Ph.D. Dissertation

Performance Monitoring System for Low Power Design

저전력 설계를 위한 성능 모니터링 시스템

by

Jongho Kim

August 2018

Department of Electrical and Computer Engineering
College of Engineering
Seoul National University

Performance Monitoring System for Low Power Design

저전력 설계를 위한 성능 모니터링 시스템

지도교수 최기영

이 논문을 공학박사 학위논문으로 제출함

2018 년 6 월

서울대학교 대학원
전기·정보공학부

김종호

김종호의 공학박사 학위논문을 인준함

2018 년 6 월

위 원 장	김태환	(인)
부위원장	최기영	(인)
위 원	이혁재	(인)
위 원	유승주	(인)
위 원	도경태	(인)

Abstract

As the semiconductor process technology continuously scales down, circuit delay variations due to manufacturing and environmental variations become more and more serious. These delay variations are hardly predictable and thus require an additional design margin, which impedes the chance to reduce area and power consumption of a chip. One of the best solutions to alleviate this problem is to measure circuit delays at run-time and control the supply voltage accordingly through a closed-loop dynamic voltage and frequency scaling scheme. The key issue of this scheme is the delay mismatch between the monitoring circuit and the target block. A large delay mismatch might lose the advantage of closed-loop dynamic voltage and frequency scaling. It becomes much worse as a circuit block operates in wider voltage range, from near-threshold voltage to super-overdrive voltage. We propose novel delay monitoring systems with multiple generic monitors for wide voltage range operation, which provide a better delay correlation between the monitoring circuit and the target block compared to conventional monitoring approaches. The proposed approaches reduce the maximum error by up to 91% for a popular processor core in a 14nm FinFET process technology, thereby bring a decrease of design margin, lower-power, and/or lower-cost design.

The second part of this dissertation deals with the method of aging compensation. Conventional approach against reliability degradation due to aging is to use a design margin. However, it is too pessimistic and requires an accurate estimation of the aging effects at design-time, which is challenging. Therefore, for low power design, it is essentially required to measure the chip slowdown by aging at run-time and compensate it dynamically. We propose a novel design methodology to turn the chip slowdown by aging into approximation without the reliability design margin or increasing the supply voltage. It guarantees the always-best quality with approximation as the delay increases by aging. It is based on the monitoring of critical path delay at run-time. The goal is to remove the reliability design margin but apply an appropriate precision reduction at run-time by curtailing the critical path. We evaluate the proposed approach on an RTL component as well as a system at the microarchitecture level. The experimental results on the component show a significant improvement in terms of mean squared error. And the experimental results on the system show that the proposed approach removes the aging-induced timing violation errors without large quality degradation. It reduces the dynamic and static power consumptions by 19.8% and 10.2%, respectively, with 0.4% area overhead.

Keywords: Performance monitor, adaptive control, design margin, low power, chip reliability, approximate computing

Student Number: 2014-30306

Contents

List of Figures

List of Tables

Part I

Delay Monitoring System with Multiple Generic Monitors for Wide Voltage Range Operation

Chapter 1

Introduction

Semiconductor products have been perpetually shrinking over the past decades to allow performance enhancements at a lower fabrication cost per transistor. However, this process scaling has also brought serious circuit delay variations [1]-[4] mainly due to manufacturing and environmental variations including inter-/intra-die variability, temperature shift, supply voltage droop noise and circuit aging. Typically, circuit delay variations have been covered by design margins to ensure 'no-error' operations under the variations. This is the most pessimistic approach considering all worst-cases and thus incurs additional costs that would be unnecessary in better-than-worst-cases. Moreover, the problem becomes more serious as the variation becomes larger and larger. Even, it is very difficult to determine the optimal design margin considering all the operating conditions at manufacturing test. Especially,

this problem might be too serious in CPU/GPU designs because such a block requires wide range of operating voltages, from near-threshold voltage to super-overdrive voltage, which generates much worse delay variations than that of previous chip operation [5]-[9].

Most commercial CPUs/GPUs adopt an open-loop dynamic voltage and frequency scaling (DVFS) scheme using look-up tables (LUTs) for low power operation. In general, it requires a large design margin added to the original DVFS mapping table. The design margin should be large enough to cover the worst case circuit delay variations due to supply voltage droop, temperature variation, and/or circuit aging. Thus, the level of power reduction achieved by the approach can be much lower than the expected one. Recently, a closed-loop DVFS scheme has been used to monitor a circuit delay dynamically at run-time. This scheme can prevent from increasing the design margin too much since it only needs to compensate for current circuit delay variations, which are typically much smaller than the worst case. In a closed-loop DVFS scheme, the key point is how to implement a monitoring circuit to estimate a circuit delay accurately. That is, the delay mismatch between the target block (such as a CPU or a GPU) and the monitoring circuit should be minimized. It is directly connected to the effectiveness of the closed-loop DVFS scheme. There have been various circuits proposed to implement a delay monitor. Generally, they can be classified into two groups according to the dependency on the block or design that the monitor is targeting: generic monitor and design-dependent monitor.

A generic monitor is mainly implemented by a simple inverter-based ring oscillator (RO). It does not have any dependencies on the target block. It is very practical and suitable for a short-time-to-market design because it can be easily implemented and reused for any chip design platforms. It might be difficult, however, to use it for a design implemented by a mix of device types (e.g., multi-threshold voltage design) possibly causing a big gap between the measured delay and the actual delay of the target block. Also, it still requires a large design margin because it has a relatively large delay mismatch with the actual critical path of the target block under various operating conditions. On the other hand, a design-dependent monitor is designed to be highly correlated with the target block in terms of delay. However, it typically requires a high area overhead and design complexity. And it might be difficult to use it for commercial chip designs because it has longer design turnaround time than a generic monitor and it cannot be reused unless the same target block is reused. The two types of monitoring circuits are presented in Chapter 2.3 in more detail. Again, these two types of monitoring circuits have limitations in their commercial use. A generic monitor has a lower accuracy and thus it is not suitable for wide voltage range operation, whereas a design-dependent monitor has low reusability/productivity and high design complexity and accordingly it is not good for short-time-to-market designs (we target application processor design but the basic concept applies to other kinds of system design). In this dissertation, we propose a novel monitoring circuit based on a generic monitor, and schemes that give both design simplification and an accurate circuit delay estimation.

The first part of this dissertation is organized as follows. Chapter 2 gives an introduction to open-loop DVFS and closed-loop DVFS based on monitoring circuits as well as related work for monitoring circuits. Chapter 3 presents a proposed monitoring circuit and monitoring schemes for wide voltage range operation and Chapter 4 describes a design methodology for delay monitoring system design. Chapter 5 provides the experimental results as well as the analyses of them. Finally, Chapter 6 summarizes our proposal with concluding remarks.

Chapter 2

Background and Related Work

DVFS is widely used for reducing power consumption during off-peak processing time and for preventing from overheating problems. The goal of this technique is to run a chip at the lowest possible voltage while achieving a desired performance or operating frequency. According to the use of feedback information from monitoring circuits, the DVFS approaches can be classified into two different schemes: open-loop DVFS and closed-loop DVFS. Chapter 2.1, 2.2, and 2.3 briefly introduce the two DVFS schemes and the supporting monitoring circuits.

2.1 Open-loop DVFS Scheme

Open-loop DVFS is the most commonly used DVFS scheme. The operating voltage for each desired operating frequency is determined at the manufacturing test step while considering operating conditions. Then the frequency-to-voltage mapping information is generally stored into a LUT in memory and used to scale the supply voltage up and down as requested by the applications. Typically, a large design margin is assigned to the operating voltage for each target frequency stored in the LUT for safety reason. Thus, there is a limitation in reducing power consumption since the operating voltage is pre-determined and cannot be adjusted dynamically depending on the run-time conditions.

2.2 Closed-loop DVFS Scheme

Closed-loop DVFS has been recently emerged to get more aggressive power reductions in low power competitive markets. Design margins in a supply voltage should be reduced as much as possible. Since the actual operating speed changes with various process/voltage/temperature (PVT) conditions, there is a need to calculate and assign an optimal voltage dynamically at run-time. For example, circuit aging, supply voltage droop noise and temperature shift always occur in the real environment and they affect the actual chip operating speed. This problem can be resolved by a feedback loop including a monitoring circuit which provides

information on how fast or slow a chip is actually running. As shown in Figure 2.1, a feedback loop is facilitated by a monitoring circuit that enables closed-loop DVFS, where the operating voltage is adaptively scaled to an optimal point. Compared to open-loop DVFS, it does not require a large design margin because it tracks an operating speed at run-time. However, it is impossible to remove a whole design margin because there can be a delay mismatch between the target block and the monitoring circuit. Therefore, the advantage of closed-loop DVFS absolutely depends on the delay correlation between the target block and the monitoring circuit, and thus it is the most important factor to decrease the delay mismatch between the two circuits in such a system.

2.3 Related Work on Monitoring Circuits

The accuracy of monitoring circuit is very important to maximize the effectiveness of closed-loop DVFS scheme. There have been many researches on monitoring delays and various circuits have been proposed. However, they can be simply classified into two categories: generic monitor and design-dependent monitor. A generic monitor is typically designed as a simple inverter-based RO. Process-specific ROs (PSROs) have been proposed to measure process parameters or variations of a chip [10][11]. Phase-locked loop (PLL) is used for an alternative monitoring circuit [12]. This monitor is very simple and easy to design. Also, it does not take a large area and can be easily reused for any other chip designs. In that sense, ROs are a

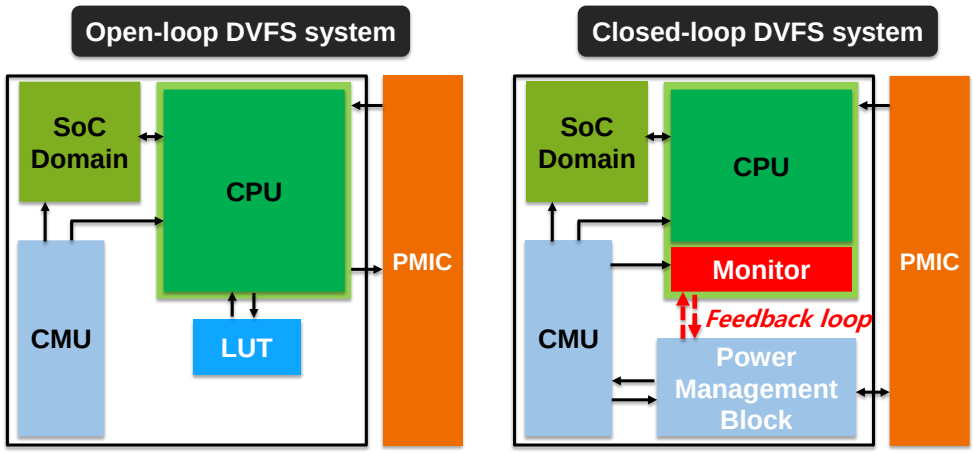


Figure 2.1 Open-loop and closed-loop DVFS system.

good example of design-for-manufacturability. However, it is in general less accurate than a design-dependent monitor and thus incurs a large design margin due to large delay mismatches.

A design-dependent monitor is tuned so that its delay characteristics are better correlated with those of a target block. Thus it is more accurate than a generic monitor, but many calibrations and parameter storage resources are required. Such a design-dependent delay monitor can be implemented based on a design-specific delay model [13]. A design-dependent RO can be synthesized according to the target design and process information [14]. It is relatively simple and has lower area overhead than other kinds of design-dependent monitors, but additional design turnaround time and characterization of the target block are required. Also, it cannot be reused for other chip designs. Critical path replica [15] is to make a copy of critical path and use it for circuit performance estimation. In-situ monitors [16][17] directly

track delays of critical paths for error detection or delay slack monitoring. Although they show good estimation accuracies, they cost a large area overhead and long design turnaround time. Various reconfigurable monitors have also been presented. For example, a critical path monitor is configured by various serial/parallel paths [18][19] and it is calibrated by different coarse/fine delay elements [20]. Or a tunable replica circuit is calibrated to mimic critical paths by different delay paths [21][22]. They can be flexibly tuned according to the circuit delay of a target block. Hence, it provides more accurate circuit delay estimations, but still has complex calibrations and large area overheads. A delay sensor is integrated directly into the critical paths to measure process/environmental variations [23]. It provides the most accurate circuit delay information because it directly tracks the delay of real critical paths, but it incurs a higher design complexity compared to any other monitors.

In Chapter 3, we present our proposed monitoring circuit and schemes, which are based on a generic monitor, but has an accuracy comparable to that of design-dependent monitors.

Chapter 3

Proposed Circuit and Scheme

3.1 Conventional Approach with a Generic Monitor

Generic monitors are widely used for commercial chip designs requiring short-time-to-market due to their simplicity and reusability. It is typically implemented by a single device type, whereas a target block such as a CPU or a GPU tends to be implemented by a mix of multiple device types supported by a multi-threshold voltage library for achieving both low power and high performance at the same time. Examples of such device types include low-threshold voltage (LVT) transistors, regular-threshold voltage (RVT) transistors, and high-threshold voltage (HVT) transistors. Such a design with multiple device types exhibits delay responses quite different from a monitor made of a single device type under various PVT (especially

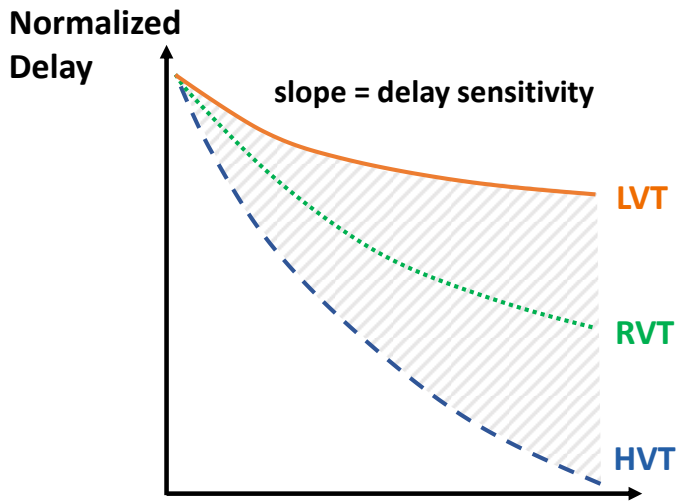


Figure 3.1 Delay sensitivity of multi-threshold voltage transistors.

voltage) conditions. As shown in Figure 3.1, a path implemented by only LVT transistors shows the lowest delay sensitivity to the change of voltage, whereas the path implemented by HVT transistors shows the highest delay sensitivity. Thus, one monitor with a single device type cannot represent the critical path delay properly. The key point is that the normalized delay curve (i.e., the sensitivity of delay to voltage variation) of any path implemented by a mix of multiple device types (LVT/RVT/HVT) would be located inside the shaded area between the LVT and HVT curves in Figure 3.1. As the operating voltage range of a chip becomes wider, from near-threshold voltage to super-overdrive voltage, the delay mismatch is aggravated requiring a large design margin because the design margin is dominated by the maximum delay mismatch.

Generic monitors are typically implemented by LVT transistors under the assumption that most critical paths consist of LVT transistors. However, they are sometimes implemented by HVT transistors when high delay sensitivity is needed. Such implementation decisions still have a limitation in estimating the circuit delay of the target block implemented with a mix of multiple device types. Instead, it is possible to mix these types with a specific ratio for a generic monitor implementation. However, it is very difficult to determine the optimal ratio of different threshold voltage transistors used in many different critical paths of the target block, which also forces the designer to give up the merit of generic monitors in terms of reusability and simplicity. To resolve these problems, we present a novel generic monitor and adaptive delay monitoring schemes in Chapter 3.2 and 3.3.

