Normal distribution

ments for TTF varies from 25 to 70 according to the shape of measured data. (2.23) shows the meaning of x and p used in this section. x is a normalized parameter in units of standard deviation. Note that if x is bigger, the probability p is smaller as can be seen in Figure 8.

$$\int_{x}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-t^2/2} dt = p \tag{2.23}$$

For CPU A, the mean and standard deviation of the delay failure rate distribution at 56°C and 60°C are extrapolated to be (mean, standard deviation)=(1015 ps, 26.2 ps) and (1023 ps , 26.3 ps), respectively based on the data shown in Table I. Considering that the CPU is rated to operate at 800 MHz (clock period=1250 ps), it is expected that the majority of failures are observed at approximately 9 standard deviations away from the operating clock period. The analysis of the means and standard deviations from the two empirical distributions taken from the two temperature points show that the standard deviation/mean ratio is within 2.5% of one another. The MTTF is observed at the time point when 63 % of the failures occur cumulatively and the MTTF of the CPU A is calculated to 1.14E5 Hr at 60°C and 2.33E6 at 56°C. Assuming the linear delay-temperature dependency, the MTTF at 50°C, 800 HMz,

Table I. The results of CPU A experiments

Experiments	No. of Data	Test Frequency (MHz)	MTTF(sec)
60^oC /1.6V	69	828	982
	40	834	97
56^oC /1.6V	34	832	2714
	32	834	1229

Table II. The results of CPU B experiments

Test ID	Test Frequency (MHz)	Time (ps)	MTTF (sec)	Std Dev.(x)
I	867	1153.4	4621	7.225516
II	872	1146.8	2232	7.126760
III	875.5	1142.2	1388	7.061611

and 1.6 V is calculated to be approximately 2.6E8 Hrs. On these experimental results for time-to-failure distributions, Anderson-Darling goodness of fitness (GOF) test is conducted to validate the assumption of (2.11), i.e, these cumulative distributions are exponential.

To validate the Gaussian characteristic of the delay failure rate distribution, one additional set of experiments are conducted at three different clock frequencies and the results are shown in Table II. The CPU is rated to operate at 750 MHz which corresponds to clock period of 1333 ps.

Any two data set combination can yield the standard deviation and mean of the underlying delay failure rate distribution as illustrated in Figure 9. From the three data points, three sets of standard deviations and means are calculated in Table III.

Table III. Estimated delay failure rate distributions for CPU B

Data Sets	Mean	Std.Dev	Delay failure rate at 750MHz
(I,II)	670	66.8	1.83E-23
(II,III)	643	70.6	8.04E-23
(I,III)	659	68.3	3.30E-23

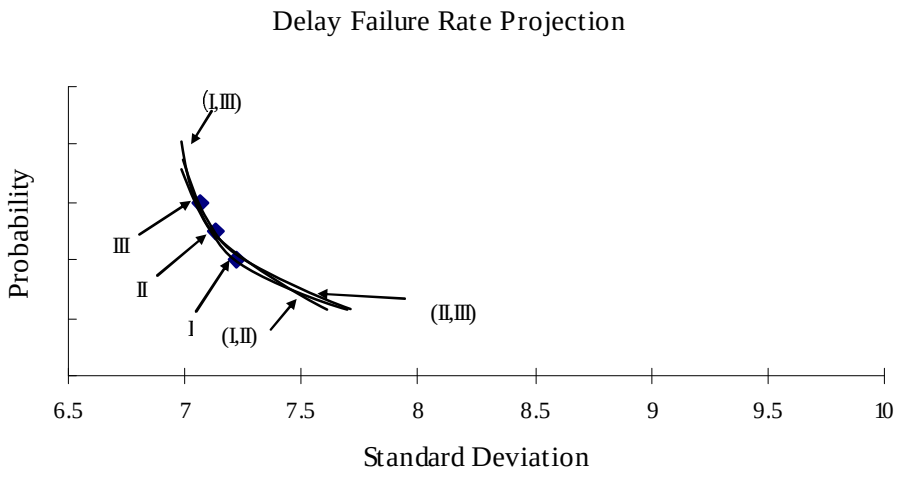


Fig. 9. Delay failure rate projection

The variation among standard deviation is ± 2 ps and ± 23 ps for the mean. This validates that the underlying distribution is Gaussian within the bounds of error.

The standard deviations on the data sets obtained from CPU B are about 10% of their means and these are larger than that of CPU A. The discrepancy may be from the variations in the experiment conditions. For CPU B experiments, the voltage adjustment range to keep the temperature constant is larger than that of CPU A experiments and it may be attributed to the slightly elevated room temperature during the experiments and inaccurate temperature measurements of the CPU.

Some of experimental results for CPU A are shown in Figure 10 with fitted curve.

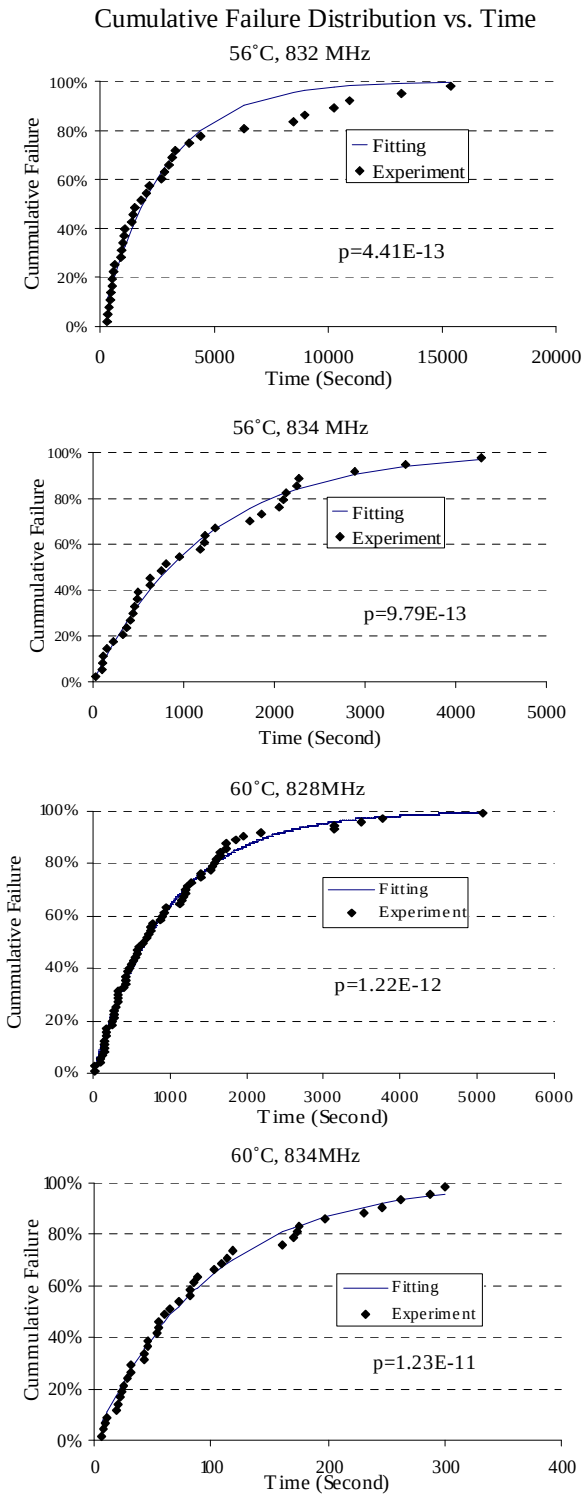


Fig. 10. Cumulative failure distribution vs. time

H. Conclusion

The operating noises bring increasingly more ambiguity in timing estimation of design as the feature size of technology is decreased and the power supply voltage is reduced. Without field-based, practical timing estimation, the chance of overdesign and underperformance may increase in the future as reliability becomes more concerns.

In this chapter, a methodology for estimating the timing of integrated circuits is presented. First, delay failure rate distribution is modeled starting from discussions of delay distribution and related probabilities. Next, the effects of operating noises factors like local thermal emission, power supply noise, and crosstalk on delay distribution are analyzed and used for parameterization of delay failure rate distribution of CPU. The contributions from the noise factors on delay are analyzed to be Gaussian and the resulting delay distribution of a combinational circuit is Gaussian. Because the delay failure rate distribution is modeled as a product of delay distribution and probabilities related to signal propagation, the overall delay failure rate distribution is assumed to be a Gaussian distribution and a set of Monte-Carlo simulations are conducted to validate the assumption. Delay failure rate distribution at normal operating condition is finally estimated by combining the model and experimental results. To observe timing failure, which is extremely small in normal operating condition, stress test technique is developed and experiments are conducted using commercial CPU chips. By aggregating the empirical time-to-failure distributions at various temperature-clock combinations and fitting them to the cumulative delay failure curve, the delay failure rate at test condition is estimated.

From these delay failure distribution, the delay failure rate at normal operating condition is estimated to correspond to approximately 2.6E8 Hrs of MTTF for 63 %

failures which is significantly greater than the typically specified VLSI device MTTF of 1E5 hours by the manufacturer. This estimation may be limited by the fact that the device degradation over long period of time is not considered in our stress tests. In the failure rate calculation, the tail part of the distribution is used to extrapolate the normal operating condition results and it leaves a possibility that small changes in the distribution parameters caused by the aging effect of the device may result in a significant failure rate increase.

CHAPTER III

APPLICATION OF ESTIMATED DELAY DISTRIBUTION TO FMAX DISTRIBUTION

A. Introduction

Ensuring the timing of integrated circuits (ICs) is becoming increasingly difficult as the market trends ask power-economic, high performance products. The factors affecting the timing of ICs can be classified into two groups; the ones related to fabrication process and the other related to operating conditions. Traditionally, within-die (WID) variation has been neglected and die-to-die (D2D) variation has been emphasized [46]. As the feature size of VLSI technology decreases below the wavelength of light source used in stepper of lithography process, the WID fluctuations in defined pattern are comparable to the D2D fluctuations [7]. Hence their effects on timing have been widely studied [47][48][49].

The operating conditions variations as well as crosstalk can affect the signal propagation time of fabricated ICs. As the power supply voltage decreases, the signal integrity is more vulnerable to any noise in signal propagation. In reducing feature size of fabrication technology, the vertical dimension has not shrunken as much as the horizontal dimension has and it results in crosstalk problems from the surrounding circuits [36][50]. Thermal dissipation per unit area is rapidly increasing and its non-uniform distribution has a significant effect on timing distribution of a circuit [51]. Non-uniform power consumption in power grid can cause non-uniform voltage drop which has a direct effect on signal timing [52].

The accurate estimation of factors affecting timing of ICs is critical in manufacturing economics. The overestimation of them may result in more complicated design

of longer design time, accordingly bigger development cost, or rejection of acceptable design [6]. On the contrary, the underestimation causes rejection of fabricated products by not sufficing the specification. This chapter presents the operating related noises' effects on timing of ICs in terms of the maximum operating clock frequency (FMAX) distribution. In Section B, previous works related to device parameter fluctuation's impact on FMAX distribution are described. The FMAX distribution considering operation related noises are presented in Section C and followed by discussions in Section D and conclusion in Section E.

