MTTF-AWARE DESIGN METHODOLOGY FOR ADAPTIVE VOLTAGE SCALING

Masanori Hashimoto[1]*, *Yutaka Masuda*[1]

[1] Dept. Information Systems Engineering, Osaka University, Suita, Japan

*Corresponding Author's Email: hasimoto@osaka-u.ac.jp

ABSTRACT

Adaptive voltage scaling (AVS) is a promising approach to overcome manufacturing variability, dynamic environmental fluctuation, and aging. This paper focuses on design of AVS circuits. For pursuing efficient AVS, this work presents a design methodology that optimizes both voltage-scaled circuit and voltage control logic achieving practical long mean time to failure (MTTF), e.g., years. Evaluation results show the proposed AVS achieves 20.8% voltage reduction while satisfying target MTTF.

INTRODUCTION

Aggressive device miniaturization due to technology scaling has been improving the device performance. On the other hand, circuits have become sensitive to static manufacturing variability and dynamic environmental fluctuation. These static and temporal variations directly lead to circuit reliability degradation. The most effective tuning knob for post-silicon compensation is supply voltage control, and adaptive voltage scaling (AVS) is intensively studied [1][2]. AVS is expected to minimize process, voltage, temperature, and aging (PVTA) margin of each chip and allocate only a small margin for the entire lifetime as shown in Figure 1. The excessive conventional PVTA margins existing in most of the chips can be exploited as the source for power reduction. Conventional works [1][2] focus on where to insert sensors and how to control supply voltage and discuss the design methodology of voltage control system including sensing circuit.

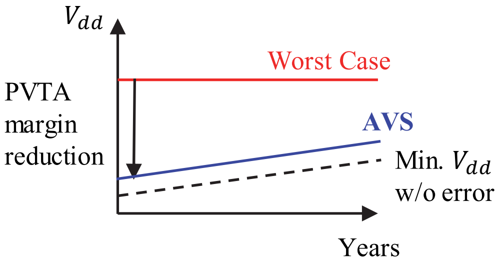

Figure 1 : Supply voltage of AVS and conventional worst case (WC) in device lifetime.

On the other hand, for implementing AVS systems that fully exploit run-time adaptation and eliminate the redundant margin, we have found that we should pay attention to the main logic circuit under AVS in addition to the sensing circuit. In the conventional VLSI design flow, there are many critical paths since the timing slack is exploited for power and area reduction. However, we observe that inherent critical paths whose path delays cannot be reduced at all are limited. This observation suggests that adaptive slack assignment (ASA) to the main logic circuit under AVS, which allocates larger slack to highly active paths, could improve the efficacy of the AVS and enable further supply voltage reduction.

This work focuses on the error prediction based AVS (EP-AVS) and presents a design methodology for EP-AVS circuits [3]. The proposed methodology optimizes both the main logic under AVS and sensing circuit. In the main logic design, we perform mean time to failure (MTTF) aware ASA that utilizes critical path isolation (CPI) [4] and estimates MTTF of AVS circuits with a stochastic framework [5]. As for the sensing circuit design, we propose a novel sensor insertion method that minimizes the sum of gate-wise timing failure probabilities, where the timing failure probability is the joint probability of activation and timing violation. By exploiting the information on the paths with higher timing failure probability, the proposed sensor insertion makes EP-AVS efficiently monitor the timing-critical and highly-active FFs.

The rest of this paper is organized as follows. First, this paper proposes a design methodology that optimizes both the main logic under AVS and sensing circuit. Then, we demonstrate the supply voltage reduction and speed-up thanks to the proposed EP-AVS. Lastly, concluding remarks are given.

PROPOSED DESIGN METHODOLOGY

This section first explains the assumed EP-AVS and the overview of the proposed methodology. Then, The ASA and the sensor insertion are presented in series.

Assumed EP-AVS

Figure 2 illustrates an EP-AVS circuit assumed in this paper. The EP-AVS circuit is composed of the main circuit, timing error predictive flip-flop (TEP-FF) [6] and voltage control unit. The TEP-FF consists of a flip-flop, delay buffers, and a comparator (XOR gate), and works with the main FF. When the timing margin is gradually decreasing, a timing error occurs at the TEP-FF before the main FF captures a wrong value due to the delay buffer, which enables us to know that the timing margin of the main FF is not large enough. An error prediction signal is generated to predict the timing errors, and this signal is monitored during a specified period. Note that timing errors are predicted, not detected, which is a distinct difference from Razor [1]. Once an error prediction signal is observed, the higher supply voltage is given to reduce circuit delay. Note that clock frequency is fixed throughout this paper. If no error prediction signals are observed during the monitoring period, the circuit is slowed down for power reduction. This proactive AVS is expected to overcome the variation of the timing margin which is different in every chip and varies depending on operating condition and aging.