3.2 Proposed Monitoring Circuit

To overcome the limitation of a single generic monitor and still have a good reusability, we develop a combination of three generic monitors implemented respectively with LVT, RVT, and HVT transistors, considering that those types are used for the target block implementation. And the combination of multiple generic monitors can show a good accuracy with adaptive selection schemes presented in Chapter 3.3.

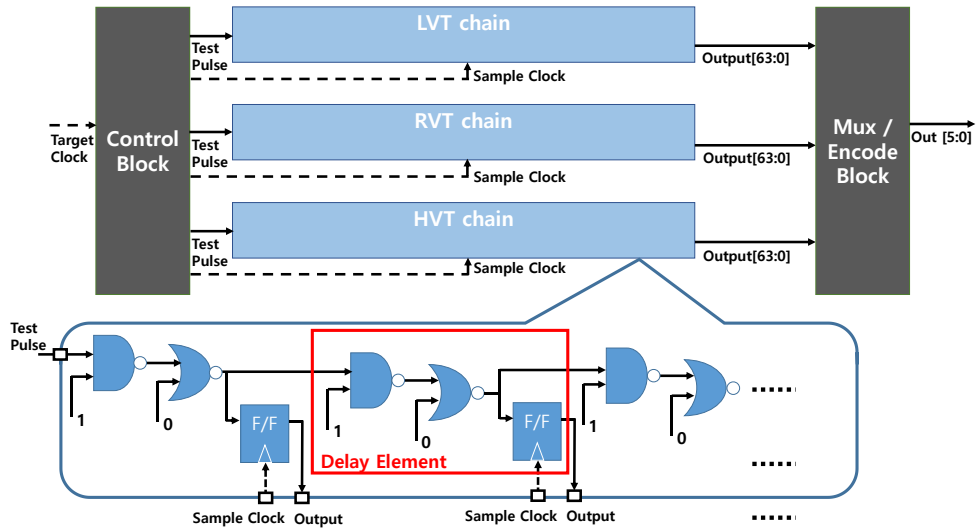


Figure 3.2 Proposed monitoring circuit structure.

We illustrate a simple view of the proposed monitoring circuit structure in Figure 3.2. Each delay chain has its own delay characteristics according to PVT (especially voltage) conditions. In some cases, it is possible to add an additional delay chain that has different delay characteristics; for example, transistors with yet another threshold voltage, a mix of different threshold voltage transistors, or different gate types can be used. We implement a monitoring circuit as an array of 64 delay elements, each of which contains a 2-input NAND cell, a 2-input NOR cell, and a flip-flop. NAND and NOR cells are used instead of INV cells to ease test operations, which require controlling the value of a flip-flop connected to the output of each delay element. And in order to estimate a circuit delay, the output (flip-flops) of the monitor should be set or reset for initialization. Also, we use NAND and NOR cells with one-to-one ratio, in order to fairly reflect the effects of PMOS and NMOS

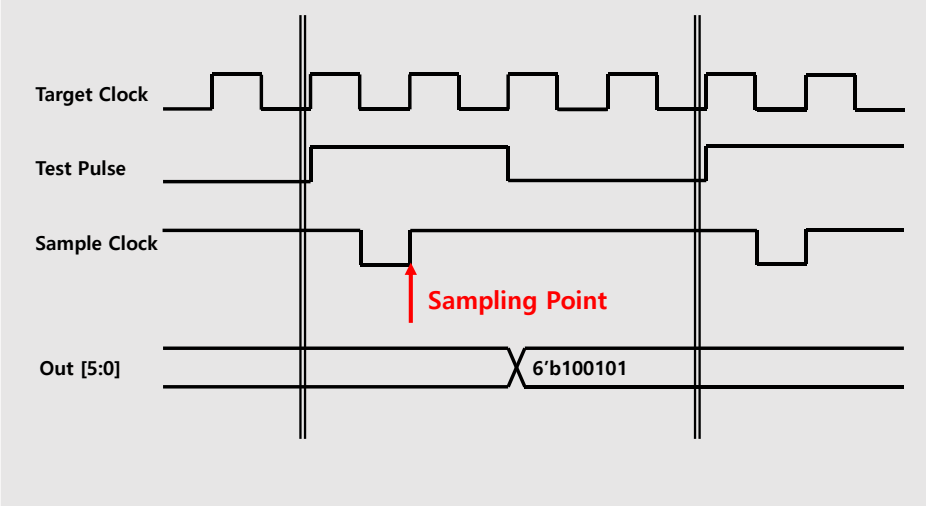


Figure 3.3 Waveform of proposed monitoring circuit.

transistors on the target circuit delay; the delays of NAND cells and NOR cells are dominated by NMOS transistors and PMOS transistors, respectively, due to the stacking structure of the cells and thus having only one type of cells in the chain might cause problems to estimate a circuit delay accurately in some process skew corners (e.g., SF/FS process corners).

The proposed monitoring circuit shown in Figure 3.2 operates as follows. First, control block generates *Test Pulse* and *Sample Clock* signals by using *Target Clock*. The *Test Pulse* signal propagates through the delay chains and *Sample Clock* is used to see how many delay elements are propagated through within one clock period. Then the output of the flip-flops (a string of 1's followed by a string of 0's) is encoded to 6-bit delay output information. The monitoring circuit outputs the information every 4-cycle period in enable mode, shown in Figure 3.3. As LVT

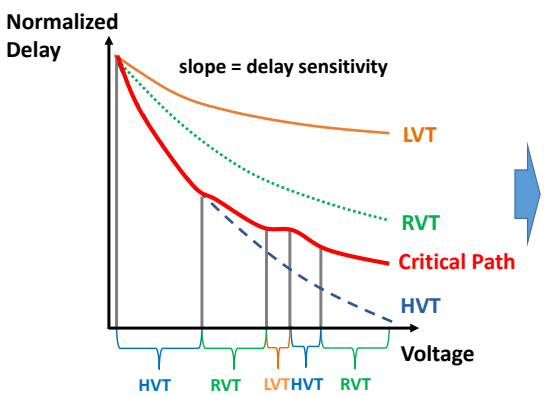
transistors are always faster than RVT and HVT transistors, the output of an LVT chain is always larger (more 1's) than that of RVT and HVT chains. And the number becomes larger if the supply voltage is scaled up and becomes smaller if it is scaled down. Thus, the output number of the proposed circuit well reflects the cell delay changes due to voltage changes.

There is an implementation guide for integrating the proposed circuit into a SoC chip. To make the effects of process and temperature variations on the monitoring circuit as close as that of the target block, the monitoring circuit should be placed closely to the target block. Also, it is desirable to share the same supply voltage rail to consider the supply voltage droop noise of the target block. These conditions are mandatory for better delay correlations between the monitoring circuit and the target block.

3.3 Adaptive Chain Selection Scheme: Hardware Approach

In general, critical paths of a target block have different delay-voltage characteristics throughout the whole range of operating conditions because they have different combinations of LVT/RVT/HVT cells and different gate types. Thus, a delay chain made of a single device type cannot properly estimate the critical path delay of the target block. We propose to use different delay chains at different combinations of voltage and temperature ranges. As the supply voltage or temperature changes, a circuit delay also changes. The rate of delay changes of the target block may match

well with that of a delay chain at a certain range of voltage and temperature. At different ranges of voltage and temperature, however, a different delay chain may match better, and that is why we use multiple delay chains. Among the delay chains, the proposed monitoring scheme adaptively selects one that matches the best with the target block at the current voltage/temperature during run-time. The voltage steps for this scheme can be determined by the resolution of a power management IC (PMIC) used in the design or configured by the designer. The finer the voltage steps, the higher the accuracy. For this, critical paths of the target block and delay chains should be characterized to obtain delay-voltage slopes throughout the entire operating voltage ranges. Regarding the temperature, since the delay is much less sensitive to temperature variations, the entire range is divided into only three sub-ranges. In the range of near-threshold voltage, the delay may be very sensitive to temperature variations, but it is not the case in the voltage range (0.6V ~ 1.2V) of our interest in this dissertation. Then, for each voltage step and temperature sub-range, the delay chain having the best matching delay-voltage characteristics is selected. That is, the proposed scheme selects the delay chain that has delay-voltage slope closest to that of the critical path and uses it for circuit delay estimation at that voltage and temperature. Before the selection, the length (number of delay elements) of each delay chain is determined such that the delay is as close to the critical path delay of the target block as possible at the given voltage and temperature. Then the delay-voltage slope of the chain with that length is used for the selection of the best matching chain. Finally, the best matching chain and its length information is stored



Voltage [V]	-40C	25C	125C
0.60	HVT, 26	HVT, 25	HVT, 22
0.65	HVT, 26	HVT, 25	HVT, 24
0.70	HVT, 28	HVT, 26	HVT, 24
0.75	HVT, 28	HVT, 26	HVT, 24
0.80	RVT, 41	HVT, 26	RVT, 40
0.85	RVT, 40	HVT, 26	HVT, 24
0.90	RVT, 40	RVT, 40	HVT, 24
0.95	LVT, 58	RVT, 40	HVT, 24
1.00	HVT, 32	RVT, 40	RVT, 40
1.05	RVT, 42	LVT, 54	RVT, 40
1.10	RVT, 42	HVT, 32	RVT, 40
1.15	RVT, 42	RVT, 42	LVT, 46
1.20	RVT, 44	RVT, 42	LVT, 48

Figure 3.4 Intuitive example of adaptive chain selection scheme.

into a LUT. The power management block (Figure 2.1) uses the information to select a proper delay chain at the current voltage and temperature and uses the length information to determine the length of the delay chain used for the monitoring. Figure 3.4 shows an intuitive example of this scheme. For each voltage and temperature range, the power management block reads the output of selected delay chain and adjusts the voltage as necessary.

To explain how we determine the contents of the LUT, the detailed algorithm for selecting a delay chain and its length is stated in Figure 3.5. Suppose there are m delay chains and n operating voltage steps. The delay and delay-voltage slope of critical paths and those of one element of m delay chains at n voltage steps should be prepared by running HSPICE simulation. They are inputs of this algorithm. At each of the n voltage steps, the chain length of each of the m delay chains is calculated such that the delay chain has the same delay as the critical path ($l_{j,i}$: length

Pseudo-code of Delay Chain Length and Selection Algorithm

Input:

n: number of voltage steps
m: number of chains
delay_critical[i]: delay of critical path at voltage step i, i=1,…,n
delay_element[j, i]: delay of one element in j-th chain at voltage step i, j=1,…,m, i=1,…,n
slope_critical[i]: delay-voltage slope of critical path at voltage step i, i=1,...,n
slope_element[j, i]: delay-voltage slope of one element in j-th chain at voltage step i, j=1,...,m, i=1,...,n

Output:

$\boldsymbol{P}_t = [(c_1, l_1)\ (c_2, l_2)\ (c_3, l_3)\ \ldots\ (c_n, l_n)]$: LUT for temperature range t, where
c_i: index of selected chain at voltage step i,
l_i: length of selected chain at voltage step i

for *each voltage step i* **do**
  for *each chain j* **do**
    $l_{j,i} = \lceil delay\_critical[i] / delay\_element[j, i] \rceil$
  end
  $k = argmin_j\ (|slope\_critical[i] - l_{j,i} \cdot slope\_element[j, i]|)$
  $(c_i, l_i) = (k, l_{k,i})$
end

Figure 3.5 Delay chain length and selection algorithm.

of j-th chain at voltage step i, j=1, ..., m, i=1, ..., n). Then the delay-voltage slope of the critical path is compared with the slope of each of the m delay chains of length $l_{j,i}$, and the delay chain having delay slope closest to the critical path slope (index k) is selected. These procedures are iteratively executed at all n voltage steps. For each temperature range t, we determine an array $\boldsymbol{P}_t = [(c_1, l_1)\ (c_2, l_2)\ (c_3, l_3)\ \ldots\ (c_n, l_n)]$ that maps a voltage step i to a delay chain k and its length $l_{k,i}$, i=1, ..., n (c_i : index of selected chain at voltage step i, l_i : length of selected chain at voltage step i). Once the array is determined, it becomes the contents of the LUT.

This approach gives much better delay correlations between the monitoring circuit and the target block than existing generic monitor approaches, without incurring a large area overhead and design complexity. Moreover, it is very easy to reuse the implementation because it does not have dependency on the target design. The only thing required is to update chain mapping and length information in LUTs. We do not claim that this monitoring circuit has higher accuracy than any other existing design-dependent monitors, but it provides both good design-for-manufacturability and accuracy.

3.4 Weighted Summation Scheme: Software Approach

To get an even higher accuracy, the adaptive chain selection scheme requires adding more delay chains that have various delay-voltage characteristics. Basically, we propose using the same number of delay chains as that of device types used for a target block implementation. However, using more delay chains brings more area overhead. As a solution, we propose a weighted summation scheme with only two delay chains for more accurate circuit delay estimation. As HVT transistors are the most sensitive to voltage change and LVT transistors are the least sensitive to that change, the delay-voltage characteristic curve of all the paths is located between that of HVT and LVT (Figure 3.1). It means that an HVT chain can work as an upper bound and an LVT chain can work as a lower bound from the view point of delay sensitivity. Using these characteristics of LVT/HVT chains, we accurately estimate

the circuit delay of critical paths through weighted summation of the two delay chain outputs. In the design step, the delay chains are characterized to obtain the weight factors for each combination of voltage and temperature steps and then the weights are stored into an LUT.

$$\boldsymbol{slope_{critical} = \alpha \cdot slope_{LVT} + (1-\alpha) \cdot slope_{HVT}}$$

$$\boldsymbol{\alpha = \frac{slope_{critical} - slope_{HVT}}{slope_{LVT} - slope_{HVT}}} \quad (1)$$

Let's denote delay-voltage slopes of LVT and HVT as $slope_{LVT}$ and $slope_{HVT}$, respectively, and that of the critical path in the target block as $slope_{critical}$. Also, let's denote the weights of $slope_{LVT}$ and $slope_{HVT}$ as α and (1 - α), respectively. We determine the value of α through the above equations.

As presented in Figure 3.5, we get the information about the delay-voltage slope of critical paths and delay chains for each combination of voltage and temperature steps. Then, it is easy to calculate the value of α for every combination. To obtain the slope information of each chain, its length should be determined beforehand in the same way as the adaptive chain selection scheme.

Even though this scheme generates a calculation overhead in the power management block, it does not require many delay chains for high accuracy. Even, the weighted summation is not a heavy computation for the power management block and it hardly affects the performance. The main role of this block is tuning the supply voltage based on the circuit delay estimation from the monitoring circuit. This scheme additionally requires setting the lengths of the delay chains and calculating

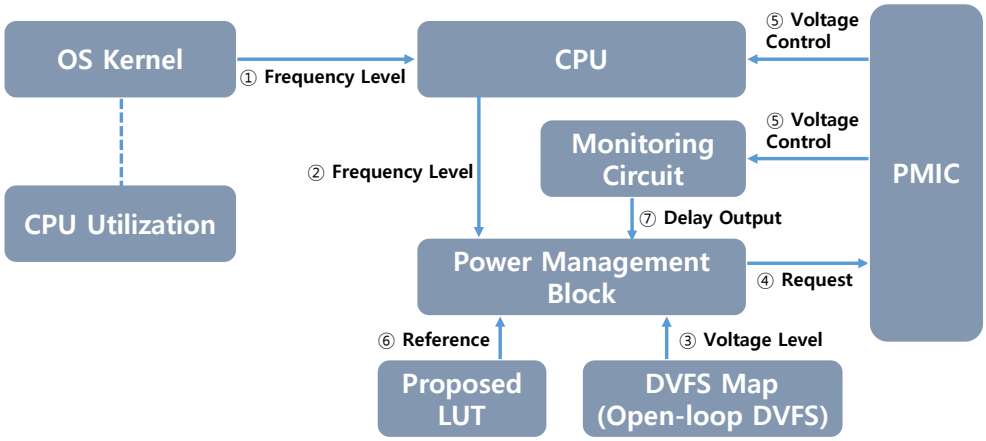


Figure 3.6 Operating scenario for proposed methods.

the weighted sum of the delay changes of the monitoring circuit outputs before voltage scaling up or down. Therefore, this scheme provides a much higher accuracy with a lower area overhead (e.g., no RVT chain) at the cost of an additional computation. In particular, the area reduction is more advantageous in many-core systems because the number of monitoring circuits is proportional to the number of cores. We analyze the experimental results of the two proposed schemes in Chapter 5.