B. Previous work

In this Section, previously published FMAX distribution model considering process parameter variation impact by K. A. Bowman et al [5], is reviewed. The timing characteristic of ICs is determined by critical paths which have the longest signal propagation delay and the number of the critical paths is increasing with process miniaturization. In the paper, FMAX distribution of 0.25 μm technology microprocessor is studied. First, the WID and D2D critical path delay distribution under process variation is assumed to be normal with different standard deviations and statistical simulations are conducted to obtain the normal distribution parameters, mean (μ) and standard deviation (σ). The ratio of σ to μ is 3% and 9% for WID and D2D variations, respectively. Next, for N_{cp} independent critical paths for entire chip, the maximum critical path delay distribution is calculated for WID and D2D parameter variations and combined. Finally, the maximum critical path delay distribution is converted to the maximum operating clock frequency (FMAX) distribution. The equations used in the study are illustrated in Figure 11[5]. b in (g) is the clock skew factor.

(a) D2D & WID Nominal critical path delay (Tcp,nor) density functions

$$f_{D2D-Tcp,nom} = N(T_{cp,nom}, \sigma^2{}_{D2D-Tcp,nom}),\ f_{WID-Tcp,nom} = N(T_{cp,nom}, \sigma^2{}_{WID-Tcp,nom})$$

(b) WID Nominal critical path delay cumulative distribution

$$F_{WID-Tcp,nom}(t) = \int_0^t f_{WID-Tcp,nom}(t')dt'$$

(c) WID Maximum critical path delay density function

$$f_{WID}(t) = N_{cp} \cdot f_{WID-Tcp,nom}(t)(F_{WID-Tcp,nom}(t))^{Ncp-1}$$

(d) D2D & WID Maximum critical path delay density functions shifted by Tcp,nom

$$f_{\Delta T_{D2D}} = N(0, \sigma^2{}_{D2D-Tcp,nom})$$

$$f_{\Delta T_{WID}}(t) = N_{cp} \cdot f_{WID-Tcp,nom}(t - Tcp,nom)(F_{WID-Tcp,nom}(t - Tcp,nom))^{Ncp-1}$$

(e) Nominal critical path delay impulse function

$$f_{Tcp,nom} = \delta(t - T_{cp,nom})$$

(f) Combined (D2D & WID) maximum critical path delay density function

$$f_{Tcp,\max} = f_{Tcp,nom} * f_{\Delta T_{D2D}} * f_{\Delta T_{WID}}$$

(g) Maximum clock frequency density function

$$f_{Fclk,\max}(b/t) = f_{Tcp,\max}(t)\frac{t^2}{b}$$

Fig. 11. FMAX distribution model from [1]

Even though the result shows that the FMAX distribution matches well with measured data, for more accurate estimation, the effects of operation related noises like crosstalk, variations in power supply voltage and temperature need to be considered.

C. Operation noises and their effects on delay failure rate distribution

The signal propagation time of ICs are affected by operating noises. When there is a signal transition in a node, the transition delay time may vary according to the transition direction (high-to-low or low-to-high) of the neighboring circuits and the coupling capacitances between the node and the circuits. While the signal integrity is becoming more vulnerable to crosstalk with decreasing feature size of circuits, the penalty in increasing die area limits the usage of available options like spacing and shielding to remedy crosstalk problem [53]. The local temperature variation also affects device performance by changing the physical characteristics of material. There can be a large temperature difference on a chip because the power consumption density is not uniform. The potential level of power network varies when the transistors are switching and the switching noises can affect the signal propagation time. All these operating noises should be included for more accurate timing estimation of ICs.

From the experimental results of chapter II, Table IV shows the extracted delay failure rate distribution parameters. Two CPUs are used for experiments and the σ to μ ratios are 2.6% and 10.4% respectively.

D. Combination of process variation and operating noises

In [5], the WID and D2D process variations in 0.25 μm technology result in the normal distributions of the critical path delay with 3% and 9% of σ/μ ratio, respec-

Table IV. Delay distribution parameters

CPU	A	B
Bin	Duron 800	Duron 750
Temp.	60°C	60°C
Mean, μ (ps)	1023	657
Std.Dev., σ (ps)	26.3	68.5
σ/μ	2.6%	10.4%

tively. If the operating noises' effects on delay are considered, the delay distribution of the critical paths is affected in WID level. Let's assume that the noises' effects on delay form a normal distribution, $N(0, \sigma^2_{ON})$, on average and they are independent of the process variations. Then, the equation (a) in Figure 11 is modified to (3.1) to include operating noises' effects in WID critical path delay fluctuations.

$$f_{WID-T_{cp,nom}} = N(T_{cp,nom}, \sigma^2_{WID-T_{cp,nom}} + \sigma^2_{ON}) \tag{3.1}$$

Then, the result of CPU A can be interpreted as a case where the process variations are minimal and that of CPU B as maximal. Hence, value of σ^2_{ON}/μ^2 is 0.0262^2 and, from the result of CPU B, σ^2_{WID}/μ^2 is calculated as 0.1042^2-0.0262^2=0.102^2. If these values from operating noises are compared to performance variation, 10% WID process variation is a lot bigger than 3% WID process variation of 0.25 μm technology even the CPUs are fabricated using 0.18 μm technology which may have wider process variation than those from 0.25 μm technology. The results can be explained by the experimental data used for delay distribution parameter extraction. In the measurement of TTF distribution, the failures are resulting from all the paths not only the critical paths considered in [5]. As the process variation mixes the critical

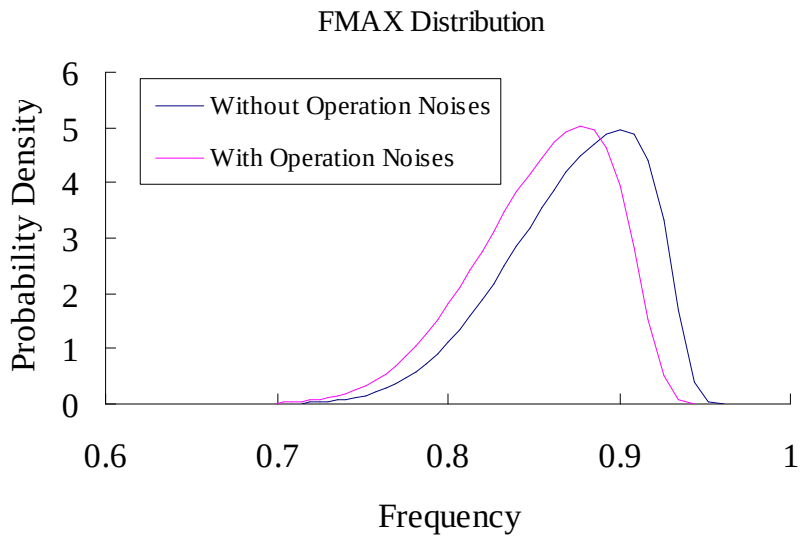


Fig. 12. FMAX distribution

paths with other paths, the WID variation may look wider than that of the critical paths. These two CPUs are from the same technology (Duron Spitfire, Model 3, 0.18 μm technology) and the relatively wider process distribution may contribute to the lower speed binning of B than that of A. The maximum critical path delay distribution and FMAX distributions with and without considering the operating noises are shown in Figure 12 and Figure 13. The peak of FMAX distribution shifts about 2% to lower clock frequency because of operating noises.

E. Conclusion

The modified FMAX distribution of a chip with operating noises in the field is presented. To empirically estimate the noises' effects on signal propagation delay, experiments are conducted to measure TTF distribution of commercial microprocessors in real program operation stress. The MTTF extracted from the measured distribution is used to calculate delay failure probability, P_f, and, with assumption of Gaussian

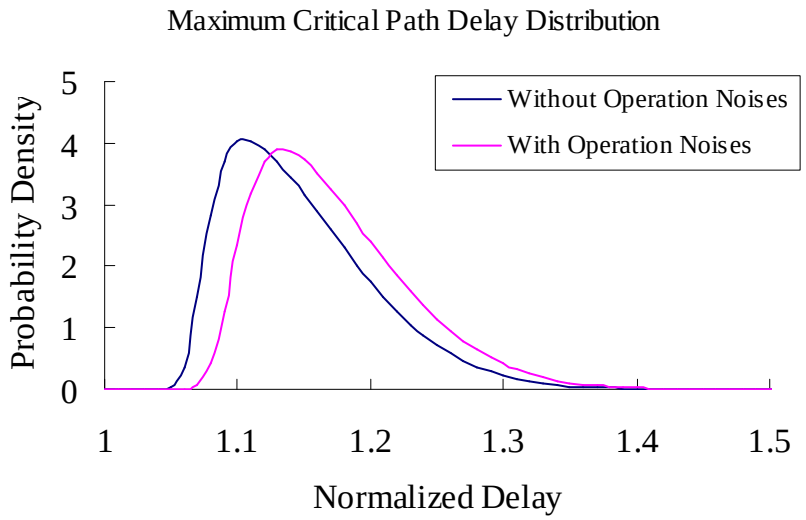


Fig. 13. The maximum critical path delay distribution

delay distribution, the parameters of normal delay distribution are calculated. The estimated noises' effects on delay have 2.6% of σ/μ value and it is comparable to 3% resulting from WID process variation. The peak of FMAX distribution is degraded about 2% because of the operating noises.

CHAPTER IV

SABRE: DELAY FAULT TOLERANT DESIGN *

A. Introduction

To guarantee quality of devices, various faults models are developed to direct the test. For example, the detection and removal of stuck-at-faults [54] have been successful owing to models and efficient test vector generations [55] even with increasing complexity of devices. On the other hand, faults related to timing are becoming more and more difficult to handle because of (1) exponentially increasing number of target paths and (2) various hard-to-model factors like crosstalk and operating conditions. The limited generation of robust test vector also restricts the coverage of the test. In commercia practices, these insufficiencies are relieved by timing margin in binning. However recent works on multi-core and/or multi-threading architecture implies that the raising operating clock frequency reached realistic boundary even with 30 stages of pipelined structure [56].

The operating clock frequency is determined by the longest signal propagation delay, setup/hold time, and margin for timing abnormalities. These are becoming less predictable with the increasing design complexity and process miniaturization. Hence aggressive operating frequency binning of devices is unlikely in the future and devices are likely to suffer from increased delay errors/failures. The difficult challenge is then ensuring that a device operating at its clock frequency is error-free with quantifiable assurance. Effort at device-level engineering will not suffice for these circuits exhibiting wide process variation and heightened sensitivities to operating

*Based on "Gate-Level Exception Handling Design for Noise Reduction in High-Speed VLSI Circuits," by Sanghoan Chang and Gwan Choi which appeared in 20th International Conference on VLSI Design, Jan. 2007. ©[2007] IEEE

condition stress. Logic-level redress of this issue is a necessity and we propose a design-level remedy for this timing-uncertainty problem.