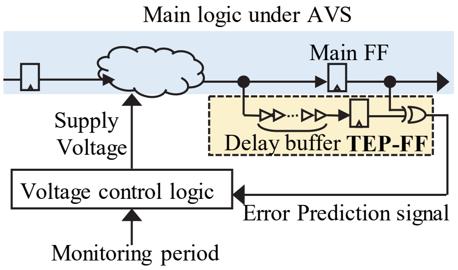

Figure 2 : Assumed EP-AVS.

Overview and Problem Definition

The proposed design methodology for EP-AVS consists of the ASA for the main logic under AVS followed by the insertion of TEP-FFs. Figure 3 illustrates the concept of ASA. In conventional design flow, cell instances that are included in non-critical paths are replaced with smaller cells and higher-Vth cells for reducing power dissipation and area. Consequently, this replacement decreases timing margin of many paths and may deteriorate MTTF. On the other hand, the ASA increases timing slacks of non-intrinsic critical paths as shown in Figure 3. Meanwhile, the path-based slack assignment is not efficient since the number of paths in a circuit is huge. Therefore, this work utilizes FF-based CPI proposed in [4] to adjust setup slacks of FFs. For each FF, we assign an individual target slack value. After the CPI, the paths ending at the FFs whose slack values increased are less likely to fail even when the gate delays in the paths vary, which contributes to MTTF extension.

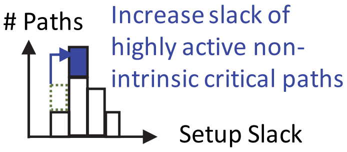
# Paths
Increase slack of highly active non-intrinsic critical paths
Setup Slack

Figure 3 : ASA for main logic optimization.

Here, it should be noted that CPI increases area and power since the intentional increase in slack was originally exploited for area and power reduction in conventional design. In this sense, we need to identify FFs that have less impact on area and power yet contribute to remarkable MTTF extension. Based on the discussion above, we formulate the design optimization of EP-AVS including CPI-based ASA and TEP-FF insertion.

Objective : To minimize V_{dd}
Variables : B_{TEP_i} $(1 \leq i \leq N_{FF})$, B_{CPI_i} $(1 \leq i \leq N_{FF})$
Constraint #1 : $MTTF \leq MTTF_{const}$
Constraint #2 : N_{TEP} $(= \sum_{i=1}^{N_{FF}} B_{TEP_i}) = N_{TEP}^{max}$
Constraint #3 : N_{CPI} $(= \sum_{i=1}^{N_{FF}} B_{CPI_i}) = N_{CPI}^{max}$

The objective of this problem is to minimize V_{dd} aiming at power minimization. The variables for optimization are B_{TEP_i} and B_{CPI_i}. B_{TEP_i} is a binary variable, and it becomes 1 when i-th FF is replaced to TEP-FF. B_{CPI_i} is also a binary variable, and it is 1 when CPI is applied to i-th FF. The primary constraint is MTTF, and the lower bound of MTTF ($MTTF_{const}$) is given as a constraint. The second constraint gives the upper bound of the number of TEP-FFs (N_{TEP}^{max}), and this limits the area increase due to TEP-FF insertion. Similarly, the upper bound of the number of FFs to which CPI is applied (N_{CPI}^{max}) is also given as a constraint to limit the area increase originating from CPI. The proposed design methodology solves this problem with a two-stage procedure. The first stage designs the main logic under AVS using CPI [4], i.e., determines B_{CPI_i}, and the second stage performs TEP-FF insertion, i.e., determine B_{TEP_i}. The following subsections explain these two stages.