3.5 Operating Scenario

We use a hybrid DVFS system using both open-loop and closed-loop DVFS schemes. Basically, open-loop DVFS is used for voltage/frequency scaling requested by the OS kernel, and then closed-loop DVFS and monitoring circuits are used for reducing

voltage while maintaining the frequency level. We describe the operating scenario of two proposed schemes when an increase or decrease of the frequency is requested as shown in Figure 3.6.

In case of adaptive chain selection scheme, when the CPU receives the request to increase/decrease the frequency level from the OS kernel based on CPU utilization, the CPU gives the requested frequency level information to the power management block. The power management block reads the DVFS map to obtain the voltage level for the requested frequency level and send the voltage control request to the PMIC. The power management block also accesses the proposed LUT to obtain the information on chain selection and length of the selected chain at current voltage and temperature conditions. Then it reads the output of the selected delay chain. The output of delay chain indicates the number of delay elements that the clock signal passed through within one clock period. The power management block compares the number with the chain length stored in the proposed LUT. The chain length works as the reference minimum value which secures the normal operation of the CPU. The output of the monitoring circuit should be always larger than the reference value. If the output of the delay chain is larger/smaller than (reference + Δ)/(reference + δ), $\Delta > \delta > 0$, it decreases/increases the voltage by one step (Δ and δ values are set in proportion to the reference value and thus the values are different at each voltage). Since there is a hysteresis in adjusting the delay, frequent switching of the supply voltage of the CPU is avoided. This monitoring procedure (⑦-④-⑤ in Figure 3.6)

is continuously iterated until the OS kernel sends a request to change the frequency level.

The operating scenario of weighted summation scheme is basically same as that of the adaptive chain selection scheme, except the method of getting the output of the monitoring circuit. In this scheme, the power management block reads the outputs of the two delay chains, calculates the weighted summation of the differences from the chain lengths. The remaining parts are exactly the same as the hardware approach.

Chapter 4

Design Methodology of Proposed System

The design methodology of the proposed system is shown in Figure 4.1. It is very easy to plug them into an existing design flow. First, we implement a monitoring circuit with multiple generic delay chains following the design specifications and multi-threshold voltage library usage options. We recommend that the number of delay chains should be the same as the number of libraries (one for each threshold voltage) used in the target block. With static timing analysis for timing closure, we extract the critical paths of the target block. Then we analyze the delay-voltage characteristics of delay chains and critical paths under all the operating conditions with SPICE simulation at each process corner, FF/TT/SS and so on. Based on the results of these analyses, the proposed algorithm gives the chain mapping/weight factor and chain length information for each voltage, temperature step, and process

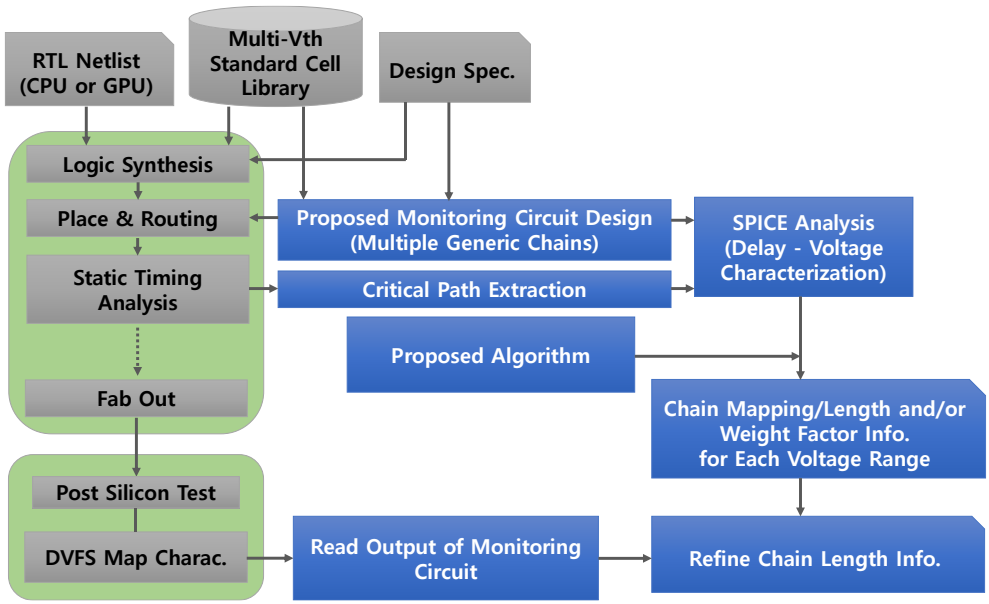


Figure 4.1 Design methodology of proposed system.

corner. In case of adaptive chain selection scheme, an additional delay chain can be added to the monitoring circuit for a better delay correlation considering the trade-off between area overhead and accuracy improvement.

However, this algorithm is based on the analysis in design time and might provide inaccurate data to be used at post-silicon step due to random process variations. That is, the variations can create changes in critical paths in the target block. The proposed method can account for this critical path changes with post-silicon characterization where the LUT is modified (only chain lengths) based on the characterization results. Here, we are assuming the application of "*speed binning*" in the characterization step, and FF/TT/SS or skewed corner (SF/FS) cases are considered as different "*bin*" groups. As the bins are different, different

configurations of the proposed monitoring circuit (different LUTs) will be used for better correlation (e.g., configuration A for TT bin, configuration B for SS bin, etc.). Even inside the same bin, there can be changes of critical paths due to variations, which will create the gap between the characteristics of monitoring circuit and target block. However, as we are configuring the monitoring circuit per bin group and the characteristics of the monitoring circuit will be centered around the nominal characteristics of target block under the speed binning condition, the design margin that is needed for covering the gap is much smaller than that of conventional monitoring circuit.

Some sample chips of each bin group are used for DVFS map characterization. In this test, each sample chip is tested by some test programs for each frequency level to obtain a minimum voltage value without an error. And then a design margin is added to this minimum voltage level and DVFS map characterizations are completed (Typically, the design margin is determined based on various factors including process parameters, operating conditions, and desired reliability levels. The issue of how much design margin should be added is beyond the scope of this dissertation). At this test time, we can also get the output number for a selected delay chain; the number means the critical path delay at the minimum voltage level to meet a target performance. It is exactly what we should define in LUTs. As the chain length information existing in the LUTs has already been determined by comparison with the critical paths at design time, it might not be much different from the measured one. However, if the difference is too large due to random process variations, we

should refine it. This procedure can be used to compensate the error due to random process variations and thus to get a more accurate information. Finally, the information on chain mapping/weight factor and chain length is stored into a LUT in memory and the power management block uses it for circuit delay estimations at run-time. The proposed approaches hardly affect the turnaround time of the target design since the characterization and mapping can be done in parallel with the existing fabrication output step.

As mentioned briefly in Chapter 3.2, we recommend that a monitoring circuit should be placed as close as the target block. Also, we recommend to use at least one monitoring circuit per core since each core has different aging rates due to different core utilizations. And, in an advanced process technology, the delay variation by on-chip process variation is too large to be ignored and the temperature inside one CPU can vary over a range of tens of degrees. It is very important to make the PVT conditions of the monitoring circuit same as that of the target block, which directly relates to an accuracy improvement of delay monitoring systems. Also, the monitoring circuit should share the same supply voltage rail as that of the target block. For instance, it should be connected to a body-bias voltage rail if the target block is connected to the body-bias voltage rail. As voltage settings are the most important factor of variations, this is a mandatory rule. These implementation guides are carefully managed when the monitoring circuit is integrating into a chip.

Chapter 5

Experimental Result

5.1 Experimental Setup

To evaluate our proposed monitoring circuit and schemes, we use an up-to-date commercial application processor, ARMv8-A Cortex-A53 design implemented in Samsung 14nm FinFET process technology. This processor is implemented by using three libraries with different threshold voltages: VTH_TYPE1, VTH_TYPE2, and VTH_TYPE3. We use FF/TT/SS process corners for HSPICE simulation and analyze the delay-voltage characteristics from 0.6V to 1.2V voltage range, where the application processor chip actually operates. The voltage granularity is set to 12.5mV to match the resolution of PMIC specifications. In case of temperature conditions, we analyze the sensitivity of delay-voltage slope to temperature at three points:

Table 5.1 Experimental environments

Target Design	ARMv8-A Cortex-A53
Process Technology	Samsung 14nm FinFET
Standard Cell Library	Multi-Vth library (3-types) VTH_TYPE1 / VTH_TYPE2 / VTH_TYPE3*
Process Corner	FF / TT / SS
Operating Voltage [V]	0.6 - 1.2
Voltage Granularity [V]	0.0125
Temperature [℃]	-40 / 25 / 125

* Due to a company confidential issue, we mark it with this terminology

minimum, maximum, and room temperature. We summarize the experimental environments in Table 5.1. In our experiment, we do not consider the error due to the resolution (granularity) of the delay chain (the typical length of a delay chain is about 20 ~ 50 and thus can incur up to 2 ~ 5% error, which should be added to the design margin).

5.2 Accuracy Results on Critical Paths

Following the proposed design methodology in Figure 4.1, we first implement a monitoring circuit using three delay chains corresponding to three different threshold voltage libraries, VTH_TYPE1, VTH_TYPE2, and VTH_TYPE3. We have analyzed the critical path distributions of our target block in various conditions and observed that all actual critical paths come from "*top 5%*" critical paths in timing-worst corner condition. There is no case of having a critical path in specific conditions, which was not in the set of top 5% critical paths. Thus, we first extract

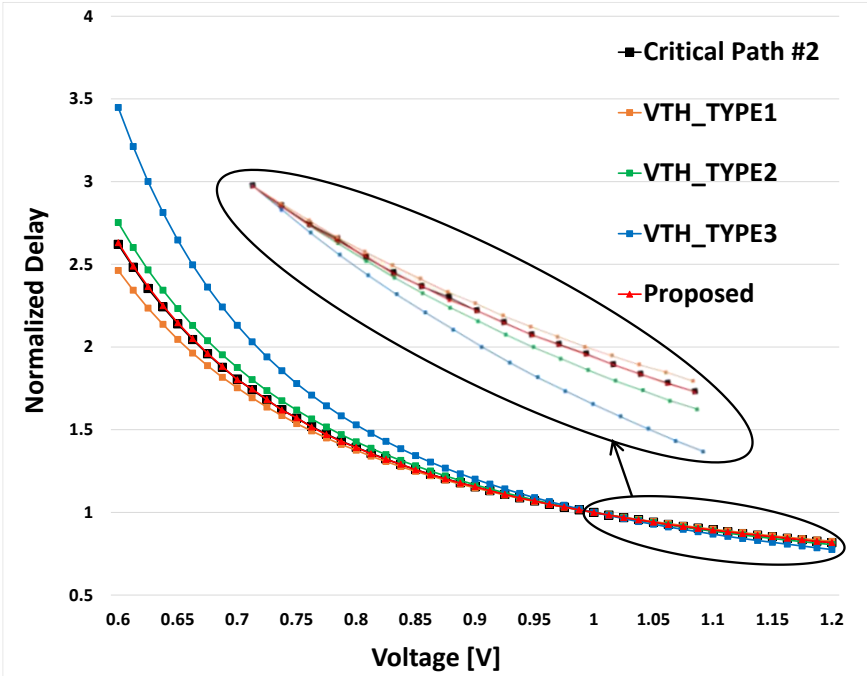
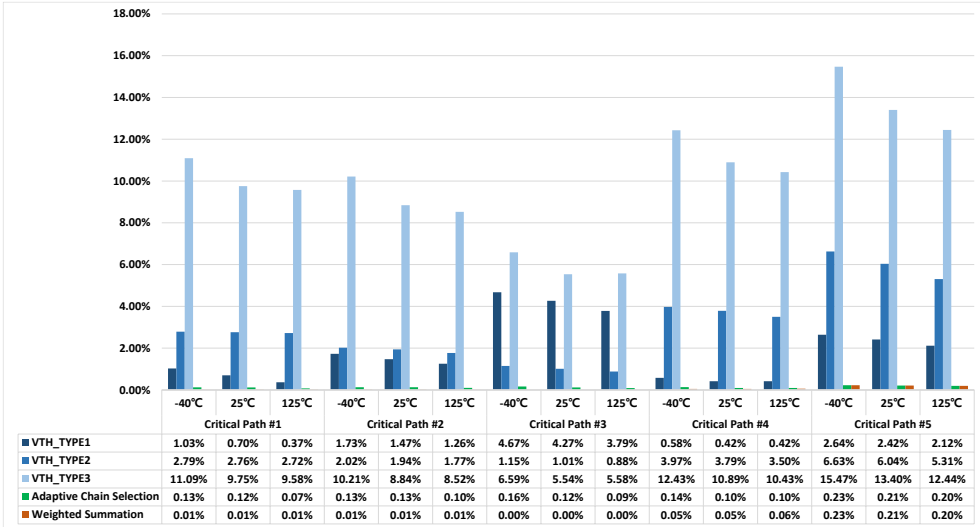


Figure 5.1 Delay-voltage characteristic curves of a critical path, single-chain generic monitors and the proposed approach (adaptive chain selection scheme).

top 5% critical paths. And then we choose 20 paths among those paths for the convenience of experiment. These paths are extracted by considering all paths: inside the core (register-to-register paths) and the interface (input-to-register paths and register-to-output paths) after the placement of our monitoring circuit and whole system simulation. Next, we run HSPICE to analyze the delay-voltage characteristics of delay chains and critical paths under various PVT conditions. In case of delay chain, we always use the same input change (pulse signal from 0 to 1) for the analysis as well as actual operations as shown in Figure 3.3. In case of critical path, we use the timing-worst case input vector which is reported in static timing analysis.

In Figure 5.1, we illustrate the delay-voltage characteristic curve of a critical path, three single-chain generic monitors (one for each threshold voltage), and the result of the proposed adaptive chain selection scheme under a given specific condition (e.g., 25℃, TT process corner). We normalize the delay of each to its own delay at 1.0V and magnify the graph at the high voltage region to see it more clearly. VTH_TYPE3 chain is the most sensitive to voltage changes and then VTH_TYPE2 and VTH_TYPE1 chains come in that order. That is, the delay-voltage slope of VTH_TYPE3 is the largest and the slope of VTH_TYPE1 is the smallest. We confirm that the delay increases fast at low voltage regions and it is accelerated at the near-threshold voltage region. As can be seen from Figure 5.1, at low voltage region, the delay-voltage characteristic curve of VTH_TYPE2 chain is closest to that of the critical path, whereas, at high voltage region, that of VTH_TYPE1 chain tracks most closely that of the critical path. However, the delay mismatch between the critical path and any single generic chain is continuously increasing as the voltage increases/decreases. We expect that those delay mismatches become larger for wider voltage range operations. However, the delay-voltage characteristics obtained by our proposed approach almost coincide with that of the critical path throughout all the operating voltage conditions. That is, the proposed approach provides almost the same delay-voltage characteristics as the critical path and thus the power management block can better control the voltage based on more accurate estimations of the circuit delay.



