TG-SPP: A One-Transmission-Gate Short-Path Padding for Wide-Voltage-Range Resilient Circuits in 28-nm CMOS

Weiwei Shan, *Member, IEEE*, Wentao Dai, *Student Member, IEEE*, Chuan Zhang, *Member, IEEE*, Hao Cai, *Member, IEEE*, Peiye Liu, Jun Yang, *Member, IEEE*, and Longxing Shi, *Senior Member, IEEE*

***Abstract*— Resilient circuits with timing error detection and correction (EDAC) can eliminate the excess timing margin but suffer from the short-path (SP) issue where SPs must be padded to exceed the detection window. SP padding (SPP) is similar to, but severer than, hold time fixing. Thus, it incurs significant area overhead, especially when working in the near-threshold region. In this article, we propose a transmission gate-based SPP (TG-SPP) method, which uses only one transmission gate to extend an SP to the negative clock phase while keeping the critical paths unaffected. Compared with the two-phase latch way or the conventional padding with tens to hundreds of buffers in an SP, our method efficiently decreases the overhead. We develop transmission gate insertion rules and an automatic insertion flow to overcome the complicated intersection problem of short and critical paths. To further reduce the EDAC area overhead, we also propose a lightweight error detection latch that has only two extra transistors compared to a conventional 24-T flip-flop for the conventional way. We implement all the proposed techniques in an SHA-256 chip using the 28-nm CMOS process. Results show that our TG-SPP method achieves the same padding effect as the two-phase latch-based method while reducing both the glitch power and sequential area overhead by a factor of 6×. The fabricated resilient chips are measured to achieve 55%–405% frequency improvement and 38.6%–69.4% power saving compared with the typical margined baseline at the near-threshold region.**

***Index Terms*— Energy efficiency, error detection and correction (EDAC), low power (LP), near threshold, resilient circuits, short-path padding (SPP).**

I. Introduction

DUE to the process, voltage, and temperature (PVT) variations [1]–[5], conservative timing margins or voltage margins are reserved during the design of digital circuits to ensure the correct operation under the worst case condition. As shown in Fig. 1, some margins are caused by static variations such as process variations and aging effect, and others are dynamic variations that affect the circuit's performance at runtime [5], [30], [33]. Among the dynamic variations, fast-changing variations (or simply fast variations) such as inductive undershoots in the supply voltage, clock jitter, and skew take effect in a few clock cycles [5], [33]. Such margins limit the performance and energy efficiency of the integrated circuits (ICs) when they operate under normal or best case conditions.

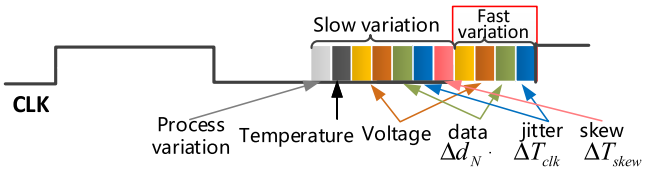


Fig. 1. Worst case timing margins due to PVT variations, including slow variations and fast variations.

Various approaches have been proposed to reduce these margins. A low cost yet less accurate solution is to use critical-path replica ring oscillators (CPR ROs) to sense the critical path delay [32], [33]. Alternatively, *in situ* timing error-detection and correction (EDAC) [4]–[31]-based adaptive voltage frequency-scaling (AVFS) techniques can eliminate such margins while ensuring correct operation across PVT variations. In these techniques, the endpoint registers in most critical paths are replaced by special error-detecting registers, latches, or flip-flops (FFs) to detect the timing information. For example, flop-based EDACs were used in [15], [18], and [25]. Error-detecting latches (EDLs) [4], [5], [10], [19], [21] usually have smaller area overhead. A transition detector (TD) with 17 transistors was designed to detect the positive and negative signal transitions for low-voltage applications [31]. With the monitored information, the AVFS system can tune the supply voltage and/or frequency accordingly to make the circuit operating on the edge of timing failure. However, these techniques suffer from the short-path (SP) issue because EDLs generate a timing error signal when a signal transition occurs during the detection window (usually the positive clock phase), no matter it is a late-arriving signal from a critical path of the previous cycle or a normal signal from an SP of the current clock cycle.