First Stage: CPI-based ASA in Main Logic

The CPI is performed referring to [4]. The CPI method focuses on gate-wise failure probability which is a metric that expresses the contribution to the timing failure probabilities at the downstream FFs. The detailed computation will be explained in the next subsection. Then, for maximally reducing the sum of gate-wise failure probabilities, this method selects target FFs by solving a covering problem of instances weighted with the failure probability as an integer linear programming (ILP) problem and thus determines B_{CPI_i}. Once B_{CPI_i} is determined, the FF-based CPI proceeds to the following two steps; (1) increase setup time of the i-th FF by $\Delta Setup_i$ artificially and re-synthesize the design as an engineering change order (ECO) process, and (2) restore the setup time for the successive analysis process. With this CPI, we enforce the paths ending at the target FF to have the slack of more than $\Delta Setup_i$. Referring to [4], $\Delta Setup_i$ is set to the upper bound value that can satisfy the setup constraint after ECO for simplicity.

Second Stage: Sensing Circuit Insertion

For making EP-AVS work well, TEP-FFs need to output the error prediction signals frequently to nicely adjust the supply voltage, and hence it is desirable that inserted TEP-FFs are highly activated. Also, FFs with small slacks need fewer delay buffers in TEP-FFs. Here, FFs having higher timing failure probability satisfies both the desirable properties above. Therefore, we propose a novel TEP-FF insertion method that minimizes the sum of gate-wise timing failure probabilities. Our insertion method consists of the following two steps; (1) calculate timing failure probabilities, and (2) find out a set of FFs that maximally reduce the sum of gate-wise failure probabilities by solving the instance covering problem as an ILP problem.

In the first step, the proposed method calculates timing failure probability of FFs, $P_{FF_{i\_}fail}$. Remind that timing failure probability is the joint probability of timing violation and activation. In this work, we calculate the timing violation probability by statistical static timing analysis (SSTA) and derive the activation probability of each path by associating the signal transition time in logic simulation with the path delay in STA. Then, we obtain $P_{FF_{i\_}fail}$ by multiplying the timing violation probability and the activation probability. Next, we compute the gate-wise failure probabilities, i.e., $P_{insk_{k\_}fail}$, as follows.

$$P_{insk_{k\_}fail} = \max\{\frac{P_{FF_{i\_}fail}}{\sum_{\mathrm{k=1}}^{N_{inst}} B_{FF_{i\_}inst_k}}\} \ (1 \le i \le N_{FF}). \quad (1)$$

In Equation (1), N_{inst} is the number of instances in the circuit. $B_{FF_{i\_}inst_k}$ is a binary valuable which is determined by the circuit topology, and it becomes 1 when *k*-th instance is included in the paths ending at *i*-th FF. $\sum_{\mathrm{i=1}}^{N_{inst}} B_{FF_{i\_}inst_k}$ is the total number of instances included in the fan-in cone of *i*-th FF. The above equation assumes that each instance included in the fan-in cone of *i*-th FF has the same contribution to the timing error at the FF, and hence the $P_{FF_{i\_}fail}$ is divided by $\sum_{\mathrm{k=1}}^{N_{inst}} B_{FF_{i\_}inst_k}$. An instance can be included in the fan-in cones of multiple FFs. For coping with this, the max operation is performed.

In the second step, we select a set of FFs that maximize the sum of gate-wise failure probabilities. We formulate this FF selection problem as an ILP problem to derive the exact solution. Our ILP formulation is as follows:

Objective : To maximize $\sum_{\mathrm{k=1}}^{N_{inst}}(P_{\mathrm{inst}_{k\_}fail} \times B_{inst_k})$
Variables : B_{TEP_i} $(1 \le i \le N_{FF})$
Constraint #1 : $0 \le B_{inst_k} \le 1$ $(1 \le k \le N_{inst})$
Constraint #2 : $0 \le B_{TEP_i} \le 1$ $(1 \le i \le N_{FF})$
Constraint #3 : $\sum_{\mathrm{i=1}}^{N_{FF}} B_{TEP_i} \le N_{\mathrm{TEP}}$
Constraint #4 : $B_{inst_k} \le \sum_{\mathrm{i=1}}^{N_{FF}}(B_{TEP_i} \times B_{FF_{i\_}inst_k})$

The objective of this ILP problem is to maximize the sum of $(P_{\mathrm{inst}_{k\_}fail} \times B_{inst_k})$, where $P_{\mathrm{inst}_{k\_}fail}$ is the gate-wise failure probability. B_{inst_k} is a binary variable, and it becomes 1 when *k*-th instance is included in paths ending at the target FFs for TEP-FF insertion. Therefore, the sum of $P_{\mathrm{inst}_{k\_}fail} \times B_{inst_k}$ represents the gate-wise failure probability reduction. In this problem, we assign binary variables B_{TEP_i}, where B_{TEP_i} becomes 1 when *i*-th FF is selected as target FFs for TEP-FF insertion.