	-40℃	25℃	125℃	-40℃	25℃	125℃	-40℃	25℃	125℃	-40℃	25℃	125℃	-40℃	25℃	125℃
	Critical Path #1			Critical Path #2			Critical Path #3			Critical Path #4			Critical Path #5		
VTH_TYPE1	1.03%	0.70%	0.37%	1.73%	1.47%	1.26%	4.67%	4.27%	3.79%	0.58%	0.42%	0.42%	2.64%	2.42%	2.12%
VTH_TYPE2	2.79%	2.76%	2.72%	2.02%	1.94%	1.77%	1.15%	1.01%	0.88%	3.97%	3.79%	3.50%	6.63%	6.04%	5.31%
VTH_TYPE3	11.09%	9.75%	9.58%	10.21%	8.84%	8.52%	6.59%	5.54%	5.58%	12.43%	10.89%	10.43%	15.47%	13.40%	12.44%
Adaptive Chain Selection	0.13%	0.12%	0.07%	0.13%	0.13%	0.10%	0.16%	0.12%	0.09%	0.14%	0.10%	0.10%	0.23%	0.21%	0.20%
Weighted Summation	0.01%	0.01%	0.01%	0.01%	0.01%	0.01%	0.00%	0.00%	0.00%	0.05%	0.05%	0.06%	0.23%	0.21%	0.20%

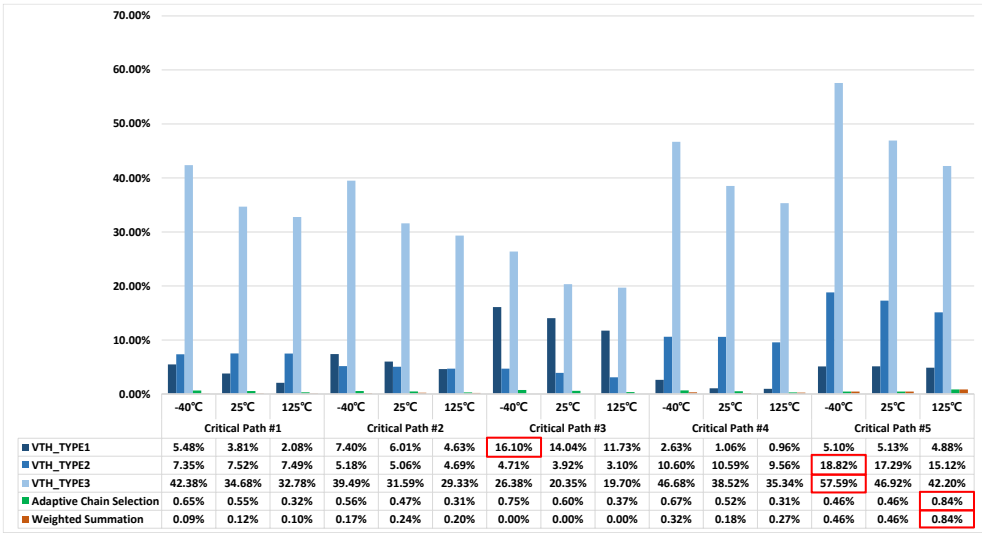
Figure 5.2 Average error rates in the delay estimation by monitoring circuits (TT process corner).

Figure 5.2 and Figure 5.3 show the experimental results for top five critical paths (#1 to #5) in TT process corner. The bars in the graphs show error rate of the three single-chain generic monitors, adaptive chain selection scheme, and weighted summation scheme under all experimental conditions.

Figure 5.2 shows average error rates in the delay estimation by monitoring circuits. In case of a conventional approach using a single-chain generic monitor, VTH_TYPE3 chain shows a large delay mismatch against the critical paths in all the operating conditions, whereas VTH_TYPE1 chain shows the best results among the three single-chain generic monitors except for critical path #3. This is because critical paths tend to use fastest transistors and thus matches well with VTH_TYPE1 which also uses fastest transistors. In critical path #3, VTH_TYPE2 chain shows a

good correlation with the critical path. It is clear that both proposed schemes have significantly low error rates compared to the single-chain generic monitors. In all critical paths, the error rates of both proposed approaches are much smaller than that of any other single-chain generic monitor. Comparing the two proposed approaches, the weighted summation scheme absolutely shows much smaller error rates except for critical path #5, since the weighted summation scheme calculates a circuit delay with an accurate weight factor instead of choosing a delay chain among candidates. Especially in critical #3, it shows almost perfect match with the critical path delay. This is the ideal case where the delay-voltage slope of the critical path is always in between lower and upper bounds in all operating voltage ranges. In critical path #5, the two proposed approaches show similar accuracy. That is because the slope of critical path #5 is even lower than that of VTH_TYPE1 chain in some voltage ranges (a path having a long wire can have a smaller delay-voltage slope than the VTH_TYPE1 chain), therefore weight factors are '1' (α = 1) in most voltage steps and thus there is no accuracy difference between the two proposed approaches.

Figure 5.3 shows maximum error rates in the delay estimation by monitoring circuits. From the view point of the design margin reduction, maximum error rates are much more important than average error rates because the design margin is determined to ensure 'no-error' operation even in the worst case. The tendency is very similar to that of average error rates, but the amount of error is much larger. VTH_TYPE3 chain shows the largest delay mismatch against the critical paths among the three single-chain generic monitors and VTH_TYPE1 chain has the



	-40℃	25℃	125℃	-40℃	25℃	125℃	-40℃	25℃	125℃	-40℃	25℃	125℃	-40℃	25℃	125℃
	Critical Path #1			Critical Path #2			Critical Path #3			Critical Path #4			Critical Path #5		
VTH_TYPE1	5.48%	3.81%	2.08%	7.40%	6.01%	4.63%	16.10%	14.04%	11.73%	2.63%	1.06%	0.96%	5.10%	5.13%	4.88%
VTH_TYPE2	7.35%	7.52%	7.49%	5.18%	5.06%	4.69%	4.71%	3.92%	3.10%	10.60%	10.59%	9.56%	18.82%	17.29%	15.12%
VTH_TYPE3	42.38%	34.68%	32.78%	39.49%	31.59%	29.33%	26.38%	20.35%	19.70%	46.68%	38.52%	35.34%	57.59%	46.92%	42.20%
Adaptive Chain Selection	0.65%	0.55%	0.32%	0.56%	0.47%	0.31%	0.75%	0.60%	0.37%	0.67%	0.52%	0.31%	0.46%	0.46%	0.84%
Weighted Summation	0.09%	0.12%	0.10%	0.17%	0.24%	0.20%	0.00%	0.00%	0.00%	0.32%	0.18%	0.27%	0.46%	0.46%	0.84%

Figure 5.3 Maximum error rates in the delay estimation by monitoring circuits (TT process corner).

smallest delay mismatch for most critical paths in all operating conditions. Only for critical path #3, VTH_TYPE2 chain is the best candidate. Also, both of the proposed schemes show much lower error rates than any other single-chain generic monitors in all critical paths and the weighted summation scheme is much better than the adaptive chain selection scheme. Again, in case of critical path #5, the delay mismatches of the two proposed schemes are similar.

As shown in Figure 5.2 and Figure 5.3, the two proposed approaches give a dramatic improvement on delay correlation considering each critical path separately. However, when we determine the design margin, we should consider all critical paths of the target block. In other words, the design margin should be determined by the critical path that renders the worst delay estimation error. In our experiment, a

thick-lined box of Figure 5.3 shows the worst error rate result and the corresponding critical path of each approach. In case of the single-chain generic monitors, the worst case of VTH_TYPE1 chain is 16.10% on critical path #3, that of VTH_TYPE2 chain is 18.82% on critical path #5, and that of VTH_TYPE3 chain is 57.59% on critical path #5. These values are directly related with the design margin that is to be added to the critical path delay. In the proposed approaches, the worst case error of the adaptive chain selection scheme is 0.84% on critical path #5 and that of the weighted summation scheme has the same value on the same critical path. Compared to the best result of single-chain generic monitors (VTH_TYPE1 chain), the proposed approaches reduce the error rates by up to 95% (from 16.10% down to 0.84%). The improvement of this delay estimation error rate absolutely brings the decrease of design margins to compensate for the delay mismatch between the monitoring circuit and the target design. And consequently it lowers design costs and power consumptions of a chip.

5.3 Accuracy Results on a Representative Critical Path

To see the effects of the proposed schemes on design margin reduction, we should apply them to all critical paths. However, it is not an efficient way and also it is very difficult to determine the modes and corners where critical paths are extracted. From the view point of whole chip operating conditions, a critical path under a specific operating condition might not be a critical path under other operating conditions. It

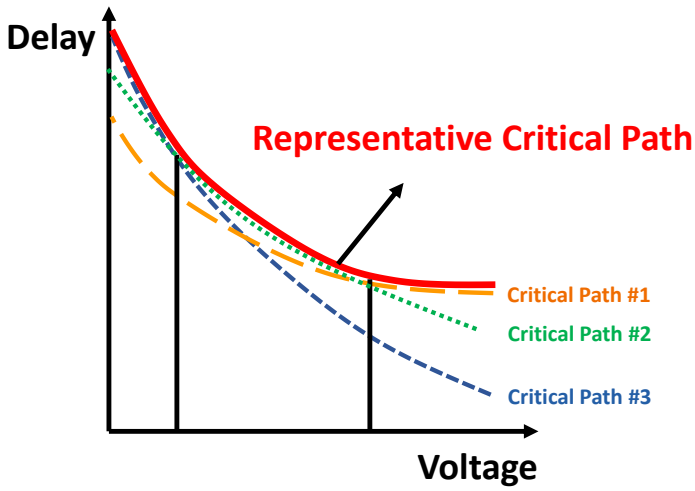


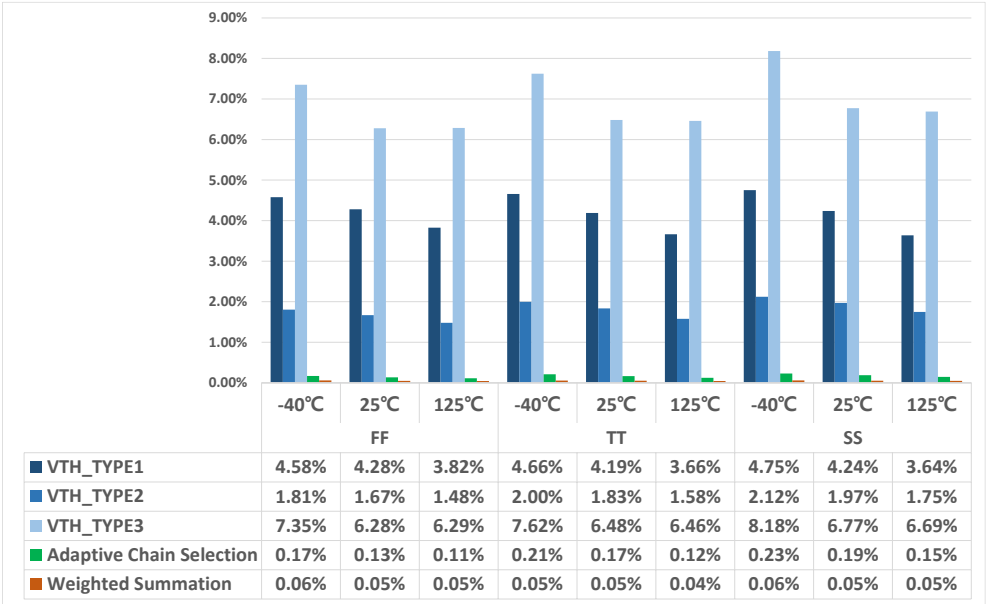
Figure 5.4 Concept of a representative critical path.

means that critical paths are changed as operating conditions (especially operating voltages) change. In Figure 5.4, for example, critical path #1 is the slowest path among the three critical paths in the figure at high voltage region, whereas critical path #3 is the slowest at low voltage region and critical path #2 is the slowest in the middle voltage region. Due to these characteristics, it is very difficult to find one path which can represent other critical paths in the whole voltage range, especially when the voltage range is wide.

To solve this problem, we extract the top 5% critical paths for the main mode of operation at the timing-worst corner. And then we define the slowest path among the critical paths for each voltage range (for a given resolution). By combining the paths, one for each voltage range, we make a virtual path which always has a critical delay in the whole voltage. This is not a real existing path, but a virtual path to

represent critical paths of a chip in all operating conditions. We call it "*a representative critical path*". Conceptually, the representative critical path is affected by "*representative critical reliability path*" [24]. When applying our proposed algorithms, we use a different critical path for each voltage range using the slowest path information. In Figure 5.4, for example, we use critical path #3 at low voltage region, use critical path #2 at middle voltage region, and critical path #1 at high voltage region for the representative critical path. That is, depending on the voltage range, different critical paths are used for comparison of delay-voltage slope with that of delay chain. The thick line in Figure 5.4 shows the delay-voltage characteristic curve of a representative critical path, which is always the most critical in all operating conditions. Actually, in case of chip operations, this path information is very important since it can define the delay characteristics of a chip. Thus, we just focus on the delay of that path instead of considering all critical paths. In a specific voltage region, a path not in the set of top 5% critical paths might become the slowest, but it has not been observed in our design and would be a very rare case.

In this experiment, we construct a representative critical path from an ARM core design and evaluate our two proposed schemes with that path. Figure 5.5 and Figure 5.6 summarize the experimental results (average and maximum error rates) for the single-chain generic monitors and the two proposed schemes. The two graphs show a similar tendency. A single-chain generic monitor using VTH_TYPE3 chain shows a large delay mismatch against the representative critical path in all voltage and temperature conditions, whereas VTH_TYPE2 chain shows the best results

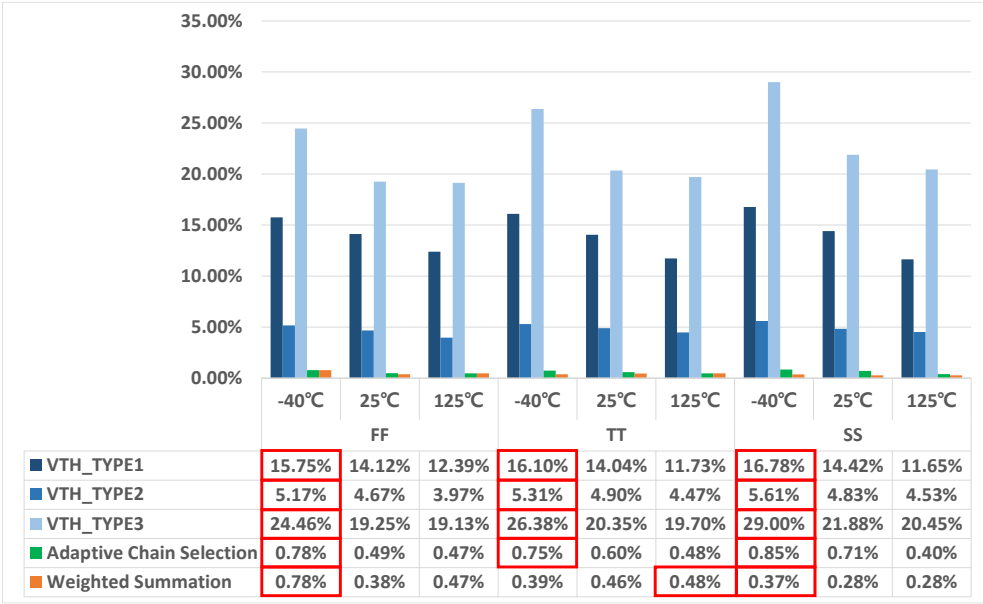


	-40℃	25℃	125℃	-40℃	25℃	125℃	-40℃	25℃	125℃
	FF			TT			SS		
VTH_TYPE1	4.58%	4.28%	3.82%	4.66%	4.19%	3.66%	4.75%	4.24%	3.64%
VTH_TYPE2	1.81%	1.67%	1.48%	2.00%	1.83%	1.58%	2.12%	1.97%	1.75%
VTH_TYPE3	7.35%	6.28%	6.29%	7.62%	6.48%	6.46%	8.18%	6.77%	6.69%
Adaptive Chain Selection	0.17%	0.13%	0.11%	0.21%	0.17%	0.12%	0.23%	0.19%	0.15%
Weighted Summation	0.06%	0.05%	0.05%	0.05%	0.05%	0.04%	0.06%	0.05%	0.05%

Figure 5.5 Average error rates on a representative critical path.

among the three single-chain generic monitors in all the conditions. Similar to the results for critical paths in Chapter 5.2, the two proposed schemes show much better delay correlation than any other single-chain generic monitors. In particular, the weighted summation scheme has a significant improvement throughout all voltage and temperature conditions.