The aim of the design approach presented in this chapter is to provide framework, SABRE, wherein an increased operating clock frequency can be achieved. The approach is a combination of empirical analysis, hardware/time redundancy design, exception handling and retry. One redundancy-design approach is coding. Often in memory systems, error checking and correcting (ECC) schemes are employed to filter the soft errors caused by cosmic rays [57]. Another example of such is a triple modular redundancy (TMR) where multiple units of identical hardware system are used in addition to a voting circuitry. Typically the overhead of a TMR system ranges from 400-500%. However, such scheme will still not tolerate errors resulting from common stress or design-induced faults. In general, applying any redundancies to a system has been considered too excessive and expensive unless for a reliability-critical application.

Our proposed design relieves such shortcomings of traditional redundancy techniques by replicating only a part of the original circuit instead of the entire circuit. The timing-critical combinational circuits between the registers are partially replicated. Combinational circuit is path-wise partitioned into two sections; long paths are segmented into the two partitioned sections. The original combinational circuit is laid out and operates without any intrusion except for the fan-out connections from the first section of the circuit to a replicated second-section of the combinational circuit. Thus only the second section of the circuit is replicated. The signals fanning out from the first section are latches, and thus are far shorter than the paths spanning the entire combinational circuit. The replicated circuit is timed at a subsequent clock cycle to ascertain relaxed timing paths. This insures that the likely hood of delay, stress, or process variation faults is minimal. During the subsequent clock cycle the

outcome of the two logically identical, yet time-interleaved, circuit outputs are compared to detect faults. When a fault is detected, the retry signal is triggered and the dynamic frequency step down takes place before a pipe flush, and retry takes place. The significant timing overhead associated with the retry is offset by the rarity of the timing violation events.

In this chapter, the research motivation is explained in Section B and delay fault probability is introduced in Section C. Proposed design scheme is described in Section D. Finally, the conclusion is drawn in Section E.

B. Motivation

The delay fault or longer path delay than the operating clock period can cause a delay error/failure. The randomness in the nature of factors affecting delay makes it difficult to precisely estimate the path delays. For example, the crosstalk can give 10% to 15% delay variation in a segment according to the input vector to the circuit [37]. Besides, the variations in the operating conditions like on-die local temperature, supply voltage add the uncertainty. This uncertainty limits the possible clock frequency gain by transistor performance improvements. Even extremely small chance of very long path delay can not be ignored in determining the operating clock frequency of devices without delay fault tolerant scheme. But if the occurrence of the path delays around the maximum path delay is low enough, it is possible to increase the clock frequency with a reasonable cost of additional delay fault tolerant circuit.

There have been several studies on the path delay distribution of a circuit. Because of the randomness in delay of path segments, with the help of the central limit theorem, the path delay distributions are often assumed to be a Gaussian [6][8]. Even for a single delay path, the signal propagation delay can have a Gaussian distribution

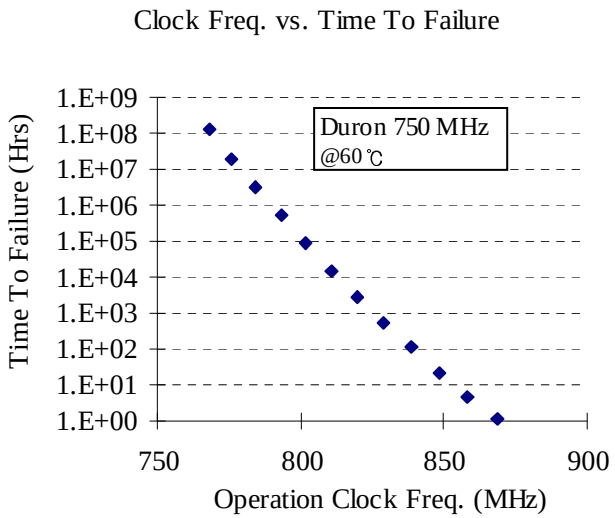


Fig. 14. Operating clock frequency vs. time to failure

in statistical approach because the effects of individual factors like crosstalk, temperature, and power supply voltage add Gaussian variations. Based on the Gaussian path delay distribution assumption, a research is carried out to estimate the delay failure rate of a 750 MHz Duron CPU at various clock frequencies as described in chapter II. Figure 14 shows the expected MTTF for different operating clock frequencies. For example, if a logic circuit to handle a failure every hour is inserted, the CPU can run around 870 MHz which is about 16 % increase compared to the original binning of 750 MHz. In developing the proposed scheme, there are several issues to be considered. First, the added logic for fault detection and removal should have a reasonable complexity and avoid any influence on the existing logic operation. Second, the delay fault detection scheme should guarantee the detection of the target level of delay fault occurrence. For example, in a circuit with n signal propagation paths and d_i for i_{th} path's delay, the fault probability at clock period t_p, $P_f(t_p)$, is

given by below equation (4.1).

$$P_f(t_p) = Probability\{\max(d_i) > t_p\} \tag{4.1}$$

If the target level of delay fault to detect is P_T and the corresponding path delay is t_T, the delay fault detection circuit should detect any path delay longer than t_T. Third, in case of error/failure, the recovery should be done in acceptable time budget. Otherwise, the performance gain by the increased clock frequency can be compensated by the process time for recovery. Because the required fault-tolerant system should handle timing related failures in over-clocked, fault-prone operation, the normal fault-tolerant techniques can not be used. For example, TMR should run at minimum clock frequencies of the three systems. Otherwise, there maybe multiple systems with timing faults and the voting can not be used to recover the correct processing result. Hence, the fault-tolerant system should secure the timing margin enough to guarantee the correct reference signal for fault detection and recovery while it process the normal operation with less timing margin.

In adding the timing margin to reference signal path, two schemes are possible, one which adds registers to sample the reference signal at generous clock frequency while keeping the same signal propagation path and the other which adds margin in signal propagation path by dividing the path and process the latter in the next clock cycle. One example of the former approach is RAZOR [58] [59]. In RAZOR, redundant latches (shadow latches) are connected to the primary outputs together with the original flip-flops. Delayed clock is applied to the shadow latches to provide the necessary timing margin for delay-fault free signal propagation. The sampled result of shadow latch is compared with the original result stored in register and if they are different, a timing fault is detected and the sampled value at the shadow latch is fed back to the primary output through a MUX. Figure 15 illustrates the idea

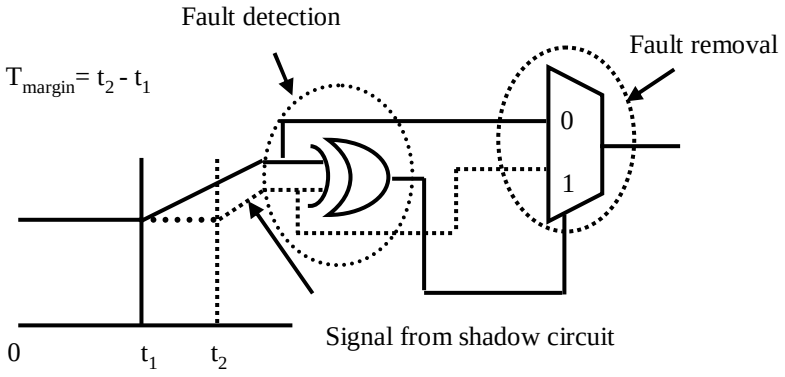


Fig. 15. RAZOR

of RAZOR [22].

The RAZOR scheme is used to detect and recover timing faults in a dynamic voltage scaling processor to save power consumption by running the chip at supply voltage corresponding to optimal failure rate. As the power supply voltage lowers, less energy is consumed and more delay failures are expected because of insufficient current drivability of device. It is shown in [59] that the area burden by the redundant shadow latch is negligible compared to the area of the original design and it can achieve up to 50% of energy savings over worst case operating conditions at a frequency of 120 MHz. As maintaining two separate clocks can be an excessive overhead, to relieve this problem, using the negative edge of the original clock to trigger the shadow latches is proposed [59]. In this case, the duration of positive clock phase determines the timing margin and it may not have a stable value, which, in return, unstable delay fault detection coverage. Another issue in RAZOR design is that, the shadow latches are supposed to sample the signal from the previous clock but, for paths with small delay, they may sample the signal from the current positive clock edge. So a timing constraint for the minimum path delay is imposed as Figure 16 [58]. If inverted clock is used for clock_del as [59], the minimum path delay should be greater than

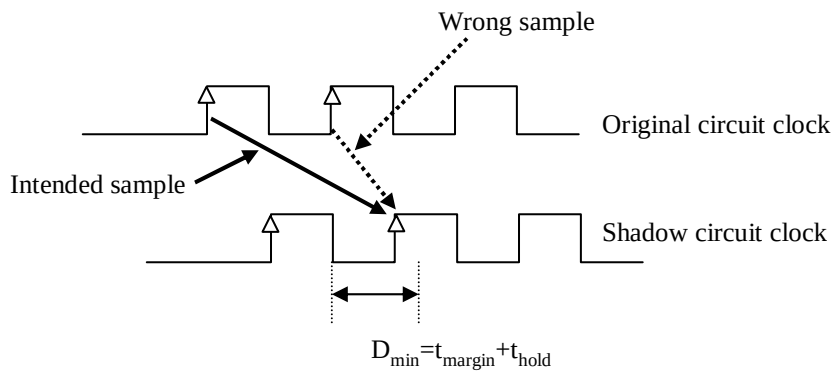


Fig. 16. Short paths constraint

50% of operating clock period and it may impose some limitations for the design's application.

Another design approach we took for delay fault tolerant design is to use parity checking circuits. The main idea is that in circuits, there are paths with long delay and with short delay and if the long paths and short paths are grouped to generate parity code, it can function as a fault detection circuit because the signal in short path will play a role of reference signal. However, if they are closely related in timing, there is a chance that both will fail due to the same delay fault. To avoid this situation, the correlationships between each output signal are investigated. The parity generation circuit is divided into 2 parts and the second part is placed in the next pipeline stage to make sure enough timing slack in signal propagation.in parity generation circuit. In parity generation circuit, it should include all the function of the original circuits and it makes the area overhead comparable to that of the original circuit.

In this chapter, a design approach to avoid the fore-mentioned issues by adding timing margin in the reference delay path is presented. Instead of using relieved sample clock period, whole combinational circuits are divided into two groups, Partition 1 (P1) and Partition 2 (P2) according to implementation considerations like delay

fault coverage and complexity. The P2 is duplicated in the next pipeline stage and comparison block (CP) for fault detection is added. In the next Sections, delay failure probability is discussed with emphasis on the possible operating clock frequency gain and followed by the description on the proposed delay error handling design.