SPs in resilient circuits will cause erroneous error detection and functional failure; thus, they must be padded to be longer

Manuscript received April 9, 2019; revised July 5, 2019 and September 18, 2019; accepted October 9, 2019. Date of publication October 31, 2019; date of current version April 23, 2020. This article was approved by Associate Editor Vivek De. This work was supported by the National Natural Science Foundation of China under Grant 61574033 and Grant 61774038. *(Corresponding author: Jun Yang.)*

W. Shan, W. Dai, H. Cai, J. Yang, and L. Shi are with the National ASIC System Engineering Research Center, Southeast University, Nanjing 210096, China (e-mail: dragon@seu.edu.cn).

C. Zhang is with the National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China, and also with the Purple Mountain Laboratories, Nanjing 210096, China.

P. Liu is with the Beijing Key Laboratory of Intelligent Telecommunication Software and Multimedia, Beijing University of Posts and Telecommunications, Beijing 100876, China.

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSSC.2019.2948164

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

than the detection window. SP padding (SPP) is similar to hold time fixing, but it is much severer in resilient circuits due to its much longer padding requirement. For example, SPs usually need to be padded to be longer than half-a-cycle or positive clock phase. SPP is typically solved by inserting a large number of delay elements (e.g., buffers and inverters) in SPs. Moreover, for low-voltage applications especially those operating at near-threshold or sub-threshold region, variations become more significant [3], [12], [27], [34] and would cause an increased number of endpoint FFs to be replaced by error-detecting registers and more efforts in SPP. In the simulation of an eighth-order filter, the FF replacement ratio is increased to 30% at 0.5 V from 10% at 1.1 V, and the SPs need to be padded to 56% of the clock cycle at 0.5 V [27]. In three-stage pipeline test circuits of multipliers [12], the area overheads of conventional FF-based and latch-based EDAC techniques were up to 2.1× and 40%, respectively, at the near-threshold voltage (NTV).

A common way to decrease the number of SPP cells is to shorten the EDAC detection window, by using the duty-cycle correction. For example, Razor-lite [9] corrected the global clock duty cycle and used initial self-calibration and run-time calibration to reduce the chances of SP data falling within the detection window. This global clock, however, had varying degrees of pulsewidth deformation and amplitude attenuation when it arrived at the leaf registers. Thus, it increased the timing closure effort. The improved iRazor [10] used a local clock generator but still needed to pad the SPs across various PVT conditions. Bubble Razor based on two-phase latches [11] used consecutive opposite-clock-controlled latches to solve the SP problem at the cost of up to 103% area overhead at NTV [12]. Sparse error detection was proposed to reduce the sequential logic cost [12], [22]. Pulse-latch-based EDAC was proposed to reduce the clock duty cycle, along with a multi-Vth cell library to reduce padding overhead [24] caused by the long delay and low power (LP) consumption of the high-Vth cells. PushPull [6] was proposed with a global view to derive padding values, utilizing spare cells and dummy metals to further reduce padding at physical implementation.

In this article, we propose to use clock-controlled transmission gates (CTGs) to solve the SP issue. As opposed to using a series of padding buffers, one CTG can extend an SP to be longer than half a cycle when working as a transparent-low latch, which effectively reduces padding overhead. We develop a CTG insertion mechanism and an automated design flow for real circuit design where there are abundant overlaps of short and critical paths. A preliminary version of this article was published in [27]. To evaluate the robustness and effectiveness of the proposed insertion mechanism, experiments are conducted on an SHA-256 encryption circuit and compared with other SPP methods.

Our main contributions are as follows.

1) Unlike the conventional buffer-based padding method, we use only one CTG in an SP to extend it to over half a cycle while keeping the critical paths unaffected.
2) We exploit virtual buffers in the CTG insertion procedure to solve the insertion problem of the complicated intersections of short and critical paths. The virtual buffers are added in the intermediate process to balance multiple paths and facilitate the identification of proper CTG insertion positions. They will not be inserted in the final circuit. This special flow is customized as an automated design flow to make it suitable for large-scale real-life circuit implementation.
3) We also propose a lightweight EDL for wide-voltage-range operation with a 12-transistor TD. It has only two extra transistors over a typical 24-T FF. We implement the EDL with the proposed CTG insertion mechanism in a 28-nm CMOS process.

Our resilient circuit solves the SP issue with low area overhead and gains up to 105% frequency improvement or 63.58% power saving as compared to the margined baseline when operates at the NTV region.

The remainder of this article is organized as follows. Section II analyzes three representative EDAC techniques and evaluates their overhead of SPPs. Section III presents our transmission gate-based SPP (TG-SPP) method. Section IV presents a lightweight EDL circuit design and the implementation of our proposed EDAC system in a 28-nm CMOS process. Section V shows the measurement results. Finally, Section VI concludes this article.