To evaluate the effects on design margin reductions, thick-lined boxes are drawn to mark maximum error rates. As the proposed method is applied to different bin groups (different process corners), each process corner has its own boxes. In case of conventional approaches in TT process corner, the worst case of VTH_TYPE1



	-40℃	25℃	125℃	-40℃	25℃	125℃	-40℃	25℃	125℃
	FF			TT			SS		
VTH_TYPE1	15.75%	14.12%	12.39%	16.10%	14.04%	11.73%	16.78%	14.42%	11.65%
VTH_TYPE2	5.17%	4.67%	3.97%	5.31%	4.90%	4.47%	5.61%	4.83%	4.53%
VTH_TYPE3	24.46%	19.25%	19.13%	26.38%	20.35%	19.70%	29.00%	21.88%	20.45%
Adaptive Chain Selection	0.78%	0.49%	0.47%	0.75%	0.60%	0.48%	0.85%	0.71%	0.40%
Weighted Summation	0.78%	0.38%	0.47%	0.39%	0.46%	0.48%	0.37%	0.28%	0.28%

Figure 5.6 Maximum error rates on a representative critical path.

chain is 16.10%, that of VTH_TYPE2 chain is 5.31%, and that of VTH_TYPE3 chain is 26.38% in -40℃.

In the proposed approaches, the worst case error of the adaptive chain selection scheme is 0.75% in -40℃ and that of the weighted summation scheme is 0.48% in 125℃. Compared to the best result of the single-chain generic monitors (VTH_TYPE2 chain), the adaptive chain selection scheme reduces the maximum error rates by up to 86% (from 5.31% to 0.75%) and the weighted summation reduces the maximum error rate by up to 91% (from 5.31% to 0.48%). These evaluation results show the potential decrease of the design margin for compensating the delay mismatch between the monitoring circuit and the target block. That is, the advantage

Table 5.2 Area overhead of proposed approaches

		Normalized Area*	Ratio [%]
ARMv8-A Cortex-A53		4774.07	99.98
Monitoring Circuit	Adaptive Chain Selection (Three Generic Chains)	1.85	0.04
	Weighted Summation (Two Generic Chains)	1.43	0.03
	Conventional Approach (Single Generic Chain)	1	0.02

of closed-loop DVFS scheme can be maximized by using the proposed approaches and thus the power consumption can be minimized. We also observe similar accuracy improvements in other process corners.

5.4 Area Overhead and Accuracy Comparison

We propose two delay monitoring schemes with multiple generic monitors. These approaches provide a much higher accuracy than the conventional approach using a single-chain generic monitor, but it inevitably brings an additional area overhead due to the use of multiple generic monitors. We summarize the area overheads of the conventional approach and the two proposed approaches in Table 5.2.

The area overheads of the two approaches are 85% and 43%, respectively. These numbers might be considered as large area overheads, but they still incur small area overheads compared to design-dependent monitors. And the area of a monitoring circuit using generic chains takes a very small portion of the total chip die area. Shown in Table 5.2, the monitoring circuit area is about 1/2500 of ARM

Table 5.3 Accuracy comparison with up-to-date design-dependent monitors

	[18]	**[20]**	**[23]**	**Proposed Approach**
Error Rate [%]	1.15 ~ 2	1.25 ~ 1.5	0 ~ 2	0 ~ 0.85

Cortex-A53 area, which is used for a little core in a big-little architecture of an application processor chip.

Table 5.3 shows the accuracy comparison between up-to-date design-dependent monitors and our proposed approach. It shows a comparable accuracy with the design-dependent monitors.

Chapter 6

Conclusion

We propose a monitoring circuit composed of multiple generic chains and two monitoring schemes. One scheme adaptively selects a proper chain among multiple chains and the other scheme uses two boundary chains to monitor the delay through weighted summation. The proposed design methodology can be easily integrated into existing design flows. And it significantly reduces the maximum delay mismatch between the monitoring circuit and the target design by up to 91% for a wide range of operating voltage without large area overheads. Therefore, it can reduce the design margins for compensating the delay mismatch in the closed-loop DVFS scheme. In an advanced process technology, the effects of variations on a circuit delay can be much more serious and thus the accuracy improvement of measured circuit delay becomes very important. The proposed delay monitoring

systems can effectively resolve this problem and thus achieve a low power and/or low cost design.

Part II

Aging Gracefully with Approximation

Chapter 7

Introduction

Recently, the chip reliability problem is getting much worse as the process technology scales down. Among others, bias temperature instability (BTI) is a key reliability problem to degrade the chip performance by increasing the threshold voltage and decreasing the drain current [25]. This incurs a chip slowdown, and after all, generates timing violation errors. In addition, the aging accelerates in metal-oxide thin-film transistors and the aging-induced delay incurs much larger timing variations in low supply voltage systems. Therefore, aging is a more serious problem in low power systems such as internet-of-things (IoT) or biomedical devices.

The conventional approach to compensating for this aging-induced timing violation error is assigning a reliability design margin to supply voltage [26]. However, there are many problems in applying this approach to low power design.

First of all, it is a too pessimistic approach since it should assign a relatively large design margin that can compensate for the chip slowdown after many years (say 10 years) of aging. This approach wastes extra power/area that is unnecessary during the first 10-years. Secondly, it is very difficult to determine an accurate design margin at design-time because the chip slowdown by aging depends on operating conditions including supply voltage, temperature, and application scenario. That is, the design margin should be determined by considering the worst-case operating conditions; it is very risky to determine the design margin by considering the statistical or balance conditions because it cannot ensure the normal chip operation in worst-case conditions. Thirdly, increasing the supply voltage to secure the design margin makes the aging accelerate.

Instead of increasing supply voltage and/or using faster (but larger) circuit as in the conventional approach while maintaining the accuracy, adopting the concept of approximate computing can be a more efficient solution in applications such as image/video processing, where the output quality is less sensitive to small errors. Under the assumption that the quality degradation by the approximation is not large, approximate computing can effectively increase performance and/or reduce power consumption. Therefore, simplified or approximate arithmetic circuits (adders, multipliers) are widely used to generate acceptable quality results in approximate computing applications, especially in image/video processing applications [27][28].

In this dissertation, we propose an approach that enables the system adapt to aging dynamically. Thus, it does not need to add the reliability design margin at

design-time. Instead, it monitors the aging-induced delay at run-time and compensate for the increased delay by curtailing the critical path in a way of minimizing the accuracy loss due to the approximation. Therefore, it is essential to measure the chip slowdown due to aging and compensate it adequately at run-time. For the implementation, we use an on-chip monitor, presented in the first part of this dissertation, and measure the chip performance periodically. It does not need to measure the chip slowdown very often because the aging mechanism is a very slow process. Ideally, this approach does not need to add any reliability design margins. Practically, however, we put a small design margin to compensate for the delay mismatch between the monitoring circuit and the actual critical path delay of the target block. The approach also requires reconfigurable circuits that can adjust the level of approximation. In this dissertation, we show how to design a reconfigurable adder. Instead of adopting approximate computing, we can increase the supply voltage dynamically as the aging-induced delay increases. Again, however, it might accelerate the aging process and also increase the power consumption.

The second part of this dissertation is organized as follows. Chapter 8 describes a motivational case study to show the effect of aging-induced timing violation errors in an image processing application and introduces previous work on various approximate adders and approaches to aging compensation. Chapter 9 presents the proposed system to compensate for the aging-induced delay with approximation at run-time. Chapter 10 describes the design flow of the proposed

system. Chapter 11 presents the experimental results and the analyses. Finally, Chapter 12 summarizes our proposal with concluding remarks.

Chapter 8

Motivational Case Study and Related Work

8.1 Motivational Case Study

The main issue is whether or not a reliability design margin is essentially required in every application. Conventionally, even in error-tolerant systems, the design margin is indispensably required to gain reasonable outputs. To demonstrate this, we experiment with an image processing system that performs discrete cosine transform (DCT) and inverse discrete cosine transform (IDCT). For aging simulation, we leverage degradation-aware cell libraries which have been recently proposed and made publicly available [29]. The libraries are compatible with existing EDA tool flows like Synopsys and hence one can directly use them to perform static timing analysis of a circuit netlist without requiring any modifications. In practice, these

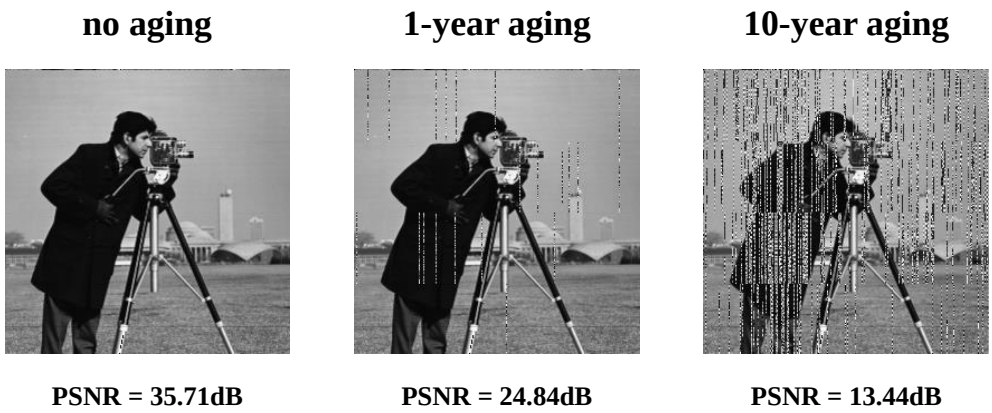


Figure 8.1 Impact of the aging-induced delay in an image processing application.

libraries contain the delay information of standard cells under the effects that aging has on the electrical characteristics of NMOS and PMOS transistors (e.g., threshold voltage and carrier mobility). Detailed experimental setup is described in Chapter 11.

As shown in Figure 8.1, removing the design margin in the design results in a significant quality drop when encoding and then decoding an image. In such a chain of circuits, errors are increasingly accumulated. The errors first occur in the encoder and then they are propagated to the decoder which leads to a larger impact on the quality of the final output image. In a more advanced process technology, this problem becomes worse and worse because the aging-induced delay by BTI and hot carrier injection (HCI) increases [30].

8.2 Related Work

Many approaches have been studied to avoid aging-induced timing violation errors. The conventional approach to compensate for the errors is assigning an additional reliability design margin to supply voltage (or to slack of the critical path) [26]. However, it is too pessimistic and it is very difficult to determine an accurate design margin at design-time. In [29], the circuits are optimized against aging through logic synthesis with degradation-aware cell libraries. It enables the optimization process to select most suitable cells for each set of operating conditions such as input slew and output load capacitance, while considering aging effects. The work in [31] quantifies the impact of aging-induced errors and approximation errors on quality loss when the design margin is removed. It replaces the design margin with an equivalent reduction of precision in approximate computing applications. The aging effects are characterized and applied at design-time. However, such design-time approaches naturally renders an overdesign. A more aggressive optimization can be done by measuring the chip slowdown due to aging and compensating it at run-time. There are many techniques for variation-resilient design that allow timing violation errors and manage the design reliability dynamically [32]. For the measuring of the chip slowdown, many on-chip aging monitors are presented [33]-[36]. However, all these approaches are through scaling the voltage to suppress the errors for accurate computing.

Our proposal is the first to measure the aging-induced delay of a basic arithmetic circuit (we focus on adders in this dissertation) in an approximate computing system and then truncate its least significant bits (LSBs) to reduce the critical path delay at run-time if it is increased by aging. Thus, to avoid timing violation, the approach adjusts the approximation level of the arithmetic circuit instead of increasing the supply voltage. Note that most significant bit (MSB) errors generated when signals cannot propagate to the MSB within a clock period due to aging-induced delay is much more critical than the errors generated by truncating LSBs.

Various approximate adders that can trade off accuracy for power/speed are studied. The approach in [37] splits input operands into two parts: accurate part and inaccurate part (the accurate part is on the MSB side, while the inaccurate part is on the LSB side). The approach in [38] from the same research group splits the operands into multiple sets of bits resulting in multiple sub-adders combined with carry chains. Then they improve the speed of carry chains by applying the concept of carry select adder [39]. The re-design technique for yield improvement is presented in [40]. It replaces some original modules with simplified functional modules to reduce critical path delay. As the functional modules, they consider approximate adders to truncate some bits to get power/speed improvement. All those approaches do not consider the aging-induced timing violation errors and the circuit structures are fixed at design-time.

Various configurable adders, which can compensate the errors at run-time, are also studied. The accuracy configurable adder proposed in [41] changes the accuracy of results by selecting the operation mode during run-time. It basically outputs approximate results by cutting propagation path, and if needed, it corrects the errors for accurate results with multiple overlapping sub-adders. The gracefully-degrading adder in [42] comprises of fixed multiple sub-adder units with selectable length for carry prediction bits, which is limited to multiples of the sub-adder length. It does not provide an error function that can predict the accuracy of the selected configuration and any error correction units. The generic accuracy configurable adder in [43] is comprised of multiple smaller sub-adders with carry prediction and error correction units. However, it is configured at design-time to meet the requirement of delay, area, and accuracy. Most of the state-of-the-art configurable adders have some error detection and correction units and thus incur design overhead. They do not consider the aging-induced timing violation errors, but just configure the structure while satisfying the required accuracy. Whereas they are basically approximate adders, our proposed adder is basically an accurate adder which can become an approximate one by reducing its bit-width. Since the bit-width can be reduced dynamically at runtime, the proposed adder does not need to set the initial reliability design margin, but can start with a full precision quality and then degrade the accuracy gracefully as the delay increases by aging. And it does not generate large design overheads, but just needs a couple of gates. In Chapter 9, we describe

in detail the proposed schemes and circuits to avoid the aging-induced timing violation error.