C. Delay fault probability

In the previous Section, the possible gain in clock frequency of Duron 750 MHz CPU with delay fault tolerant system is illustrated for various delay failure rate. In this Section, we will explore the possible clock frequency gain in individual circuit using ISCAS benchmark circuits. First, the gate delays for various gates like nand and nor in typical operating conditions are calculated by SPICE simulations using 0.18 μm TSMC BSIM ver.3 model parameters. Then, delay simulations are conducted to obtain the path delay distributions of ISCAS Benchmark circuits using random test vectors. The results are the nominal gate delay distributions without operating condition variations. Figure 17 shows the delay distribution of ISCAS benchmark circuit C499 from the delay simulation. Totally 10^6 test vectors are used and all path delays from outputs are summed up.

To accommodate the effects of crosstalks and operating conditions variations on path delay, each path delay is modeled as a Gaussian distribution with mean and standard deviation. The nominal path delay is taken to mean value of the distribution and the standard deviation is set to be proportional to the mean and randomly chosen from 4% $\pm$ 2% of the mean value. In setting the range of the standard deviation, the effects of operating conditions variations and crosstalk on delay are considered. In [60], it is observed that microprocessor has 8% performance variation for 10% power supply voltage variation. Power supply voltages on main

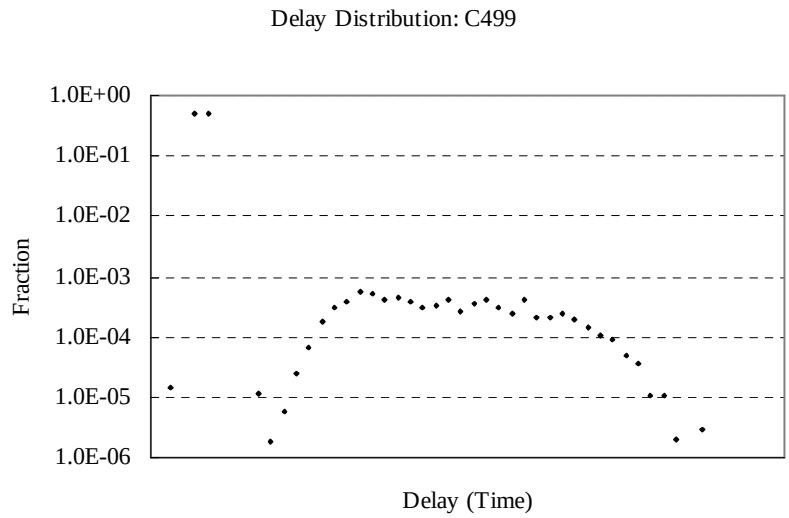


Fig. 17. Delay distribution of ISCAS C499 circuit

board are sampled and it has ± 5% variation for three standard deviations (3σ). We assume the similar value for the internal voltage variation. For temperature variation, it is shown that wide temperature variation exists across a die but no observation is made for a local position. Hence, it is assumed as 5^oC variation for 3σ. SPICE simulation for gate shows 0.8%/^{o}C delay variation. Considering equal delay contribution from interconnect and gate, 0.4%/^{o}C is taken for delay variation from temperature. In [37], the crosstalk can cause ± 10% delay variation according to test vector. So the nominal standard deviation for the individual path delay distribution is set to $\sqrt{\sigma^2_{Power} + \sigma^2_{Crosstalk} + \sigma^2_{Temperature}}$, which is 4%. Delay fault probability, P_f, is calculated from the modified delay distribution $D(t)$ as in equation (4.2). The correlations between each path delay are ignored because, in the clock frequency range of interests, P_f is very small and they will have a minimal difference.

$$P_f(t) = \int_{x=t}^{x=\infty} D(x)dx \tag{4.2}$$

Delay fault can propagate to the final pipeline output and result in delay failure

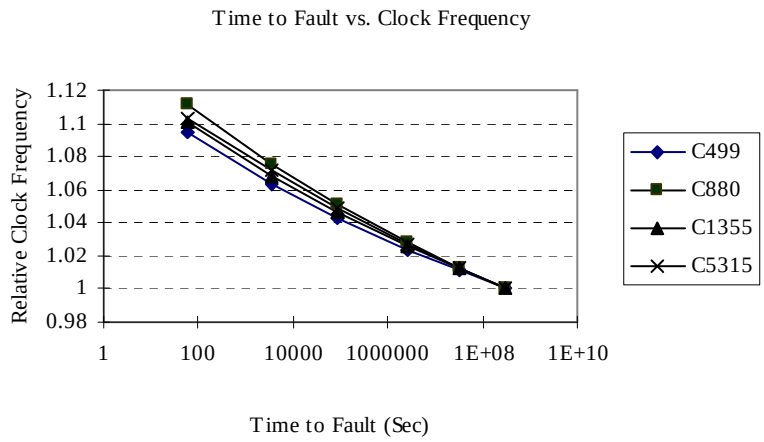


Fig. 18. Mean time to fault vs. clock frequency

or be masked. But in the proposed scheme, it is assumed that the delay fault, not delay failure, is detected and removed before it propagates to the next pipeline stage. In the calculation of the MTTF, the operation of each clock is treated as a Bernoulli trial. F is the frequency and assumed to 1 GHz for simplicity.

$$1 - \exp^{-\frac{t}{MTTF}} = 1 - (1 - P_f)^{F \cdot t} \tag{4.3}$$

$$MTTF = -\frac{1}{F \cdot \ln(1 - P_f)} \tag{4.4}$$

Figure 18 illustrates the possible clock gains for various mean-time-to-faults. We can expect 10% clock increase between 1 fault/min and 1 fault/10years. If speed margin for degradation is added for a new device, the gain will be larger as illustrated in Figure 19. In the figure, to accommodate degradation effect in delay, the mean of the delay distribution is assumed to increase by 5 % after degradation.

To achieve a performance gain from the fault tolerant scheme, the equation (4.5) should be satisfied where T_R is the fault recovery time, T_{pn} is the clock period of the proposed design, and T_{po} is the clock period of original design. In the proposed

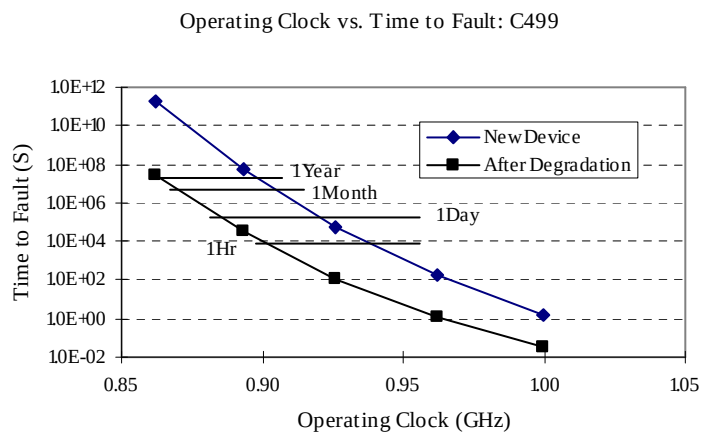


Fig. 19. Degradation effect for ISCAS C499

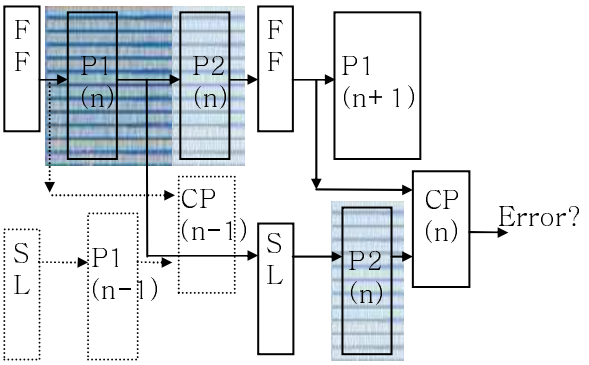
design, by using the reference signal from P2, T_R is minimized.

$$P_f \cdot T_R + T_{pn} < T_{po} \tag{4.5}$$

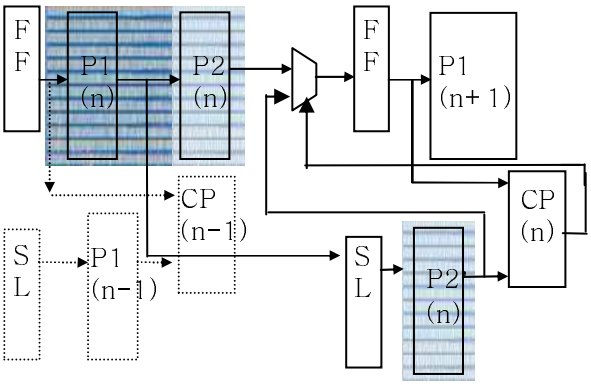
D. Delay fault detection and recovery

Fault tolerant systems can employ any redundancies in hardware, time, or information. In hardware redundancy, multiple replicas of a system are used to detect a fault by comparison and to remove it by voting. Using a single system, we can mimic the hardware redundancy by processing a work repeatedly. Coding is a widely used example of information redundancy. In the proposed scheme, we partially use the hardware redundancy and time redundancy. Figure 20 show the proposed architecture for pipelined structure with layouts in same scale from Silicon Ensemble.

The combinational circuit between latches is divided into two parts, P1 and P2. Therefore, every signal propagation path is divided accordingly. The Comparison circuit (CP) compares the outputs of original circuit processed in n^{th} pipeline stage (P1+P2) and ones from added circuit processed in $n+1^{th}$ pipeline stage (P2'). If



(a) Flush and clock retreat



(b) Signal replacement using MUX

Fig. 20. Proposed design for C880

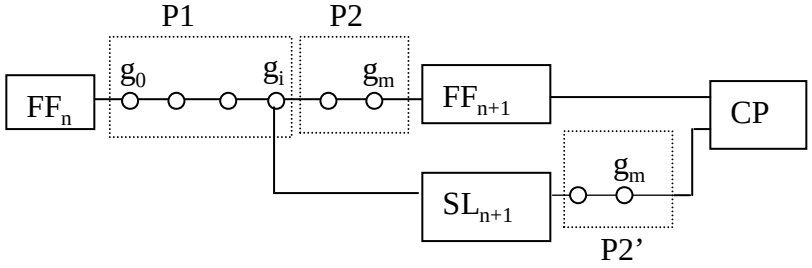


Fig. 21. Illustration of P1, P2 and P2'

they do not match, it is a fault and it sends control signal to refresh and decrease the operating clock frequency. After the process which causes the delay fault is carried out in the decreased clock frequency, the system recovers the original clock frequency. For modern CPUs, it takes several tens of microseconds to adjust clock frequency [61]. Figure 20.(a) illustrates the pipeline flush and clock retreat design. One possible modification is Figure 20.(b) where the reference signal inputs to CP from P2' are directed to the inputs of flip-flops (FFs) through MUX as employed in RAZOR. The output signal of CP controls the MUX. Usually the signal delay for MUX is two times of normal gate and it uniformly shifts delay distribution to worse delay time. Hence the design scheme in Figure 20.(b) may not be appropriate for the proposed design which aims at higher operating clock frequency through delay fault detection and removal.