II. Analysis of Existing Resilient Methods

To understand the cause and solutions of the SP issue, we analyze three representative EDACs and their SPP methods at the NTV region: 1) flop-based EDAC (Section II-A) [18]; 2) two-phase latch-based EDAC (Section II-B) [11]; and 3) single-phase latch-based EDAC (Section II-C) [10]. In Section III-E, circuits using each of these techniques will be designed and evaluated for area overhead of their SPPs.

A. Flop-Based Error Detection

Razor [18], [35] was the first proposed flop-based error-detecting registers. It is mainly composed of a main FF sampling at a positive clock edge, a shadow latch, and an XOR. The shadow latch is either a positive transparent latch [18] or a negative transparent latch with a delayed clock signal [35]. A simplified Razor FF (RFF) with a positive transparent latch is shown on the right of Fig. 2(a); the metastability detector and MUX are not shown for simplicity [18]. As shown in the timing diagram of Fig. 2(c), if there is a timing violation such that the input data of FF arrive late, which means that the critical path exceeds one clock cycle and the output of the shadow latch QL will be different from that of the FF because a latch is transparent in positive clock phase. After being XORed with the output of the main FF, a timing error will be generated.

However, this approach suffers from the SPP issue because SPs are usually intersected with critical paths, as shown on the left of Fig. 2(a) and its timing diagram in Fig. 2(d). If the critical path (Tc1) does not suffer from timing violation, and the data coming from the SP (Ts) arrive within the detection window; it also causes a timing error. However, this is a false timing error. This issue is called an SP issue and it is a challenge for timing error detection. To avoid such a false

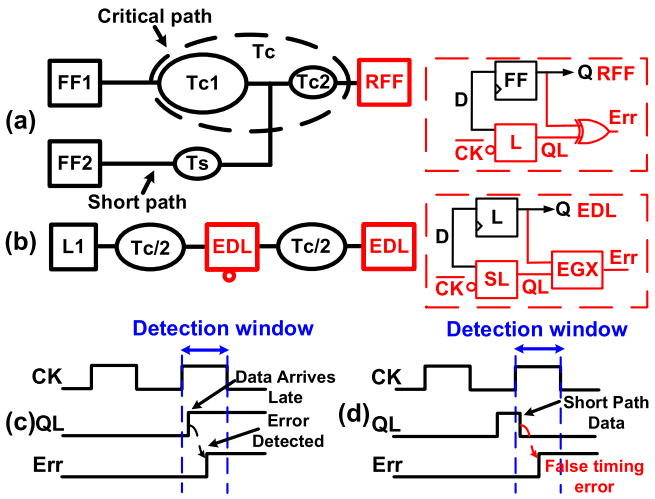


Fig. 2. (a) Illustration of the simplified RFF error detection (RFF refers to Razor FF). (b) Two-phase latch error detection (EDL refers to error-detecting latch and EGX refers to error generation XOR [11]). (c) Error-detection timing diagram. (d) Timing diagram of the SP issue.

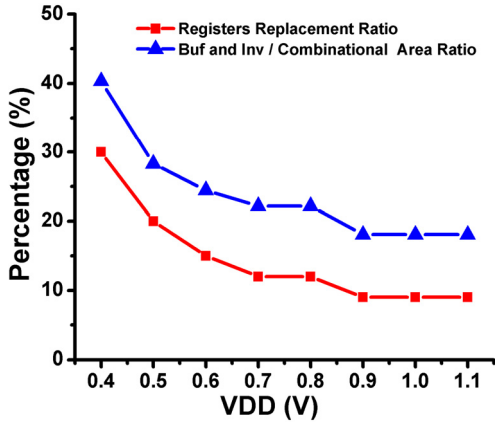


Fig. 3. Registers replacement ratio and SPP cells to combinational area ratio of padding cell across a wide VDD from 0.4 to 1.1 V, for an eighth-order filter circuit [27].

error, padding cells (e.g., buffers and inverters) are usually inserted to ensure that the delay of SPs will exceed the length of the detection window, which incurs the area overhead. This SPP issue also happens in other flop-based EDAC designs such as Razor II [19] and Razor-lite [9].