The first and second constraints are given to restrict B_{inst_k} and B_{TEP_i} to binary numbers. The third constraint means that the number of target FFs for TEP-FF insertion should be equal or less than N_{TEP}. The fourth constraint is a key constraint that defines the relation between B_{inst_k} and B_{TEP_i}. B_{inst_k} becomes 0 only when the product of B_{TEP_i} and $B_{FF_{i\_}inst_k}$ is 0 for all the FFs. On the other hand, if *k*-th instance is included in the paths ending at the target FFs, at least one of the products of $B_{TEP_i}$and $B_{FF_{i\_}inst_k}$ become 1. In this case, B_{inst_k} can be 1. In our ILP formulation, we are maximizing the sum of $(P_{\mathrm{inst}_{k\_}fail} \times B_{inst_k})$ and hence B_{inst_k} is necessarily assigned to be 1.

EXPERIMENTAL EVALUATION

This section experimentally evaluates the performance improvement thanks to the proposed EP-AVS.

Experimental Setup

In this work, we used the advanced encryption standard (AES) circuit and OR1200 OpenRISC processor, which is a 32-bit RISC microprocessor with five pipeline stages, as target circuits. These two circuits were designed by a commercial logic synthesizer with a 45nm Nangate standard cell library, where the synthesized OpenRISC has 2500 FFs and AES has 530 FFs. Also, standard cell memories [7] were used as SRAMs in OpenRISC processor. We used Gurobi Optimizer 7.0 to solve the ILP problem defined in the previous section. The solver was executed on a 2.4 GHz Xeon CPU machine under the Red Hat Enterprise Linux 6 operating system with 1024 GB memory. For calculating meaningful MTTF, practical delay variations should be considered. Our evaluation took into account the following variations; (1) Dynamic supply noise, which is assumed to temporally fluctuate between -50 mV and 50 mV by 10mV. (2) Intra-die random variation and inter-die variation, which consist of NMOS and PMOS threshold voltage variation of σ = 30 mV and gate length variation of σ = 1 nm. (3) NBTI aging, whose model was obtained by fitting the measured data in [8] to the trapping/de-trapping model [9]. Six degradation states of 0 mV, 0.5 mV, 1 mV, 5 mV, 10 mV and 15 mV are prepared.

As for workload in OpenRISC, we selected three benchmark programs (CRC32, SHA1, and Dijkstra) from MIBenchmark [9]. In AES, 1,000 random test patterns were used. We set MTTF of 1.0×10^{17} cycles, i.e., 3.3 years in Open-RISC and 1.6 years in AES, as $MTTF_{const}$. With this setup, we performed CPI-based ASA to both AES and OpenRISC. The constraint of ASA area overhead is set to 6.0% for AES and 1.0% for OpenRISC. Next, we inserted several TEP-FFs to the voltage-scaled circuits. The constraint of area overhead for TEP-FF is set to 1.0%

for both AES and OpenRISC, i.e., N_{TEP}^{max} is set to 18 in AES and 13 in OpenRISC, respectively. When inserting TEP-FF, we need to determine the number of delay buffers for each TEP-FF. In this work, we inserted the delay buffers whose delay were comparable to the delay variation caused by 100 mV supply noise. This determination of the number of delay buffers includes room for improvement.

MTTF and average supply voltage under PVTA variation are evaluated by the stochastic MTTF estimation framework [5]. In our experiment, the monitor period for EP-AVS was set to 10^6 cycles. We prepared nine supply voltages from 1.20 V to 0.80 V with 50 mV interval.