Figure 21 illustrates a delay path consisted of m gates between flip-flops, FF_n and FF_{n+1}. To detect delay fault, it is essential to guarantee delay-fault-free output for the reference to be used in the comparison (CP). By storing the outputs of g_i instead of those of g_m, it adds the time slack corresponding to path delay across P2. Increasing the fault coverage can be done by increasing the portion of P2 to the original logic. If you clock the circuit aggressively, the division point should move

toward primary inputs, FF_n, which, in turn, increases the area for P2. Hence, it is important to balance the signal propagation delays through P1 and P2. Another factor in complexity comes the shadow latches, SL, which is determined from the number of interconnects between P1 and P2.

Figure 22 shows the complexity of added logic and delay detect margin of selected ISCAS benchmark circuits. Complexity of added logic (C_A) is defined in (4.6) as ratio of gates number in P2 and latches to that of original circuit and D-FFs. w is the area ratio of D flip-flop to normal gate. The fault detection coverage is defined as the ratio of the minimum signal propagation delay across P2 to the longest path delay in the original circuit as in (4.7) where d_{P1} is the delay from input to output gate in P1 and d_i is the delay from PI to PO gate in the original circuit.

$$C_A = \frac{N_{gate\_P2} + N_{gate\_CP} + N_{gate\_Latch}}{N_{gate} + w \cdot N_{FF}} \tag{4.6}$$

$$C_F = 1 - \frac{\max\{d_{P1}\}}{\max\{d_i\}} \tag{4.7}$$

In dividing the circuits into P1 and P2, the delay from the PI to the gate and the number of fanin and fanout are considered because the former determines C_F and the latter, C_A. The division flow is described in Figure 23. At first, delay simulation is conducted using test vectors to find the PI to gate delay. The number of test vectors is between 10^6 and 10^7 and they are randomly generated. First, PO with the longest path delay is chosen to P2. Other POs are chosen one by one in decreasing path delay order. Then, among the gates connected P2, the gate with the longest path is chosen and followed by the gates whose number of fanout is bigger than that of fanin to relieve added complexity. It repeats until the fault detection coverage is larger than 0.6. According to Figure 22, to achieve 10% delay fault detection coverage, the

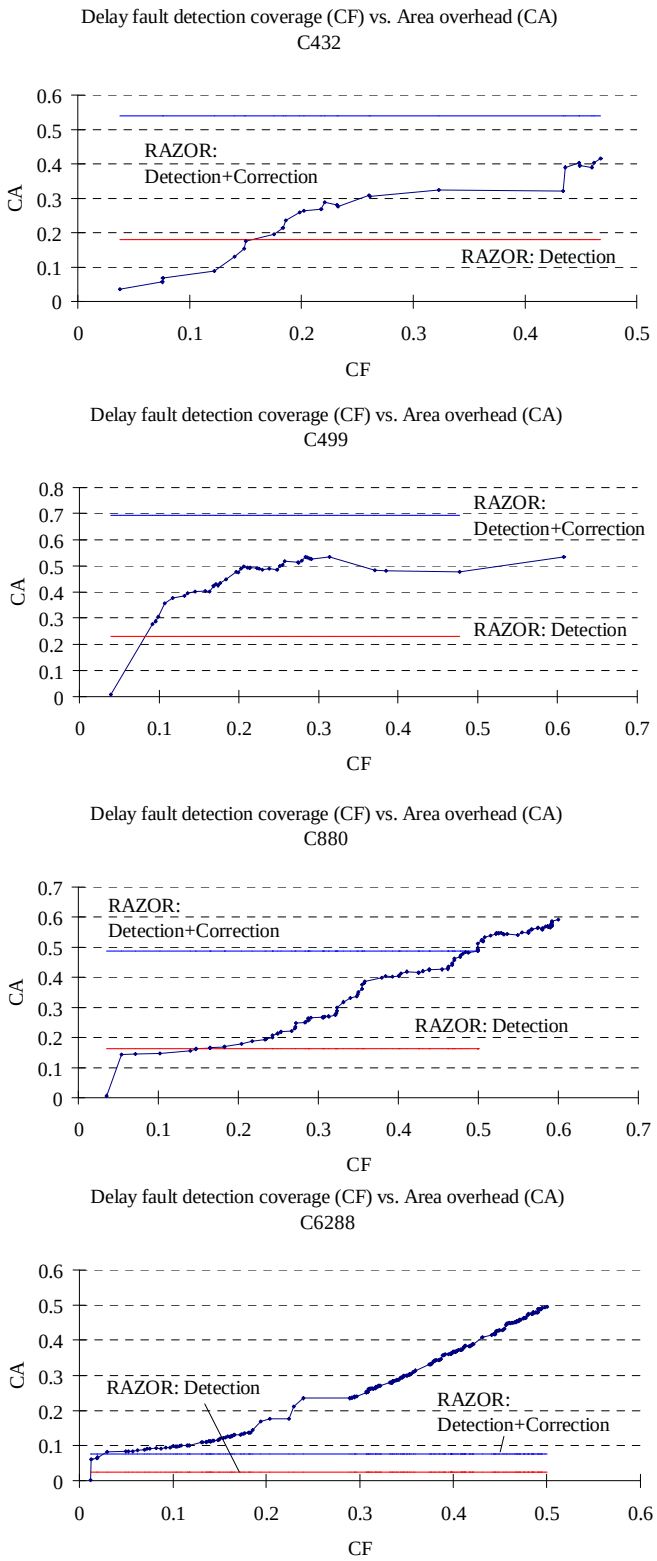


Fig. 22. Delay fault detection coverage vs. area overhead

```
for 1 to Ng //Ng: number of gates
  Max_delay_from_PI_to_gi=0;
for 1 to Nv //Nv: number of test vectors for delay simulation
  for 1 to Ng
    {if (delay_from_PI_to_g_i > Max_delay_from_PI_to_g_i)
        Max_delay_from_PI_to_g_i= delay_from_PI_to_the_g_i;
    }
Sort POs in decreasing order of delay
Add the first PO to P2;
  {calculate C_A and C_F;}
if any PO with ( N_Fanin > N_Fanout)
  {add the PO to P2; calculate C_A and C_F;}
for the remaining POs
  {add the PO to P2; calculate C_A and C_F;}

while(C_F<0.6)
  {  for all gates in P1
        for gates connected to any gate in P2
           Sort gates in decreasing order of delay
             {add the first gate to P2; calculate C_A and C_F;}
           if any gate with ( N_Fanin > N_Fanout)
              {add the gate to BL; calculate C_A and C_F;}
           add the remaining gate to P2
              {calculate C_A and C_F;}
 }
```

Fig. 23. Circuit partition flow

complexity increases by 25% for C499, 8% for C432, 12% for C880 and 8% for C6288. The dominant factor in complexity is the number of interconnects which requires latch between P1 and P2.

E. Conclusion

A design scheme to increase the operating clock frequency of circuit is presented. Delay faults which limit the aggressive clocking are detected by increased slack in redundancy circuit and removed by reprocessing at a lower clock frequency. The complexity of the redundancy circuit is minimized by sharing the delay path with the original circuit. Delay simulations on ISCAS benchmark circuits are conducted to estimate the operating clock frequency gains. Factors affecting path delay like operating conditions and crosstalks are taken into account in forms of Gaussian distributions. The estimation shows that up to 10% clock frequency increase is obtainable with moderate hardware penalty.

CHAPTER V

SINGLE EVENT TRANSIENT FAULT DETECTION USING PARTIAL REDUNDANCY

A. Introduction

In the previous chapter, a delay fault tolerant design is proposed. As an example of its application, a design implementation for detecting signal abnormalities caused by cosmic particles is presented in this chapter. When an ionic particle enters semiconductor substrate, it leaves a trace of generated charges and the collection of these charges in junction or at any node can result in enough potential changes to flip the state of flip-flop or memory. These phenomena are called single event upset (SEU) or soft error [62]. In combinational circuits, a node's potential level can be turbulent and this glitch, single event transient (SET), can propagate and result in wrong value latched at primary output. Traditionally, researches on radiation immunity of electronic devices have focused on space or military applications [63] due to their exposure to severe radiations in operating environments.

Recently, as the minimum feature size of circuit device shrinks down to deep-submicron, the critical amount of charge for SEU or SET decreases and the circuit is becoming more and more sensitive to radiation even at terrestrial area [64]. The low power supply voltage also makes the circuit signal's integrity more vulnerable to glitches.

The relationship between the energy of injected cosmic particle and the generated charge is studied and it is given in (5.1) [65]. Q is the amount of charge in pico Coulumb (pC), L is the Linear Energy Transfer of ion in $MeV/cm^2/mg$, and t is the depth of semiconductor diffusion region in μm. For example, the average energy

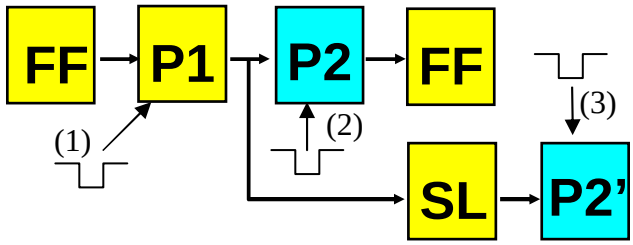


Fig. 24. Venues of fault injection

required to generate an electron-hole pair in silicon is 3.6 eV.

$$Q = 0.01036 \cdot L \cdot t \tag{5.1}$$

The resulting current which will cause a voltage spike or glitch at a node is modeled as a double exponential function as in (5.2) [66]. t_a and t_b are the time constants for charge collection and the ion track establishment, respectively.

$$I(t) = \frac{Q}{(t_a - t_b)}(e^{-t/t_a} - e^{-t/t_b}) \tag{5.2}$$

The focus of this chapter is on design implementations of the proposed delay fault tolerant circuit to detect a SET fault due to the current given in (5.2). In Section B, a general overview of the design approach is described. SPICE simulations to calculate the size of the glitch due to the current spike and delay simulations to propagate the glitch are introduced in Section C. For two design implementations, the probability of SET detection is calculated and discussed in Section D and the conclusion is drawn in Section E.