Chapter 9

Proposed System

9.1 Overview of the Proposed System

Figure 9.1 is the simplified block diagram of the proposed system. It comprises of the monitoring circuit, control unit, and the target block. The target block implements an approximate computing application such as an image/video codec. We consider using our proposed adders in the target block. The monitoring circuit outputs the delay information in the target block under the current operating condition. It can measure the aging-induced delay of the adders under specified operating conditions. The proposed adders have switches to cut-off the carry propagation paths according to configuration inputs. They can reduce the critical path delay of the adders by truncating LSBs according to the measured aging-induced delay information. As

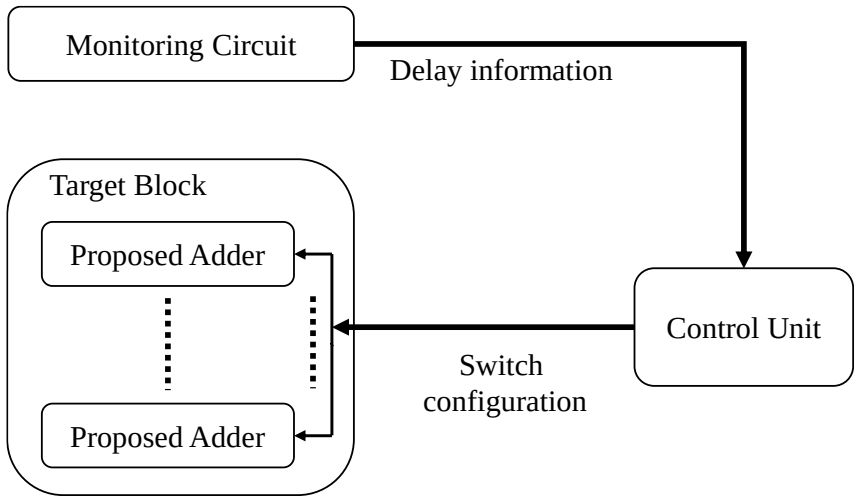


Figure 9.1 Simplified block diagram of the proposed system.

mentioned in Chapter 8.2, it basically operates as an accurate adder, but becomes an approximate adder if the aging is detected by the monitoring circuit. The control unit sets the configuration input of the adders according to the delay information from the monitoring circuit. The overhead is not too large because it just translates the 5-bit output of the monitoring circuit into the 4-bit input of the adders. Detailed explanations of each circuit block are given in Chapter 9.2 and 9.3.

9.2 Proposed Adder

The proposed adder structure is based on ripple carry adder, which is the most cost/power-saving adder among conventional adders. It shows the lowest power consumption and the best power-delay product metric, compared to other

conventional accurate adders [37]. So, it has been widely chosen for the low power design and we also start from the ripple carry adder structure. However, in a conventional ripple carry adder, errors in the most significant part (we call them MSP errors) are generated when the carry signal cannot be propagated to the MSP positions during one clock period due to the aging-induced delay. Such MSP errors are much more critical than errors in the least significant part (we call them LSP errors), especially when the sign bit in the 2's complement representation is involved in the errors.

To resolve this problem, we present two types of adders—masking and cutting—to prevent MSP errors. Figure 9.2 shows the structure of the two proposed adders. The difference from the conventional adder is that the proposed adders have a 4-bit switch to cut-off the carry propagation path. These circuits reduce the critical path delay according to the configuration input value. The masking type adder truncates some LSBs of the adder input to cut-off the propagation path, while the cutting type adder blocks the carry propagation from some LSBs (we will refer to those two types of gating as "*truncation*" afterwards). For example, in case of masking type adder, when setting the configuration input to mask [3:0] = 4'b1011, the carry out of the third full adder (FA) is always zero. Then the critical path becomes shorter; the new critical path is from a [3] to sum [31]. Also, in case of cutting type adder, when setting the configuration input to cut [3:0] = 4'b0111, the carry out of the fourth FA is always gated not to propagate into the next (fifth) FA. They are configured dynamically at run-time only when the aging-induced delay is

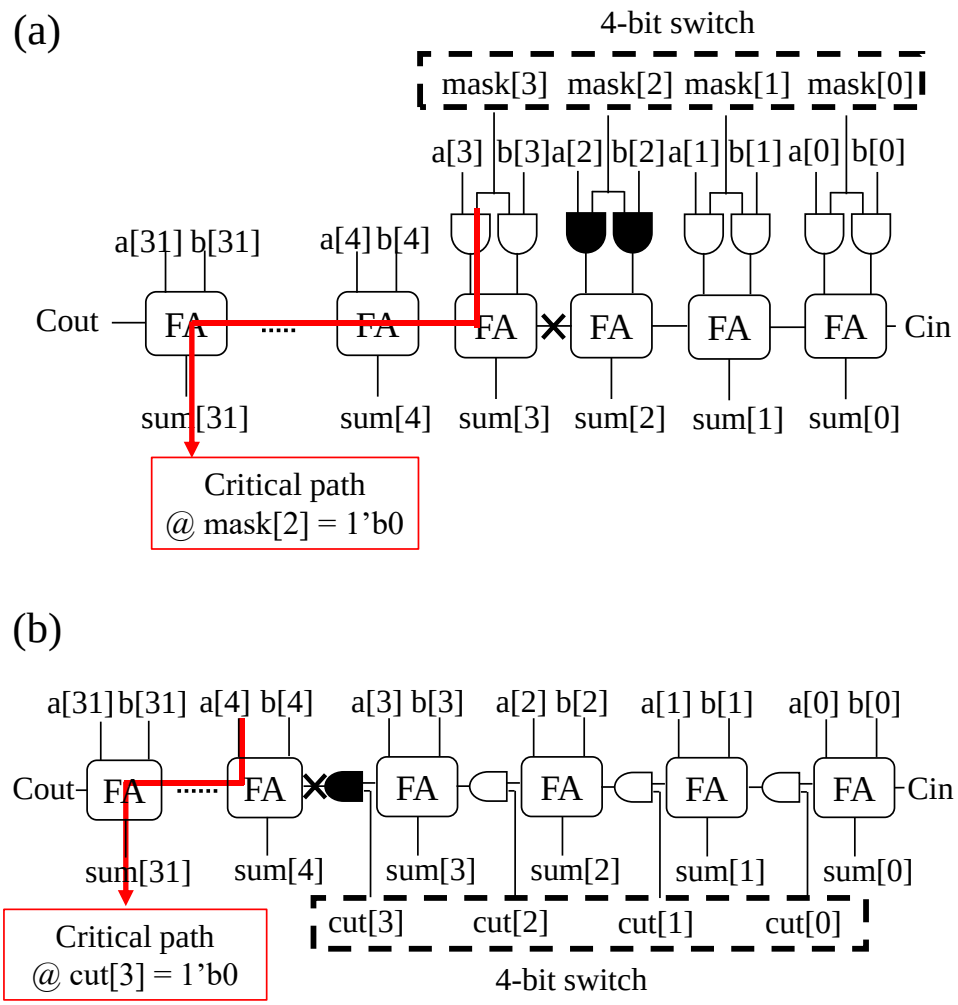


Figure 9.2 Proposed adders (a) masking type (b) cutting type.

detected by the monitoring circuit. The blocking of carry propagations in these examples may generate many LSP errors instead of a few critical MSP errors. In our experiments, we observe that the approach increases the error rate but significantly lowers the mean squared error (MSE).

In Figure 9.2, we show only the 4-bit switch to cut-off the carry propagation path. However, the optimal number of bits depends on the maximum amounts of the delay increase due to aging, which in turn depends on operating conditions and process technology. And the proposed adders are not the only components that are applicable for this system. Other types of adder or other arithmetic circuits, which can be configured to reduce the critical path delay with precision reduction, are also applicable.

9.3 Monitoring Circuit

The monitoring circuit is a key component to determine the quality and effectiveness of the proposed system. More accurate aging-induced delay measurement makes the quality degrade more gracefully with approximation. On the other hand, inaccurate measurement might bring excessive quality degradation or still incur MSP errors, in spite of using the run-time monitoring system. Dynamic approach gives better quality results than static approach under the assumption that the sensor/monitor gives accurate information.

We design the monitoring circuit to be used for the proposed system based on the monitoring circuit, presented in the first part of this dissertation. Figure 9.3 shows the detailed structure of the monitoring circuit. It consist of three main blocks: delay chain, control block, and encode block. Delay chain is an array of 32 delay elements, each of which contains 2-input NAND, NOR, XOR cells, and a flip-flop. The

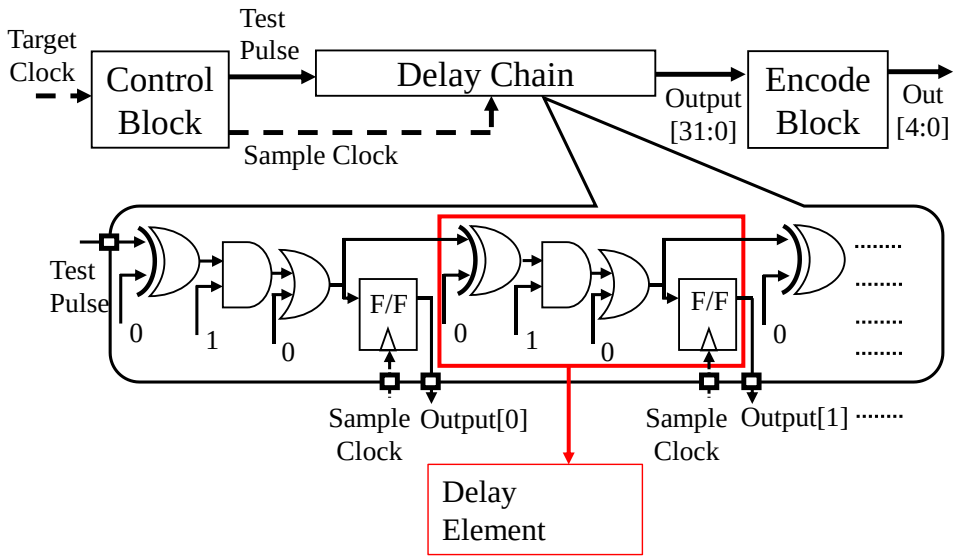


Figure 9.3 Proposed monitoring circuit with 32 delay elements.

combinational cell composition of a delay element is the same as the critical path of an FA. That is, the whole delay chain has completely the same cell composition as the critical path of the 32-bit ripple carry adder. That is why the monitoring circuit can measure the aging-induced delay of the proposed adder accurately. The control inputs of the delay chain are assigned with '0' or '1', in order to propagate the input signal through the delay chain correctly. Although the delay chain of our monitoring circuit has 32 delay elements, the number of delay elements depends on the number of adder bits.

The monitoring circuit operates as follows. First, the control block generates *Test Pulse* and *Sample Clock* signals by using *Target Clock*. While the *Test Pulse* signal propagates through the delay chain, the *Sample Clock* samples it to see how

many delay elements are propagated through within one clock period. Then the output of the flip-flops (a string of 1's followed by a string of 0's) is encoded to 5-bit delay output information. For example, at initial year (0-year without the design margin), the output [31:0] is always 32'hffff_ffff. However, if the input pulse signal cannot propagate through the whole delay chain within one clock period due to the aging-induced delay, it becomes 32'hffff_fffe, 32'hffff_fffc, 32'hffff_fff8, or 32'hffff_fff0 as the amount of delay increases. Based on this output information from the monitoring circuit, the control unit can cut-off the appropriate carry propagation path.

Aging is a very slow process with alternating phases of stress and recovery. Therefore, the monitoring and the closed-loop control operations do not need to be executed at all times. However, the monitoring circuit requires to have switching activity similar to the adders in the target block in order to have closely correlated aging characteristics.

9.4 Aging Compensation Scheme

We illustrate the aging compensation scheme with approximation in Figure 9.4. At 0-year (no aging), the proposed adders of target block operate as accurate adders. And the monitoring circuit periodically gives the speed information to the control unit. If the aging-induced delay is detected, the control unit determines the number of LSBs to be truncated as the amount of delay increase by aging. Then the proposed

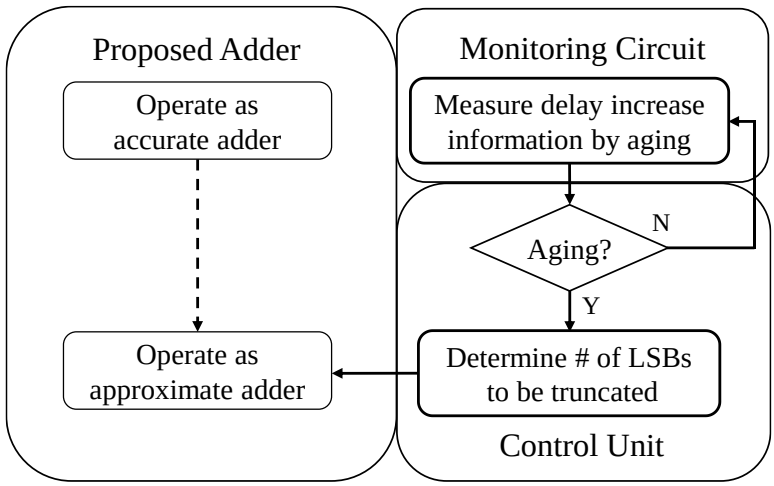


Figure 9.4 Aging compensation scheme with approximation.

adders operate as approximate adders by truncating some LSBs. When the delay by aging further increases, the control unit configures the adders to truncate more LSBs. This scheme is automatically operated at run-time. We do not have to run the time-consuming simulation at design-time. Just one simulation run is required to determine the maximum number of LSBs to be truncated during the projected lifetime, say 10 years.

Chapter 10

Design Methodology of Proposed System

It is required to implement the proposed adder and monitoring circuit for a given design specification and process technology. The design methodology of the proposed system is shown in Figure 10.1. The flow chart on the left-hand side shows the design analysis steps for the proposed system implementation, and the flow chart on the right-hand side is the design integration steps to integrate the proposed circuits into the target block design. It is very easy to plug the proposed methodology into a conventional design flow.

In the design analysis steps, we synthesize a ripple carry adder as the reference adder structure with general standard cell library, which does not consider the aging-induced delay. With static timing analysis, we analyze the aging-induced timing violation errors after projected lifetime, say 10 years, with the degradation-aware

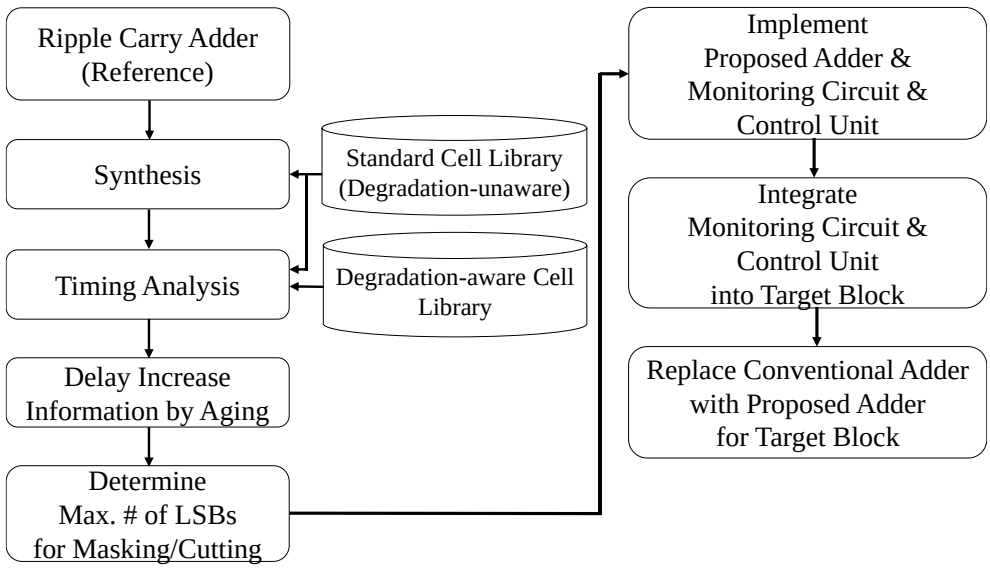


Figure 10.1 Design methodology of proposed system.

cell libraries [29]. These libraries are characterized under various aging periods (1-year, 2-year, ..., 10-year). We can calculate the quality degradation by aging, by comparing the timing analysis result at 0-year to that after 10-year aging. Based on the results of this analysis, we can determine the maximum number of LSBs to be truncated for aging compensation.