Evaluation Results

Figure 4 shows the trade-off curves between the minimum average supply voltage and the clock cycle under the MTTF constraint of 10^{17} cycles, where (a) in OpenRISC and (b) in AES, respectively. The black square plots represent the conventional WC design with guard-banding for PVTA variation. The yellow circular and blue triangular plots correspond to the conventional EP-AVS which optimizes only the sensing circuit, and the proposed EP-AVS which optimizes both the main logic under AVS and sensing circuit, respectively. First, we compare the black square and blue triangular plots for clarifying the overall performance improvement thanks to the proposed EP-AVS. Figure 4 shows that the proposed EP-AVS reduces average supply voltage and clock cycle time while keeping the target MTTF. For example, in Figure 4(a), at a clock period of 1040 ps, the proposed EP-AVS achieved the target MTTF with an average supply voltage of 0.95 V, whereas the conventional WC design required 1.20 V operation. In other words, EP-AVS achieved 20.8% V_{dd} reduction from 1.20 V to 0.95 V. Similarly, in Figure 4(b), at a clock period of 390 ps, the proposed EP-AVS achieved 7.5% V_{dd} reduction from 1.20 V to 1.11 V. As for clock period reduction, the proposed EP-AVS achieved 34.0% speed-up at 0.80 V in OpenRISC (Figure 4(a)) and 19.5% speed-up at 0.80 V in AES (Figure 4(b)), respectively. We experimentally confirmed that the proposed EP-AVS made the significant performance improvement at the cost of 7.0% area increase in. AES and 1.4% in OpenRISC.

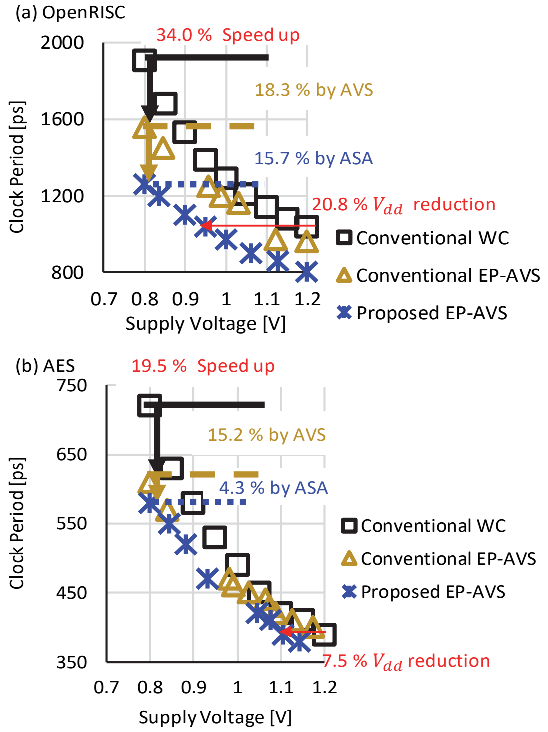

Figure 4 : Trade-off curves between supply voltage and clock period (a) OpenRISC, (b) AES.

Next, we compared the conventional EP-AVS and proposed EP-AVS. Figure 4 shows that the proposed EP-AVS further improves performance from the conventional EP-AVS. For example, at 0.80 V, the proposed EP-AVS achieved 15.7% speed-up from 1560 ps to 1260 ps in OpenRISC and 4.3% speed-up from 610 ps to 580 ps in AES. This reveals that the ASA for the main logic works well regarding speed-up and V_{dd} reduction and the simultaneous optimization of the main logic under AVS and the sensing circuit enhances the efficacy of EP-AVS.

CONCLUSION

This paper focused on EP-AVS and proposed a design methodology for EP-AVS circuits. The proposed methodology optimizes both the main logic under AVS and sensing circuits. The quantitative MTTF and supply voltage evaluation results showed that the proposed EP-AVS achieved 20.8% voltage reduction while satisfying target MTTF. One of our future works include power-oriented design optimization of EP-AVS.

ACKNOWLEDGEMENTS

This work is partly supported by STARC, Socionext, and ICOM Foundation, Japan.

REFERENCES

[1] S. Das, *IEEE Journal Solid-State Circuits*, vol. 41, no. 4, pp. 792–804, 2006.
[2] K. A. Bowman, *IEEE Journal Solid-State Circuits*, vol.46, no. 1, pp. 194–208, 2011.
[3] Y. Masuda, in *Proc. ASP-DAC*, 2018.
[4] Y. Masuda, in *Proc. ICCAD*, 2016.
[5] S. Iizuka, in *Proc. ITC*, 2015.
[6] H. Fuketa, *IEEE Trans. VLSI Systems*, 2012.
[7] J. Shiomi, in *Proc. PATMOS*, pp. 44–49, 2016.
[8] H. Awano, in *Proc. ESSDERC*, pp. 218–221, 2014.
[9] B. J. Velamala, *IEEE Trans. Electron Devices*, vol. 60, no. 11, pp. 3645–3654, 2013.
[10] M. R. Guthaus, in *Proc. IEEE Workshop on Work load Characterization*, pp. 3–14, 2001.