B. Approach

Figure 24 shows the overall partition of a circuit and the venues of SET fault injection. While the original signal propagates through P1 and P2, the reference

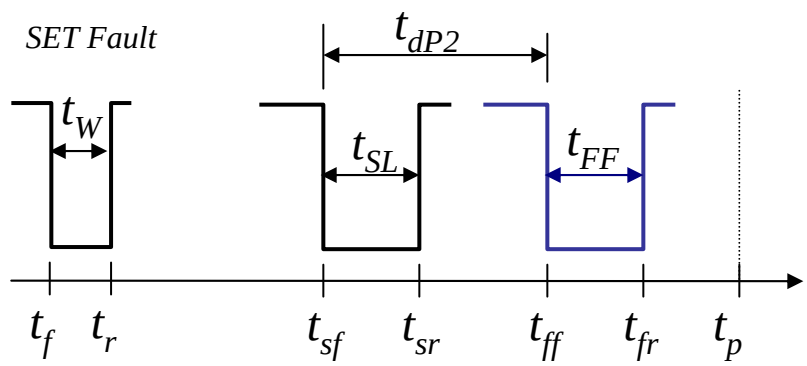


Fig. 25. Fault signal timing

Table V. Fault samplings at SL and FF and fault detection

Case	At SL	At FF	Result
1	$t_{sr} < t_p$	$t_{fr} < t_p$	No fault sampling
2		$t_{ff} < t_p < t_{fr}$	Fault sampling at FF -> Detection
3		$t_{ff} > t_p$	No fault sampling
4	$t_{sf} < t_p < t_{sr}$	$t_{ff} < t_p < t_{fr}$	Detection Failure
5		$t_{ff} > t_p$	Fault sampling at SL -> Detection
6	$t_{sr} > t_p$	$t_{ff} > t_p$	No fault sampling

signal branches at the output of P1 and goes through the shadow latch (SL) as well as the duplicated Partition 2 (P2'). By the comparison of the signals from the outputs of FF and P2', any delay faults originating from the circuit are guaranteed to be detectable if the timing slack added to the reference signal path is big enough to cover the delay fault size. In case of SET faults where the glitches caused by ionic particles behave as wrong signals, the fault detection depends on the timing of signals latched on FF and SL. Hence, any SET fault originating from FF, SL, P2 or P2' is detectable because at least one of the latched signals at FF or SL is correct. But for SET fault from P1, there exists a chance that the wrong signals are sampled at both of FF and SL depending on the fault's propagation timing. Hence, in evaluating the partitioned circuit's fault detection performance, the focus should be on the fault originating from P1.

Table IV lists all the possible cases for signals latched at FF and SL according to their timing. The terms used are illustrated in Figure 25. $t_f(t_r)$ is the fault injection time(recovery time) and t_w is the duration of the fault signal. The operating clock period is t_P and the arrival of fault and recovery signal at SL (FF) is $t_{sf}(t_{ff})$ and $t_{sr}(t_{fr})$, respectively. The gap between t_{sf} and t_{ff} is the fault propagation delay through P2, (t_{dP2}).

The overall approach is illustrated in Figure 26. First, delay simulations are conducted for the target circuit using randomly generated test vectors. In delay simulations, for each node, the maximum delay time from the primary inputs to the node is updated and used to calculate the delay fault detection coverage introduced in the previous chapter. The test vectors for the longest delays in the primary outputs are recorded to be used in fault simulations. Then, the circuit partitions are carried out to generate the designs for SET fault detection. The resulting designs correspond to different delay fault detection coverage and, accordingly, hardware overhead with cir-

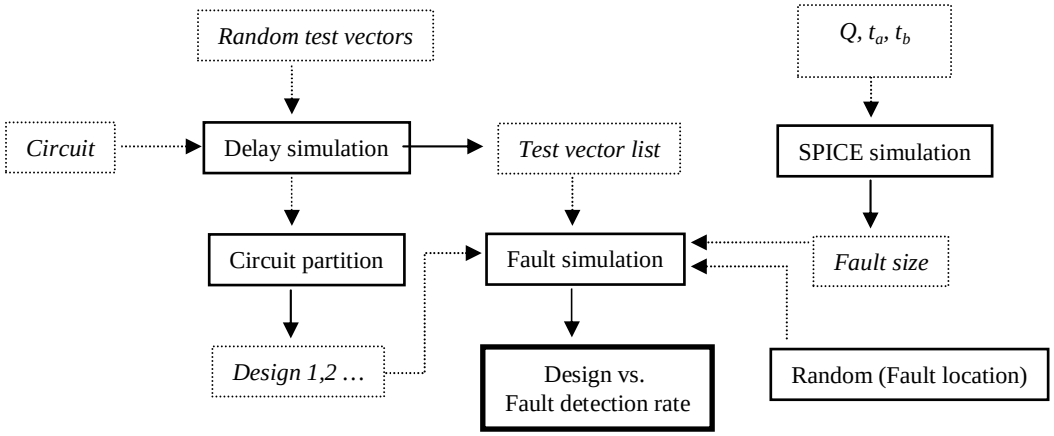


Fig. 26. Fault simulation flow

cuit partition for P2. These designs are evaluated using fault simulations to estimate SET fault detection rate. To quantify the strength of SET fault in the delay simulation, as t_W in Figure 25, SPICE simulations are conducted for various combinations of (Q, t_a, t_b). In the fault simulation, to increase the probability of invoking the smaller slack cases, test vectors for longer delays are used and fault injection nodes are randomly selected in P1. In the following sections, each step will be described in detail.

C. SET fault simulation

1. SPICE simulation

For the evaluations of each design's fault detection performance, fault simulations are conducted by calculating the fault propagation timing. To calculate the arrival time of SET fault and subsequent recovery signals at the inputs of FF and SL, the charge amount (Q) and the characteristic times of SEU (t_a, t_b) need to be represented in fault injection time and recovery time. These values are obtained through SPICE simulations with the setting shown in Figure 27. TSMC 0.18 μm model parameters

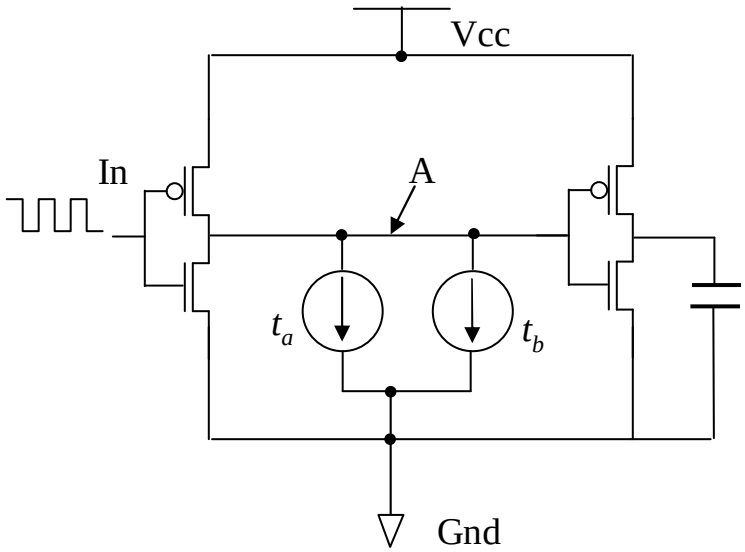


Fig. 27. SPICE simulation setting

are used in SPICE simulations. For various Q and t_b combinations in (5.1), the size of glitch is calculated with fixed value of t_a at 5 ps as studied in [65]. Because the logic simulation is used to detect the fault as described in the later Section, the faults or glitches that do not cross the assumed logic trip level (0.5Vcc) are disregarded. The current pulse is included as a double exponential current source [66]. The signal response or glitch on inverter resulting from the current pulse is calculated in the simulation because it is shown to be the most sensitive gate [65].

From the simulation results, the voltage pulse width at 0.5Vcc is calculated and used as a fault size. The pulse widths are shown in Table V.

2. Fault simulation

The propagation of transition edges resulting from the current pulse is calculated for the fault injection and the recovery in transient fashion. The delays in gates are only considered and the signal transition delay for various gate input combinations are

Table VI. Voltage pulse width(W_{pulse}) for various Qs and t_bs for t_a at 5 ps

Q	t_b=10 ps		t_b=30 ps		t_b=50 ps	
(fC)	I (mA)	W_{pulse} (ps)	I(mA)	W_{pulse} (ps)	I(mA)	W_{pulse} (ps)
8	1.6	16	0.32	N/A	0.18	N/A
9	1.8	30	0.36	N/A	0.20	N/A
10	2	43	0.4	N/A	0.22	N/A
15	3	90	0.6	59	0.33	N/A
20	4	94	0.8	113	0.44	87
25	5	97	1.0	126	0.56	154
30	6	99	1.2	132	0.67	179
35	7	99	1.4	137	0.78	185
40	8	100	1.6	138	0.89	189
45	9	102	1.8	140	1.00	195
50	10	103	2.0	142	1.11	198

calculated using SPICE simulation and applied in the simulation as in the previous chapter. The vectors with the longest path delay for each primary output are selected by running 10^8 randomly generated vectors and placed at the test vector list for fault simulation to the cover critical paths together with randomly generated test vectors. For each test vector, multiples of SET faults are injected as descried in the following Sections.

3. Fault injection

As it is clear that any fault injected to P2, P2', SL, or FF is guaranteed to be detected, in the simulations, the fault is injected only to the gate outputs in P1 with fault width as calculated in Table 1. The time of the fault injection is randomly chosen at any moment between the signal transition at the gate output and the data latch at the primary outputs.

4. Fault detection

To determine the output values, the fault signal transition times at the outputs are compared with the operating clock period. If the clock period comes between the fault transition time and the recovery transition time at the primary outputs, wrong values are latched. A fault is detected when any primary output's value and the value of the corresponding P2' gate output do not match. To count the cases when both outputs have wrong values, the values are compared to the value from the fault-free output.

5. Circuit partition

ISCAS Benchmark circuit C499 is studied for circuit partition. In partitioning of the circuit for the proposed SET fault detection design, two factors are considered:

the fault detection coverage and the hardware overhead. The performance of the circuit is determined by the fault detection coverage. As the fault's width increases, the probability that both of the primary output and the redundant circuit's output have wrong values also increases. To detect these wider faults, the time slack in the P2 circuit needs to be increased and, accordingly, the hardware overhead of the redundant circuit increases too. Hence, the gain in the fault detection coverage and the accompanying hardware overhead need to be optimized. The minimum hardware overhead design is when only the primary outputs are assigned to P2. If any primary output does not accompany the redundant output of P2', all faults propagating to the primary output will cause errors without being detected. We can increase the timing slack of the reference path by adding more gates to P2. In the next Section, the fault simulation results for two design implementations, where only the primary outputs are assigned to P2 (Design I), and several gates are added to Design I (Design II) are explained and discussed. The numbers of gates added to the primary outputs to form Design II are 354, 695, 696, and 700 of ISCAS C499. The hardware overhead are 20% for Design I and 30% for Design II and corresponding fault detection coverages are 8% and 10%, respectively.