Next, in the design integration steps, we implement the monitoring circuit, control unit, and the proposed adder according to the specification determined in the previous steps. And we integrate the monitoring circuit and control unit into the target block. There are some important guidelines to correlate the delay characteristics between the delay chain of the monitoring circuit and the carry propagation path of the proposed adder. As the cell delay is mainly affected by PVT,

it is very important to make the PVT conditions of the monitoring circuit same as that of the adders in the target block, which directly relates to an accuracy improvement of the proposed system. First, the monitoring circuit should share the same supply voltage rail as that of the adder in the target block to correlate them closely in terms of delay characteristics and aging process, both of which are voltage dependent. Secondly, the monitoring circuit should be placed closely to the target block because the delay variation due to on-chip process variation is too large to be ignored in an advanced process technology and the temperature inside one chip can vary over a range of tens of degrees. These guidelines are a mandatory rule for better delay correlations between the monitoring circuit and the adder in the target block, and thus they should be followed carefully when the monitoring circuit is integrating into a chip. Finally, the conventional adders are replaced with the proposed adders in the target block.

From the view point of hardware, this system does not generate a large area overhead. Compared to the conventional ripple carry adder, the overhead of the proposed adder can be ignored because a couple of AND gate cells is added to cut-off the carry propagation path. The monitoring circuit and the control unit are very simple logic. In addition, this system does not require voltage scaling when the delay increases, because it compensates the aging-induced delay increase by approximation. So, it does not incur additional power consumption due to the voltage scaling or assigning the reliability design margin. However, it might be required to

assign a narrowed design margin since there can be a delay mismatch between the monitoring circuit and the adders in the target block.

Chapter 11

Experimental Result

11.1 Experimental Setup

Two experiments are conducted to evaluate our proposed system and scheme. First, we evaluate the proposed adder solely with randomly generated inputs. We implement two types (masking and cutting) of 16-bit and 32-bit proposed adders with a 2-bit and 4-bit switch, respectively, to cut-off the carry propagation. 100K random inputs with normal distributions are generated using *$dist_normal()* functions in Verilog code. Mean and standard deviation values in [44] are used for this test input generation. The random inputs with normal distribution have similar characteristics with the actual input extracted from an image encoder/decoder block. We employ Synopsys Design Compiler to synthesize the RTL code of the proposed

adders with the degradation-aware cell libraries, based on the 45nm Nangate process technology [29]. We perform static timing analysis of the synthesized netlist under aging. We employ Synopsys Prime Time for the static timing analysis and power estimation. We analyze the timing information while changing the configuration of the 2-bit/4-bit switch with the *set_case_analysis* command. Gate-level simulation is executed with Mentor ModelSim, in order to analyze the MSE and error rate by aging-induced delay. Standard delay file (.sdf) is used to consider the aging-induced delay for the gate-level simulation.

Secondly, we evaluate the proposed system and scheme with real application environment at the microarchitecture level. We perform the experiment with DCT and IDCT circuits to encode and decode images, which are widely used for image codec applications. We connect the output port of DCT and the input port of IDCT directly, to encode and then decode the image input file and see the impact of aging-induced delay on the image codec. And we use the 256 x 256 representative image input files for image processing evaluation. The peak signal-to-noise ratio (PSNR) metric is used for the evaluation of image quality. In Chapter 11.3, we show the image quality change at 1-year and 10-year with/without the proposed compensation scheme. These results are generated by gate-level simulation with the degradation-aware cell libraries [29]. It takes a couple of hours to get the results of DCT (encoding)-IDCT (decoding) chain.

11.2 RTL Component Level

We implement the two types of the proposed 16-bit/32-bit adders, which can be configured to truncate two/four LSBs. It is because the delay increase by aging (about 8%) can be sufficiently compensated by the 2-bit/4-bit truncation.

Table 11.1 shows the comparison of power, area, and critical path delay between the conventional 16-bit ripple carry adder and the two proposed adders. In terms of accuracy at the component level (i.e., adder), we use the two aforementioned error metrics (MSE and error rate). We use 100K random inputs with normal distribution for the two 16-bit inputs of the adders, since they can better represent inputs for general image processing applications [44].

The conventional ripple carry adder is used as a reference. It generates aging-induced timing violation errors when the reliability design margin is not included. The error rate increases by up to 1.35% and the MSE value also increases significantly, which is due to the critical path delay increase by about 8.4%. We confirm that over half of the aging-induced delay incurs within the first year, which is about 60% of the total delay increase by aging during the projected lifetime. Total power consumptions gradually increase due to the aging effects. The reasoning is as follows. Static power is always reduced due to aging since the aging increases the threshold voltage of transistors. However, dynamic power depends on the circuit design and application, and thus it can increase or decrease by aging [45]. In our experiment, the dynamic power of the system increases as the aging progresses and

Table 11.1 Comparison of 16-bit ripple carry adder and proposed adder

16-bit Ripple Carry Adder							
Aging Time	**0-year**	**1-year**			**10-year**		
Power (uW)	13.39	13.45			13.56		
Area	76.61						
Critical Path Delay (ns)	0.990	1.041			1.073		
Error Rate	0.00%	0.64%			1.35%		
MSE	0.00	6.84E+06			9.17E+06		
16-bit Proposed Adder - Masking							
Aging Time	**0-year**	**1-year**			**10-year**		
# of Truncated LSBs	**-**	**0**	**1**	**2**	**0**	**1**	**2**
Power (uW)	13.50	13.98			14.13		
Area	80.86						
Critical Path Delay (ns)	1.031	1.084	1.020	0.913	1.116	1.050	0.941
Error Rate	0.00%	0.65%	74.96%	75.13%	1.42%	75.18%	75.13%
MSE	0.00	6.96E+06	1.49	5.98	1.00E+07	8.97E+06	5.98
16-bit Proposed Adder - Cutting							
Aging Time	**0-year**	**1-year**			**10-year**		
# of Truncated LSBs	**-**	**0**	**1**	**2**	**0**	**1**	**2**
Power (uW)	13.33	13.59			13.76		
Area	78.74						
Critical Path Delay (ns)	1.024	1.076	0.995	0.913	1.108	1.025	0.941
Error Rate	0.00%	0.07%	49.97%	49.93%	1.54%	50.58%	49.93%
MSE	0.00	7.24E+06	2.00	7.99	1.10E+07	1.29E+07	7.99

the amount of increase is much larger than the amount of decrease of static power. Thus, the total power consumptions of all the adders increase by aging (we assume that the system is turned off when not in use).

In case of the proposed adders, two and four AND cells are inserted for cutting and masking types, respectively. Due to these small overheads, the delay, area, and power consumption increase a little bit compared to the reference. At 10-year, the delay of these adders increase by about 8.2% due to the aging and the error rate

increases by up to 1.42% and 1.54%, respectively. MSE shows results similar to the reference. The critical path delay decreases by configuring the switch to cut-off the carry propagation path. At 1-year, the critical path delay of 1-bit truncation is smaller than the delay of no truncation at 0-year. It means that the aging-induced delay can be compensated by 1-bit truncation, and it is enough to prevent from MSP errors. On the other hand, 2-bit truncation at 1-year rather degrades the MSE due to the excessive truncation. At 10-year, however, it is required to configure the 2-bit truncation because it is not enough to reduce the critical path delay by 1-bit truncation. This configuration absolutely improves MSE, even though the error rate increases. Note that this error rate mostly comes from the LSBs, so the impact on the quality is not large. Note also that MSE is closely related to the PSNR metric widely used to evaluate the image quality. This experimental results demonstrate that the proposed adders compensate the aging-induced delay with approximation gracefully and improve the quality on image processing.

Table 11.2 shows the comparison of 32-bit adders. Since MSP errors in this case are much more critical than the errors of 16- bit adder, our proposed adder and scheme show the better MSE improvement in this experiment. In Chapter 11.3, we use 32-bit adder as a component in order to match the required bit-width of the system.

Table 11.2 Comparison of 32-bit ripple carry adder and proposed adder

32-bit Ripple Carry Adder							
Aging Time	**0-year**	**1-year**			**10-year**		
Power (uW)	27.09	27.21			27.44		
Area	153.22						
Critical Path Delay (ns)	1.966	2.065			2.129		
Error Rate	0.00%	1.54%			2.89%		
MSE	0.00	4.85E+16			8.18E+16		
32-bit Proposed Adder - Masking							
Aging Time	**0-year**	**1-year**			**10-year**		
# of Truncated LSBs	**-**	**0**	**1**	**2**	**0**	**2**	**4**
Power (uW)	27.22	28.30			28.59		
Area	161.73						
Critical Path Delay (ns)	1.993	2.108	2.044	1.980	2.172	2.040	1.865
Error Rate	0.00%	1.54%	75.18%	75.13%	2.89%	75.53%	74.87%
MSE	0.00	4.85E+16	3.85E+16	5.98	8.18E+16	7.68E+16	95.67
32-bit Proposed Adder - Cutting							
Aging Time	**0-year**	**1-year**			**10-year**		
# of Truncated LSBs	**-**	**0**	**1**	**2**	**0**	**2**	**4**
Power (uW)	26.92	27.52			27.86		
Area	157.47						
Critical Path Delay (ns)	2.032	2.134	2.053	1.972	2.198	2.033	1.865
Error Rate	0.00%	1.53%	50.55%	49.93%	3.02%	51.18%	49.87%
MSE	0.00	4.71E+16	5.20E+16	7.99	8.31E+16	1.14E+17	127.67

11.3 Microarchitecture Level

To evaluate the proposed system and scheme at the microarchitecture level, we use DCT/IDCT codec blocks which encode and then decode the image files. With this experiment, we show the feasibility of our proposed system in a real image processing application. We replace the adders of DCT/IDCT with the 32-bit proposed adders. We perform this experiment with two cutting type adders;

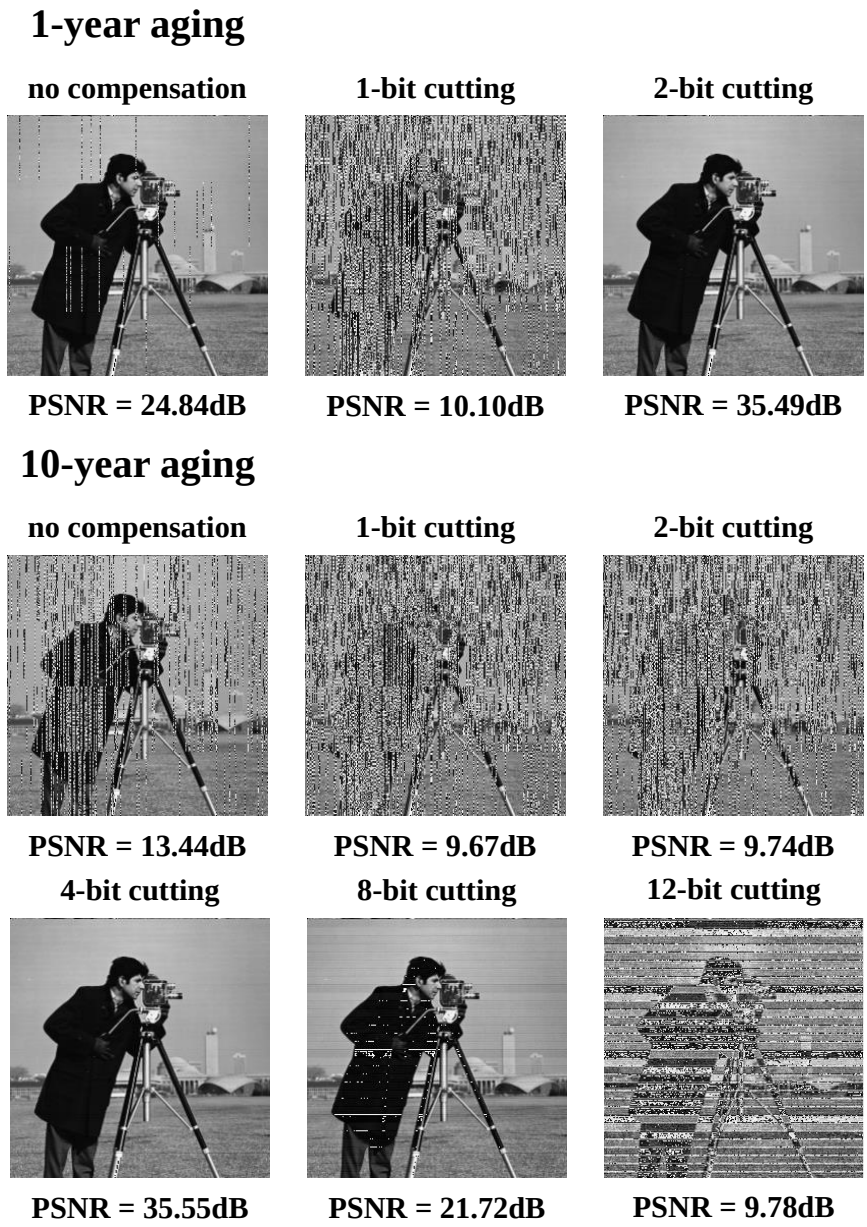


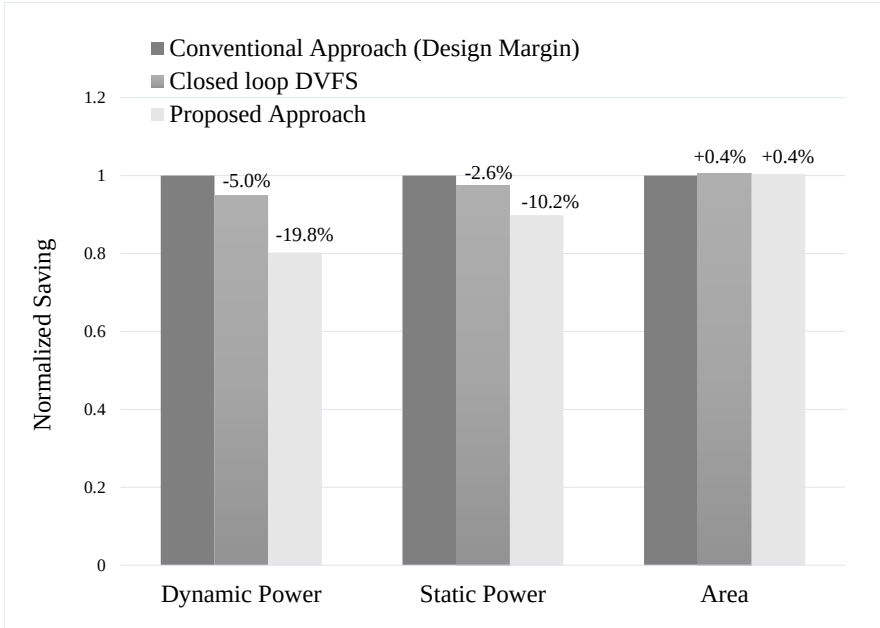
Figure 11.1 Evaluation of aging compensation of proposed system with approximation in image processing application.

One enables truncating 1, 2, 3, and 4-bits and the other enables truncating 4, 8, 12, and 16-bits. We perform this experiment since 4-bit truncation might not be enough in this microarchitecture level system because there is some extra logic in the critical path in addition to the adder. We also want to analyze the impact of the excessive truncation on image quality at this microarchitecture level.

Figure 11.1 shows the output image of the DCT/IDCT codec blocks without the reliability design margin. PSNR is degraded down to 24.84dB after 1-year aging and 13.44dB after 10-year aging, respectively. It means that reliability design margin is essentially required even in error resilient applications such as image processing. We also observe that most of the aging incurs within initial 1-2 year of the projected lifetime.