When VDD decreases, the SPP issue becomes severer because of two reasons. First, delay variation becomes more significant at lower VDD [3], [12], [27], [34]. More paths are likely to cause timing violations due to higher sensitivity to PVT variations. Therefore, the insertion rate of error-detecting registers will increase and the increased error-detecting registers will cause more SPs, further enlarging the SPP overhead. We investigate the number of error-detecting registers needed when VDD scales down on the eighth-order filter circuit in a 28-nm CMOS process. As shown in Fig. 3, the register replacement ratio increases so much when VDD decreases that over 30% of endpoint FFs must be replaced by EDACs when VDD = 0.4 V. As a result, more padding cells will be needed to fix the SPs. The second reason is that SPs endure severer variation at NTV. For example, when VDD = 0.4 V, paths need to be padded to be longer than 58% of TCK when considering 3σ delay variation incurred by local process variations [27].

B. Two-Phase Latch-Based Error Detection

Bubble Razor [11] overcomes the SP issue by converting the FF-based design to the two-phase-latch based design [Fig. 2(b)] using commercial retiming tools. By using the consecutive opposite error-detection latches, one latch stage is transparent, whereas the other latch keeps shut-off in the positive clock phase. They reverse to the opposite state during the negative clock phase, and thus, the SP issue is avoided.

However, this approach incurs extremely high overhead in sequential logic because of three reasons.

1) Replacing all the registers by two-phase latches leads to about twice as many latches as that in the FF-based design and, therefore, twice more critical paths. Therefore, it will have a high replacement rate of error-detecting registers.
2) These latch-based paths are about half of an FF design, and thus, each stage has less averaging effect across logic gates, which worsens the delay variability induced by local variations [29]. Therefore, the SP issue also worsens.
3) A pair of latches has about 45% larger area than the area of a 24-T DFF.

Furthermore, this method needs to modify the entire system architecture since all the FFs need to be converted into the two-phase latches in register transfer level (RTL), which causes difficulty in automatic design flow. Finally, it becomes challenging for the timing analysis because the complex clock network of the two-phase latches makes the total design very complicated.

C. Single-Phase Latch With Error Detection

Single-phase latch-based EDACs have attracted much attention recently because of its low area/power overhead. In these approaches such as TD with time borrowing (TDTB) [4] and iRazor [10], it is usually composed of or integrated with the function of a TD and a latch. Similar to the flop-based EDAC techniques, they replace the endpoint FFs by EDLs instead of the error-detection FFs. Thus, they have small area overhead and the advantage of using time-borrowing ability to improve the performance.

However, the SP issue is usually severer because the latch is transparent during the entire positive clock phase. The area overhead due to SPP is related to the length of the detection window (the positive clock phase). Having a typical 50% duty cycle enables sufficient detection window and time-borrowing ability, but it also requires much more padding cells to extend the SPs. Moreover, the overhead will increase further when the supply voltage decreases, especially when it gets close to the NTV region. This is due to the same reasons as we have analyzed above in the case of flop-based error detection.

III. TG-SPP

In this section, we elaborate on our proposed TG-SPP to address the SPP issue. Considering that SPs and long paths

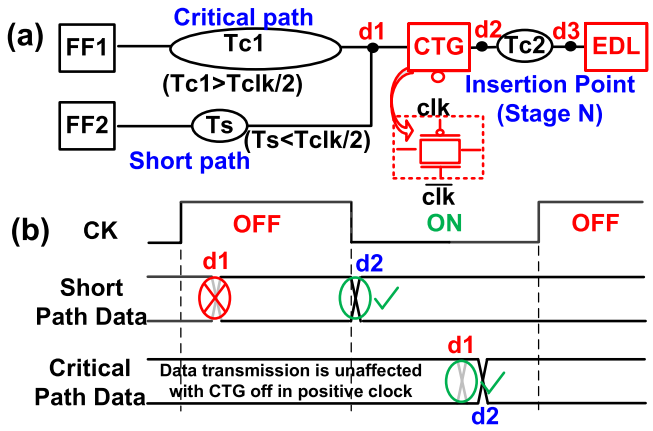


Fig. 4. (a) Illustration of our TG-SPP with a transmission gated inserted in near the critical path end. (b) Its impact on critical-path and SP data. Abbreviations: Tc1 refers to most the majority part of critical path delay; Tc2 refers to the small part of critical path delay; Ts refers to SP delay.