D. Simulation results and discussion

Table VII shows the fault simulation results. The first column, W_{pulse}, is the width of the glitch at the fault injection. For two design implementations, N_{Miss} is the number of cases when the faulty signals are latched at both SL and FF. N_{Det} is the number of the cases when fault is detected. P_{Miss} is $N_{Miss}/(N_{Miss} + N_{Det})$. The results show small differences in P_{Miss} for the two circuit implementations. That is, the size of timing margin added to the reference path by skipping P2 is not critical

Table VII. SET fault simulation for two design implementations

W_{pulse}	Implementation I			Implementation II		
(ps)	N_{Miss}	N_{Det}	P_{Miss}	N_{Miss}	N_{Det}	P_{Miss}
16	5053	15992986	0.03%	4996	15953568	0.03%
30	5878	15950905	0.04%	5761	15927523	0.04%
45	6839	15913129	0.04%	6777	15950877	0.04%
60	8094	15918035	0.05%	8010	15949538	0.05%
90	8445	15981441	0.05%	8357	15952829	0.05%
100	8774	15981937	0.05%	8432	15950727	0.05%
110	8899	15977042	0.06%	8788	15963411	0.06%
130	9391	15984516	0.06%	9147	15947608	0.06%
140	9303	15980902	0.06%	9096	15956132	0.06%
150	9794	15968086	0.06%	9047	15956947	0.06%
180	10849	15970462	0.07%	10278	15964666	0.06%
200	11913	16006048	0.07%	11328	15961134	0.07%

for the detection rate. It may seem to be contradictory to the expectation that by increasing the timing gap between the fault arrival times at FF and SL (t_{dP2}), we can reduce the fault miss. One possible explanation is that the propagation of the signal across a circuit portion adds the delay and may shift the relative falling edge timing and rising edge timing. Hence, the fault size at the inputs of FF and SL can drastically increase compared to the injected fault size. This wider width of fault at the inputs of the latchesFFs makes the detection rate of the two different circuits almost same.

E. Conclusion

In this chapter, SET fault detection circuits are implemented for ISCAS Benchmark circuit C499 based on SABRE approaches in the previous chapter. The effect of SET on a circuit is quantified as the width of resulting glitch using SPICE simulation. The fault simulations are conducted for the propagation timing of the SET faults and the values latched at SL and FF are compared to calculate the fault detection rate. The simulation results show that with 20% hardware overhead of the original circuit, more than 99% of the SET faults can be detected. While the overhead associated with the proposed

CHAPTER VI

SUMMARY AND CONCLUSIONS

For the last several decades, as characterized by Moore's law, the integration density of IC has increased exponentially and the operating clock frequency has reached at multi-GHz area owing the continuing improvements and innovations of VLSI technology. However, to guarantee the timing of device is becoming increasingly difficult because of the reduced timing margin and the rapidly growing complexity.

Operating clock frequency is one of the key parameters representing IC's performance and determined by the longest signal propagation delay, setup/hold time, and timing margin. Hence, inaccurate timing evaluation in design stage may result in production of devices with too much timing margin (overdesign) or ones failing to satisfy the specification required by market. Similar cases can happen in speed binning: if it is too conservative, the gap between the operation performance and realizable performance will be big (underperformance) and in the other case, it may not guarantee the time-to-failure specification. But the accurate timing estimation is hindered by uncertainties in every stage of device manufacturing processes. First, in design stage, the complete design evaluation coverage is impossible because of the design complexity as the number of the critical paths is expected to increase exponentially as the feature size is reduced. Second, in fabrication process, the electrical parameter fluctuations resulting from the intra-die process variations is not ignorable anymore in addition to those from the traditional, die-to-die process variations. Finally, the timing of ICs are becoming more and more vulnerable to operating noises like temperature variations, power supply voltage fluctuations, and crosstalks.

To ensure that a device operating at its clock frequency is error-free with quantifiable assurance, effort at device-level engineering will not suffice for these circuits

exhibiting wide process variation and heightened sensitivities to operating condition stress. Logic-level redress of this issue is a necessity and we propose a design-level remedy for this timing-uncertainty problem.

In this dissertation, approaches for timing analysis of VLSI circuit based on experiments and design techniques for accommodating various faults are presented. The specific objectives addressed in this research are: (1) empirical estimation of delay failure rate of CPUs, and (2) delay fault tolerant design for operating clock frequency gain.

Starting with the individual path delay distribution, the delay failure rate distribution of circuits is assumed to be summation of individual delay distributions multiplied by their probabilities of excitation and propagation. Then, the effects of operation noises' on delay distribution are analyzed. Temperature variation, power supply voltage variation, and crosstalks are considered as operating noises and their effects on delay distribution are modeled to be Gaussian distribution. Because of the huge number of circuits' paths, it is impossible to find the distribution parameters for each Gaussian distribution and one assumption that the summation of Gaussian distribution can be approximated to be a normal distribution is applied and verified using Monte-Carlo simulations. In experiments, the time-to-failure measurements are carried out in stress conditions to invoke the failures, which has extremely small probability in nominal operating conditions because of the timing margin added in speed binning. From the measured TTF distributions, the MTTFs are extracted by curve fitting. Using delay failure probabilities from experiments of various temperatures and operating clock frequencies, the normal distribution parameters, mean and standard deviation, are calculated. The results show that the CPUs have enough MTTF resulting from delay fault for new product. The estimated operating noises' effect is applied to calculate the maximum operating clock frequency (FMAX) distribution

and is shown to be comparable to that from intra-die process variation.

The results from the delay failure rate estimation suggest that if the circuits have delay fault tolerant circuit, the performance gain in operating clock frequency can be considerable. Hence, a delay fault tolerant design, SABRE, is proposed.

In the proposed delay fault tolerant design, delay fault is detected by comparison of a signal propagating original circuit and a reference signal of bigger timing slack. The essential part of the design is how to ensure timing slack for error-free reference signal with negligible or moderate area overhead. In the proposed design, the reference signal circuit path shares some part of the path with the original circuits to reduce area complexity. All circuits are divided into two groups: Partition 1 and Partition 2. Partition 2 is replicated and placed in the next pipeline stage to give timing slack, which is the delay across it, to reference signal path. To optimize the delay fault detection coverage and area overhead, a generic algorithm is used. ISCAS benchmark circuits are exemplified to show that proposed design can be implemented with moderate area complexity.

The proposed design scheme is applied to remedy SET fault which is a voltage spike or glitch caused by ionic particles. For ISCAS benchmark circuit C499, the circuit is partitioned to give two different design implementations and the fault simulations are conducted to evaluate their fault detection ratios. The results show that more than 99% of SET faults are detected for 20% hardware overhead of the original circuit.

The future work includes (1) device degradation's effect on delay failure rate distribution and (2) proposed design's application to practical circuits like ALU.

REFERENCES

[1] G. E. Moore,"Cramming more components onto integrated circuits," *Electronics Magazine*, vol. 38, pp. 114–117, Apr. 1965.

[2] P. K. Bondyopadhyay,"Moore's law governs the silicon revolution," in *Proc. of the IEEE*, vo. 86, pp. 78–81, Jan. 1998.

[3] K. Kikuta, T. Takewaki, Y. Kakuhara, K. Fujii, and Y. Hayashi,"Comparative study of W-plug, Al-plug and Al-dual damascene for 0.18 μm ULSI multilevel interconnect technologies," in *Proc. of the IEEE International Interconnect Technology Conference*, San Francisco, CA, Jun. 1998.

[4] Y. Y. Cheng, S. M. Jang, C. H. Yu, S. C. Sun, and M. S. Liang, "A high performance and reliable low-k inter-metal dielectric using hydrogen silsesquioxane (HSQ)", in *Proc. of the IEEE International Interconnect Technology Conference*, San Francisco, CA. May 1999.

[5] K. A. Bowman, S. G. Duvall, and J. D. Meindl, "Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration," *IEEE Journal of Solid-State Circuits*, vol. 37, pp. 183–190, Feb. 2002.

[6] M. Orshansky, "A general probabilistic framework for worst case timing analysis," in *Proc. of 39th Design Automation Conference*, New Orleans, LA, Jun. 2002.

[7] S. G. Duvall, "Statistical circuit modeling and optimization," in *Proc. of 5th Intl. Workshop on Statistical Metrology*, Honolulu, HI, Jun. 2000.

[8] S. Chang and G. Choi, "System level delay error analysis," in *Proc. of 5th IEEE Latin America Test Workshop*,Cartagena, Colombia, Mar. 2004.

[9] K. T. Tang and E. G. Friedman, "Delay uncertainty due to on-chip simultaneous switching noise in high performance CMOS integrated circuits," in *Proc. of IEEE Workshop on Signal Processing Systems*, Lafayette, LA, Oct. 2000.

[10] Z. Yang and S. Mourad, "Deep submicron on-chip crosstalk [and ANN prediction]," in *Proc. of the 16th IEEE Instrumentation and Measurement Technology Conference*, Venice, Italy, May 1999.

[11] Y. K. Malaiya and R. Narayanaswamy, "Modeling and testing for timing faults in synchronous sequential circuits," *IEEE Design and Test of Computers Magazine*, vol. 1, pp. 62–74. 1984.

[12] G. L. Smith, "Model for delay faults upon paths," in *Proc. of 16th IEEE International Test Conference*, Philadelphia, PA, Nov. 1985.

[13] J. L. Carter, V. S. Iyengar, and B. K. Rosen, "Efficient test coverage determination for delay faults," in *Proc. of 18th IEEE International Test Conference*, Washington, DC, Sept. 1987.

[14] A. K. Pramanick and S. M. Reddy, "On the computations of the ranges of detected delay fault sizes," in *Proc. of 7th IEEE International Conference on CAD*, Santa Clara, CA, Nov. 1989.

[15] A. K. Pramanick and S. M. Reddy, "On the fault coverage of gate delay detecting tests," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 16, pp. 78–94, Jan. 1997.

[16] A. K. Majhi, J. Jacob, L. M. Patnaik, and V. D. Agrawal, "On test coverage of path delay faults," in *Proc. of 9th IEEE International Conference on VLSI Design*, Bangalore, India, Jan. 1996.

[17] K. Heragu, J. H. Patel, and V. D. Agrawal, "Segment delay faults: New fault model," in *Proc. of 14th IEEE VLSI Test Symposium*, Princeton, NJ, Apr. 1996.

[18] R. C. Elliott, R. K. Nurani, D. Gudmundsson, M. Preil, R. Nasongkhla, and J. G. Shanthikumar, "Critical dimension sample planning for sub-0.25 micron processes," in *Proc. of Advanced Semiconductor Manufacturing Conference and Workshop*, Boston, MA, Sep. 1999.

[19] E. Chang, B. Stine, T. Maung, R. Divecha, D. Boning, J. Chung, K. Chang, G. Ray, D. Bradbury, O. S. Nakagawa, S. Oh, and D. Bartelink, "Using a statistical metrology framework to identify systematic and random sources of die- and wafer-level ILD thickness variation in CMP processes," in *Tech. Digest of International Electron Devices Meeting*, Washington, DC, Dec. 1995.

[20] M. Eisele, J. Berthold, D. Schmitt-Landsieldel, and R. Mahnkopt, "The impact of intra-die parameter variations on path delays and on the design for yield of low voltage digital circuits," in *Proc. International Symposium on Low Power Electronics and Design*, Monterey, CA, Aug. 1996.

[21] H. Mahmoodi, S. Mukhopadhyay, and K. Roy, "Estimation of delay variations due to random-dopant fluctuations in nanoscale CMOS circuits," *IEEE Journal of Solid-State Circuits*, vol. 40, pp. 1787–1796, Sep. 2005.