In the proposed system without the design margin, the aging-induced delay is compensated gracefully. In case of 1-year aging, the proposed system cannot compensate the errors by only 1-bit truncation. Rather, it makes PSNR degrade to 10.10dB due to the LSB truncation. It means that the critical path delay after 1-bit truncation is still longer than one clock period. It can recover from the image quality degradation by 2-bit truncation as shown in the figure, implying that the proposed system can compensate the aging-induced delay very well. In case of 10-year aging, PSNR is significantly degraded to 13.44dB without the design margin. Even the 2-bit truncation makes the image quality worse than the quality of no compensation. However, it can also recover from the degraded quality to the quality comparable to the original image by 4-bit truncation. The proposed system dynamically

Table 11.3 Power and area comparison of conventional and proposed approaches



compensates the aging-induced delay with approximation at run-time, based on the delay increase information from the monitoring circuit (e.g., 2-bit truncation after 1-year and 4-bit truncation after 10-year). However, the excessive truncations by inaccurate delay information make the quality worse and worse as shown in the figure for the cases of 8-bit and 12-bit truncations, where PSNR is degraded to 21.72dB and 9.78dB, respectively. That is, the appropriate truncation of LSBs is very important to get an optimal result as the delay increases by aging, which justifies the use of an accurate monitoring circuit.

In Table 11.3, we summarize the power and area comparison of three approaches, conventional one with reliability design margin, closed-loop DVFS, and

the proposed one. First, we get a frequency level of the conventional approach under typical voltage at 0-year. Then we calculate the required design margin in voltage, which is an increment of supply voltage to avoid timing violation errors at 10-year. With this design margin, the conventional approach consumes 19.8% and 10.2% more dynamic and static power, respectively, compared to the proposed approach which does not required the design margin. And the proposed approach also shows lower power consumptions than closed-loop DVFS. As the three approaches operate at the same frequency level, the energy consumption saving can be considered as power consumption saving. In case of area, closed-loop DVFS and the proposed approach incur 0.4% overhead due to the monitoring circuit and control unit (actually, closed-loop DVFS incurs more overhead for voltage scaling system. But, it is not considered in this comparison result). However, the area overhead is negligible, compared to the whole DCT/IDCT area. That is, in this system, the proposed approach achieves a large power/energy reduction with a small overhead under the acceptable image quality degradations.

Chapter 12

Conclusion

We propose a novel aging compensation system and scheme with approximation, which consists of a monitoring circuit, a control unit, and configurable adders. It dynamically reduces the precision of the adders by monitoring the aging-induced delay at run-time. Our proposed adders mitigate numerically significant errors to less significant errors. That is why our proposed system avoids significant image quality degradation without costly reliability design margin in an image processing application. In addition, our proposed design methodology can be easily integrated into an existing conventional design flow and can be applied to any other circuits as well as ripple carry adders, provided that they can be configured to trade off the precision with speed. In an advanced process technology, the effects of reliability on a circuit delay can be much more serious and thus such a dynamical compensation

method for mitigating the reliability problem becomes more important. The proposed system can effectively resolve the problem while maintaining low cost/power and acceptable performance degradation.

Bibliography

[1] G. F. Taylor, "Where are we going? Product scaling in the system on chip era," in *IEEE International Electron Devices Meeting (IEDM)*, pp. 17.1.1-17.1.3, 2013.

[2] K. J. Kuhn, "Reducing variation in advanced logic technologies: Approaches to process and design for manufacturability of nanoscale CMOS," in *IEEE International Electron Devices Meeting (IEDM)*, pp. 471-474, 2007.

[3] K. J. Kuhn *et al.*, "Process technology variation," *IEEE Transactions on Electron Devices*, vol. 58, no. 8, pp. 2197-2208, 2011.

[4] B. Vaidyanathan and A. S. Oates, "Technology scaling effect on the relative impact of NBTI and process variation on the reliability of digital circuits," *IEEE Transactions on Device and Materials Reliability*, vol. 12, no. 2, pp. 428-436, 2012.

[5] S. Jain *et al.*, "A 280mV-to-1.2V wide-operating-range IA-32 processor in 32nm CMOS," in *IEEE International Solid-State Circuits Conference (ISSCC)*, pp. 66-68, 2012.

[6] T. Tekeste, A. Shabra, D. Boning, and I. Elfadel, "Variability analysis of a 28nm near-threshold synchronous voltage converter," in *IEEE International Conference on Electronics, Circuits, and Systems (ICECS)*, pp. 723-726, 2013.

[7] U. R. Karpuzcu, N. S. Kim, and J. Torrellas, "Coping with parametric variation at near-threshold voltages," *IEEE Micro*, vol. 33, no. 4, pp. 6-14, 2013.

[8] S. Seo *et al.*, "Process variation in near-threshold wide SIMD architectures," in *ACM/EDAC/IEEE Design Automation Conference (DAC)*, pp. 980-987, 2012.

[9] M. Seok, G. Chen, S. Hanson, M. Wieckowski, D. Blaauw, and D. Sylvester, "CAS-FEST 2010: Mitigating variability in near-threshold computing," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 1, no. 1, pp. 42-49, 2011.

[10] M. Bhushan, A. Gattiker, M. B. Ketchen, and K. K. Das, "Ring oscillators for CMOS process tuning and variability control," *IEEE Transactions on Semiconductor Manufacturing*, vol. 19, no. 1, pp. 10-18, 2006.

[11] L. T. N. Wang, N. Xu, S. O. Toh, A. R. Neureuther, T. J. K. Liu, and B. Nikolic, "Parameter-specific ring oscillator for process monitoring at the 45nm node," in *IEEE Custom Integrated Circuits Conference (CICC)*, pp. 1-4, 2010.

[12] K. Kang, S. P. Park, K. Kim, and K. Roy, "On-chip variability sensor using phase-locked loop for detecting and correcting parametric timing failures," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 18, no. 2, pp. 270-280, 2010.

[13] T. B. Chan, A. Pant, L. Cheng, and P. Gupta, "Design dependent process monitoring for back-end manufacturing cost reduction," in *IEEE/ACM*

International Conference on Computer-Aided Design (ICCAD), pp. 116-122, 2010.

[14] T. B. Chan, P. Gupta, A. B. Kahng, and L. Lai, "DDRO: A novel performance monitoring methodology based on design-dependent ring oscillators," in *IEEE International Symposium on Quality Electronic Design (ISQED)*, pp. 633-640, 2012.

[15] K. Seno, S. Takahiro, and T. Kondo, "Semiconductor device replica circuit for monitoring critical path and construction method of the same," U.S. Patent No. 6,414,527, Jul. 2, 2002.

[16] D. Fick *et al.*, "In situ delay-slack monitor for high-performance processors using an all-digital self-calibrating 5ps resolution time-to-digital converter," in *IEEE International Solid-State Circuits Conference (ISSCC)*, pp. 188-189, 2010.

[17] X. Wang, M. Tehranipoor, and R. Datta, "Path-RO: A novel on-chip critical path delay measurement under process variations," in *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pp. 640-646, 2008.

[18] A. Drake *et al.*, "A distributed critical-path timing monitor for a 65nm high-performance microprocessor," in *IEEE International Solid-State Circuits Conference (ISSCC)*, pp. 398-399, 2007.

[19] A. J. Drake, R. M. Senger, H. Singh, G. D. Carpenter, and N. K. James, "Dynamic measurement of critical-path timing," in *IEEE International*

Conference on Integrated Circuit Design and Technology and Tutorial (ICICDT), pp. 249-252, 2008.

[20] A. J. Drake, X. Yuan, P. Owczarczyk, and M. Tiner, “Accurate model-to-hardware simulation methodology for designing critical path monitors over a wide voltage range,” in *ACM/EDAC/IEEE Design Automation Conference (DAC)*, 2013.

[21] J. Tschanz, K. Bowman, S. Walstra, M. Agostinelli, T. Karnik, and V. De, "Tunable replica circuits and adaptive voltage-frequency techniques for dynamic voltage, temperature, and aging variation tolerance," in *IEEE Symposium on VLSI Circuits*, pp. 112-113, 2009.

[22] K. A. Bowman *et al.*, "All-digital circuit-level dynamic variation monitor for silicon debug and adaptive clock control," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 58, no. 9, pp. 2017-2025, 2011.

[23] X. Wang, M. Tehranipoor, S. George, D. Tran, and L. Winemberg, "Design and analysis of a delay sensor applicable to process/environmental variations and aging measurements," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 20, no. 8, pp. 1405-1418, 2012.

[24] S. Wang, J. Chen, and M. Tehranipoor, "Representative critical reliability paths for low-cost and accurate on-chip aging evaluation," in *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pp. 736-741, 2012.

[25] S. Novak *et al.*, "Transistor aging and reliability in 14nm tri-gate technology," in *IEEE International Reliability Physics Symposium (IRPS)*, pp. 2F.2.1-2F.2.5, 2015.

[26] J. Keane and C. H. Kim, “Transistor aging," *IEEE Spectrum*, 2011.

[27] V. Gupta, D. Mohapatra, S. P. Park, A. Raghunathan, and K. Roy, "IMPACT: IMPrecise adders for low-power approximate computing," in *IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, pp. 409-414, 2011.

[28] J. Han and M. Orshansky, "Approximate computing: An emerging paradigm for energy-efficient design," in *IEEE European Test Symposium (ETS)*, pp. 1-6, 2013.

[29] H. Amrouch, B. Khaleghi, A. Gerstlauer, and J. Henkel, "Reliability-aware design to suppress aging," in *ACM/EDAC/IEEE Design Automation Conference (DAC)*, pp. 1-6, 2016. “Degradation-Aware Cell Libraries, V1.0,” http://ces.itec.kit.edu/dependable-hardware.php

[30] C. Prasad *et al.*, "Transistor reliability characterization and comparisons for a 14nm tri-gate technology optimized for system-on-chip and foundry platforms," in *IEEE International Reliability Physics Symposium (IRPS)*, pp. 4B-5-1-4B-5-8, 2016.

[31] H. Amrouch, B. Khaleghi, A. Gerstlauer, and J. Henkel, "Towards aging-induced approximations," in *ACM/EDAC/IEEE Design Automation Conference (DAC)*, pp. 1-6, 2017.

[32] S. Ghosh and K. Roy, "Parameter variation tolerance and error resiliency: New design paradigm for the nanoscale era," *Proceedings of the IEEE*, vol. 98, no. 10, pp. 1718-1751, 2010.

[33] V. Huard, F. Cacho, A. Benhassain, and C. Parthasarathy, "Aging-aware adaptive voltage scaling of product blocks in 28nm nodes," in *IEEE International Reliability Physics Symposium (IRPS)*, pp. 7C-2-1-7C-2-7, 2016.

[34] H. Mostafa, M. Anis, and M. Elmasry, "NBTI and process variations compensation circuits using adaptive body bias," *IEEE Transactions on Semiconductor Manufacturing*, vol. 25, no. 3, pp. 460-467, 2012.

[35] M. Cho *et al.*, "Postsilicon voltage guard-band reduction in a 22nm graphics execution core using adaptive voltage scaling and dynamic power gating," *IEEE Journal of Solid-State Circuits*, vol. 52, no. 1, pp. 50-63, 2017.

[36] J. Li and M. Seok, "Robust and in-situ self-testing technique for monitoring device aging effects in pipeline circuits," in *ACM/EDAC/IEEE Design Automation Conference (DAC)*, pp. 1-6, 2014.

[37] N. Zhu, W. L. Goh, W. Zhang, K. S. Yeo, and Z. H. Kong, "Design of low-power high-speed truncation-error-tolerant adder and its application in digital signal processing," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 18, no. 8, pp. 1225-1229, 2010.

[38] N. Zhu, W. L. Goh, and K. S. Yeo, "An enhanced low-power high-speed adder for error-tolerant application," in *IEEE International Symposium on Integrated Circuits (ISIC)*, pp. 69-72, 2009.

[39] N. Zhu, W. L. Goh, G. Wang, and K. S. Yeo, "Enhanced low-power high-speed adder for error-tolerant application," in *IEEE International SoC Design Conference (ISOCC)*, pp. 323-327, 2010.

[40] D. Shin and S. K. Gupta, "A re-design technique for datapath modules in error tolerant applications," in *IEEE Asian Test Symposium (ATS)*, pp. 431-437, 2008.

[41] A. B. Kahng and S. Kang, "Accuracy-configurable adder for approximate arithmetic designs," in *ACM/EDAC/IEEE Design Automation Conference (DAC)*, pp. 820-825, 2012.

[42] R. Ye, T. Wang, F. Yuan, R. Kumar, and Q. Xu, "On reconfiguration-oriented approximate adder design and its application," in *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pp. 48-54, 2013.

[43] M. Shafique, W. Ahmad, R. Hafiz, and J. Henkel, "A low latency generic accuracy configurable adder," in *ACM/EDAC/IEEE Design Automation Conference (DAC)*, pp. 1-6, 2015.

[44] I-M. Pao and M.-T. Sun, "Modeling DCT coefficients for fast video encoding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 9, no. 4, pp. 608-616, 1999.

[45] H. Amrouch, S. Mishra, V. V. Santen, S. Mahapatra, and J. Henkel, "Impact of BTI on dynamic and static power: From the physical to circuit level," in *IEEE International Reliability Physics Symposium (IRPS)*, pp. CR-3.1-CR-3.6, 2017.

국문초록

반도체 공정 기술이 지속적으로 미세화됨에 따라, 제조 및 동작 환경에 따른 회로의 성능 변동이 점점 더 심각 해지고 있다. 이러한 성능 변동은 예측하기가 매우 어렵기 때문에 추가적인 설계 마진을 필요로 하는데, 이는 칩의 면적 및 전력 소비를 증가시킨다. 이 문제를 해결할 수 있는 가장 이상적인 방법은 실시간으로 성능 변동을 측정하고, 피드백 루프를 통해 적절한 전압을 공급하는 것이다. 하지만, 이 기법의 가장 중요한 점은 모니터링 회로와 대상 블록과의 성능 상관 관계 불일치이다. 큰 불일치는 오히려 피드백 루프 방법의 장점을 잃을 수 있다. 본 논문에서는 광범위한 동작 전압을 고려한 여러 개의 일반 모니터를 갖춘 새로운 지연 모니터링 시스템을 제안한다. 이 시스템은 기존 모니터링 방법에 비해 모니터링 회로와 대상 블록간의 성능 상관 관계가 더 좋다. 14 나노미터 FinFET 프로세서 코어에 적용하여 오류를 최대 91 %까지 줄여 설계 마진을 줄이고 이를 통해 전력 소비를 감소시키고 저비용 설계를 달성할 수 있다.

또한, 본 논문은 칩의 노후화 보상 방법을 다룬다. 일반적으로 칩의 노후화로 인한 신뢰성 저하는 설계 마진을 사용하여 해결한다. 그러나 이

방법은 효율적이지 못하며, 설계 과정에서 노후화 영향을 정확하게 예측 해야 하므로 많은 어려움이 있다. 따라서, 저전력 설계를 위해서는 실시간으로 칩의 노후화로 인한 성능 저하를 측정하고 이를 적절하게 보상해 줘야 한다. 본 논문에서는 신뢰도 설계 마진이나 전압의 증가 없이 칩 노후화로 인한 성능 저하를 근사계산으로 보상하는 새로운 설계 방법론을 제안한다. 이 방법은 실시간 성능 모니터링 시스템을 기반으로 하며, 최종 목표는 설계 마진을 없애고, 칩의 노후화로 인한 성능저하를 정확도를 낮추는 근사계산으로 보상하여 칩의 전력소비를 감소시키는 것이다. 제안하는 방법을 구성 요소 레벨과 마이크로 아키텍처 시스템 레벨에서 평가한 결과, 구성 요소 레벨에서는 오류의 평균제곱값에서 상당한 개선을 보여주고, 시스템 레벨에서는 큰 품질 저하 없이 노후화로 인한 성능 저하를 보상한다. 이 방법을 통해 0.4%의 면적 증가로 전력 소비를 19.8 % 감소시켰다.

주 요 어: 성능 모니터, 적응 제어, 설계 마진, 저전력, 칩 신뢰성, 근사 계산

학 번: 2014-30306