[22] H. Abe, F. Kiyosumi, K. Yoshioka, and M. Ino, "Analysis of defects in thin SiO2thermally grown on Si substrate," in *Tech. Digest of International Electron Devices Meeting*, Washington, DC, Dec. 1985.

[23] J.-D. Lee, J.-H. Choi, D. Park, and K. Kim, "Data retention characteristics of sub-100 nm NAND flash memory cells,"*Electron Device Letters*,vol. 24, pp. 748 – 750, Dec. 2003.

[24] C. G. Shirley and S. C. Maston, "Electrical measurements of moisture penetration through passivation," in *28th Annual Proc. of Reliability Physics Symposium*, New Orleans , LA, Apr. 1990.

[25] B. S. Doyle and K. R. Mistry, "The characterization of hot carrier damage in p-channel transistors," *IEEE Trans. on Electron Devices*, vol. 40, pp. 152–156, Jan. 1993.

[26] B. S. Doyle and K. R. Mistry, "Anomalous hot-carrier behavior for LDD p-channel transistors," *IEEE Electron Device Letters*, vol. 14, pp. 536–538, Nov. 1993.

[27] F. J. Guarin, G. La Rosa, Z. J. Yang, and S. E. Rauch III, "A practical approach for the accurate lifetime estimation of device degradation in deep sub-micron CMOS technologies," in *Proc. of 4th IEEE International Caracas Conference on Devices, Circuits and Systems*, Aruba, Dutch Caribbean, Apr. 2002.

[28] Z. Cui, J. J. Liou, Y. Yue, and H. Wong, "A new approach to characterize and predict lifetime of deep-submicron nMOS devices," in *Proc. of 7th International Conference on Solid-State and Integrated Circuits Technology*, Beijing, China, Oct. 2004.

[29] Y. Takemoto and I. Arizono, "Design of accelerated reliability tests based on simple-step-stress model," in *Proc. of Annual Reliability and Maintainability Symposium*, Tampa, FL, Jan. 2003.

[30] P. G. Y. Tsui, L. Howington, P. M. Lee, T. Tiwald, B. Mowry, F. K. Baker, J. D. Hayden, B. B. Feaster, and B. Garbs, "An integrated system for circuit level hot-carrier evaluation," in *Proc. of IEEE Custom Integrated Circuits Conference*, Boston, MA, May 1990.

[31] M. Karam, W. Fikry, and H. Ragai,"Implementation of hot-carrier reliability simulation in Eldo," Mentor Graphics Deep Submicron Technical Publication, Sep. 2000.

[32] K. Baker and J. Van Beers, "Shmoo plotting: the black art of IC testing," *IEEE Design & Test of Computers Magazine*, vol. 14, pp. 90–97, Jul. 1997.

[33] K. T. Lee and J. A. Abraham, "Critical path identification and delay tests of dynamic circuits," in *Proc. of International Test Conference*, Atlantic City, NJ, Sep. 1999.

[34] M. Orshansky, and A. Bandyopadhyay, "Fast statistical timing analysis handling arbitrary delay correlations," in *Proc. of 41th Design Automation Conference*, San Diego, CA, Jun. 2004.

[35] S. Louhichi, "Rates of convergence in the CLT for some weakly dependent random variables," *Theory of Probability and Its Applications*, vol. 46, pp. 297–315, 2002.

[36] S. Khatri, R. Brayton, and A. Sangiovanni-Vincentelli, *Cross-Talk Noise Immune VLSI Design Using Regular Layout Fabrics*, New York, NY: Springer, 2001.

[37] A. Krstic, J.-J. Liou, Y.-M. Jiang, and K.-T. Cheng, "Delay testing considering crosstalk-induced effects," in *Proc. of International Test Conference*, Baltimore, MD, Oct. 2001.

[38] K. Takeuchi, K. Yanagisawa, T. Sato, K. Sakamoto, and S. Hojo, "Probabilistic Crosstalk Delay Estimation for ASICs," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 23, pp. 1377–1383, Sep. 2004.

[39] E. W. Weisstein. "Central Limit Theorem." From MathWorld-A Wolfram Web Resource. http://mathworld.wolfram.com/CentralLimitTheorem.html, May 2005.

[40] L. Yang, and J.S. Yuan, "Modelling and analysis of ground bounce due to internal gate switching," in *Proc. of IEE Circuits, Devices and Systems*, vol. 151, pp. 300–306, Aug. 2004.

[41] G. C. Gomez, A. Cadena, and V. H. Champac, "Switching noise due to internal gates: delay implications and modeling," in *Proc. of 3rd IEEE International Caracas Conference on Devices, Circuits and Systems*, Cancu'n, Mexico, Mar. 2000.

[42] M. Saint-Laurent and M. Swaminathan, "Impact of power-supply noise on timing in high-frequency microprocessors," *IEEE Trans. on Advanced Packaging*, vol. 27, pp. 135–144, Feb. 2004.

[43] G. R. Grimmett and D. R. Stirzaker, *Probability and Random Processes, 2nd edition*, Oxford, UK: Clarendon Press, 1992.

[44] H. Kim and K. G. Shin, "Evaluation of fault tolerance latency from real-time application's perspectives," *IEEE Trans. on Computers*, vol. 49, pp. 55–64, Jan. 2000.

[45] AMD Duron processor model 3 data sheet, Jul. 2003, http://www.amd.com/us-en/assets/content_type/white_papers_and_tech_docs/23802.pdf.

[46] J. G. Xi and W.-M. Dai, "Buffer insertion and sizing under process variations for low power clock distribution," in *Proc. of 32th Design Automation Conference*, San Francisco, CA, Jun. 1995.

[47] A. Agarwal, B. Blaauw, and V. Zolotov, "Statistical timing analysis for intra-die process variations with spatial correlations," in *Proc. of IEEE International Conference on Computer Aided Design*, San Jose, CA, Nov. 2003.

[48] P. S. Zuchowski, P. A. Habitz, J. D. Hayes, and J. H. Oppold, "Process and environmental variation impacts on ASIC timing," in *Proc. of IEEE International Conference on Computer Aided Design*, San Jose, CA, Nov. 2004.

[49] A. Agarwal, D. Blaauw, and V. Zolotov, "Statistical clock skew analysis considering intra-die process variations," in *Proc. of IEEE International Conference on Computer Aided Design*, San Jose, CA, Nov. 2003.

[50] F. Caignet, S. Delmas-Bendhia, and E. Sicard, "The challenge of signal integrity in deep-submicrometer CMOS technology," in *Proc. of the IEEE*, vol. 89, pp. 556–573, Apr. 2001.

[51] Y.-K. Cheng, P. Raha, C.-C. Teng, E. Rosenbaum, and S.-M. Kang, "ILLIADS-T: an electrothermal timing simulator for temperature-sensitive reliability diagnosis of CMOS VLSI chips," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 17, pp. 668–681, Aug. 1998.

[52] S. Pant, D. Blaauw, V. Zolotov, S. Sundareswaran, and R. Panda, "Vectorless analysis of supply noise induced delay variation," in *Proc. of International Conference on Computer Aided Design*, San Jose, CA, Nov. 2003.

[53] R. Kumar, "Interconnect and noise immunity design for the Pentiumr 4 Processor," *Intel Technology Journal Q1*, pp. 1–12, 2001.

[54] J. H. Patel, "Stuck-at fault: a fault model for the next millennium," in *Proc. of International Test Conference*,Washington, DC, Oct. 1998.

[55] S. M. Reddy, I. Pomeranz, and S. Kajihara, "Compact test sets for high defect coverage," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 16, pp. 923–930, Aug. 1997.

[56] Product brief on Intel Pentium 4 Processor 600 sequence, May 2006, http://www.intel.com/products/processor/pentium4/prodbrief.pdf.

[57] C. W. Slayman, "Cache and memory error detection, correction, and reduction techniques for terrestrial servers and workstations," *IEEE Trans. on Device and Materials Reliability*, vol. 5, pp. 397–404, Sep. 2005.

[58] D. Ernst, N. S. Kim, S. Das, S. Pant, R. Rao, T. Pham, C. Ziesler, D. Blaauw, T. Austin, K. Flautner, and T. Mudge,"Razor: a low-power pipeline based on circuit-level timing speculation," in *Proc. of 36th IEEE/ACM International Symposium on Microarchitecture*, San Diego, CA, Nov. 2003.

[59] S. Das, D. Roberts, S. Lee, S. Pant, D. Blaauw, T. Austin, K. Flautner, and T. Mudge, "A self-tuning DVS processor using delay-error detection and correction," *IEEE Journal of Solid-State Circuits*, vol. 41, pp. 792–804, Apr. 2006.

[60] K. Bernstein, K. M. Carrig, C. M. Durham, P. R. Hansen, D. Hogenmiller, E. J. Nowak, and N. J. Rohrer, *High Speed CMOS Design Styles*, New York, NY: Springer, 1998.

[61] Enhanced Intel SpeedStep technology and demand-based switching on Linux, Jun. 2006, http://www.intel.com/cd/ids/developer/asmo-na/eng/195910.htm.

[62] C. L. Axness, J. R. Schwank, P. S. Winokur, J. S. Browning, R. Koga, and D. M. Fleetwood, "Single event upset in irradiated 16 K CMOS SRAMs," *IEEE Trans. on Nuclear Science*, vol. 35, pp. 1602–1607, Dec. 1988.

[63] D. Binder, C. Smith, and A. Holman, "Satellite anomalities from galactic cosmic rays," *IEEE Trans. on Nuclear Science*, vol. NS–22, pp. 2675–2680, Dec. 1975.

[64] Y. Z. Xu, H. Puchner, A. Chatila, O. Pohland, B. Bruggeman, B. Jin, D. Radaelli, and S. Daniel, "Process impact on SRAM alpha-particle SEU performance," in *Proc. of IEEE 42nd Annual Reliability Physics Symposium*, Phoenix, AZ, Apr. 2004.

[65] R. Garg, N. Jayakumar, S.P. Khatri, and G. Choi, "A design approach for radiation-hard digital electronics," in *Proc. of 43rd IEEE/ACM Design Automation Conference*, Anaheim, CA, Jul. 2006.

[66] X. Yuan, "Transient Fault Modeling and Fault Injection Simulation," Master's thesis, Texas A&M University, College Station, TX, Dec. 1996.

VITA

Sanghoan Chang received a B.S degree in Physics and an M.S degree in Physics from Seoul National University, Seoul, Republic of Korea. He worked in the Flash memory division of the R&D center in Hyundai Electronics (currently Hynix Electronics). His work experience includes device engineering, process integration, model parameter verification, and reliability characterization from 0.8μm to 0.18μm CMOS technology. Mr. Chang pursued his Ph.D. degree in Electrical and Computer Engineering at Texas A&M University and graduated in May 2007. He won an Excellent Engineer Award from Hyundai Electronics and has two U.S. patents. He can be reached at the following address: 23129 Tranquil Springs Ln, Katy TX 77494. His email address is shchangtx@gmail.com.

The typist for this dissertation was Sanghoan Chang.