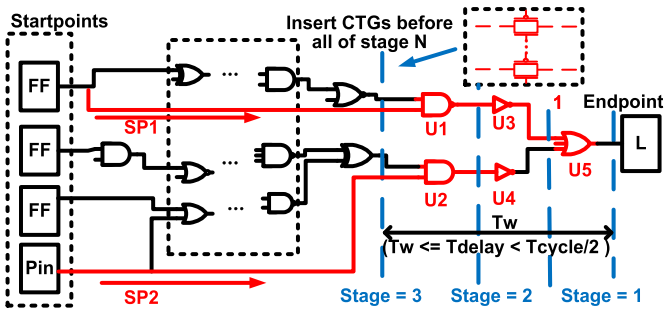


Fig. 5. Exemplary illustration of the CTG insertion mechanism in a circuit.

usually will overlap, we first give the overall concept and establish CTG insertion guidelines, and then we explore all the possible cases to develop an appropriate CTG insertion strategy in circuit implementation.

A. Concept of Transmission Gate Padding

The basic idea of our TG-SPP is to extend the SP to over the positive clock phase while keeping critical paths unaffected. As shown in Fig. 4(a), the path from FF1 to EDL is critical and the path from FF2 to EDL is an SP and they share a common part of Tc2. CTG is a basic transmission gate that is composed of parallel-connected PMOS and NMOS transistors. It works as a transparent-low latch with NMOS's gate connected to $\overline{\text{clk}}$. A CTG is inserted in the common part of the two paths. As the CTG is turned off during the positive clock phase, the SP data cannot be propagated to the endpoint latch (or EDL) until the falling clock edge arrives [Fig. 4(b)]. Thus, the SP can be easily extended to over half a cycle. On the other hand, although a CTG is inserted in the critical path too, its setup timing constraint is only slightly affected since the path delay is longer than half a cycle and the CTG is turned on in the positive clock phase [as shown in Fig. 4(b)]. The delay extension of the critical path is only a CTG's delay time, which is approximately equal to the delay of an inverter. It is small enough when compared to a typical critical path of tens of stages. As a comparison, the buffer-based SPP usually has an even longer delay extension when multiple buffers are inserted in the critical paths.

In the EDAC techniques, the endpoint FFs of critical paths are replaced by error-detecting registers (here we use EDLs). Thus, most of those replaced paths start with an FF and end with a latch, and a few paths start with latches. However, the SP issue is caused by the endpoint latch, no matter whether the startpoint is an FF or a latch. The analysis of SPs starting with latches is the same as those starting with FFs. Therefore, for simplicity, the following analysis will be illustrated on paths starting with FFs.

The CTG insertion needs to follow some basic rules because SPs are usually intersected with long or even critical paths. A basic rule is that a CTG needs to be inserted close to the critical-path endpoint but not too close to it. The reason is that, if a CTG is placed too close to the endpoint latch, the critical path (along with the CTG and the endpoint EDL) might be extended to the next clock cycle when a fast variation (mainly come from IR-droop, clock jitter, etc.) occurs. In this case, EDL is supposed to detect this timing failure, but it fails to do so because CTG is off in the positive clock phase, which stops the input data from being propagated to EDL. Thus, the target insertion stage N needs to have at least a certain delay time (defined as Tw) from the endpoint to avoid timing failures. To be clear, there is no such constraint for a single SP.

Tw is a timing window, which equals the critical path delay variability caused by fast variations, and thus, it can be set similarly as the detection window setting in conventional resilient designs [9], [17], [19], [25]. Tw is converted to N stages as illustrated in Fig. 5. The number of stages, N, is defined as the cell count number starting from the endpoint latch to the current point. For example, both input pins of U1 and U2 have $N = 3$. After a target stage N is determined, we insert CTGs before all the points with stages equal to N. The CTG works as a transparent-low latch, which means it turns off in the positive clock phase and turns on in the negative phase. Thus, all the SPs (e.g., SP1 and SP2 in Fig. 5) can be extended over the positive clock phase.

Compared to the traditional latches, one major potential problem of CTG is the leakage on the CTG high-impedance output when it is turned off. To study the risk for a data loss due to the leakage, we perform 10 000 Monte Carlo simulations to obtain the worst case retention time. The results show that the minimum leakage time is 62 ns and the mean leakage time is 341 ns when operating at NTV for the worst leakage case (worst leakage corner, high temperature). This implies that leakage will not cause a false data transition if the clock period is greater than twice the minimum time, which equals working at a frequency higher than 8 MHz (worst) or 1.5 MHz (normal) at NTV. This minimum frequency requirement is easily met at either NTV or super threshold voltage (STV), as can be seen from the measurements in Section V.

In order to show the reliability of CTG, we further analyze its yield when operating at a conservative frequency of 5 MHz

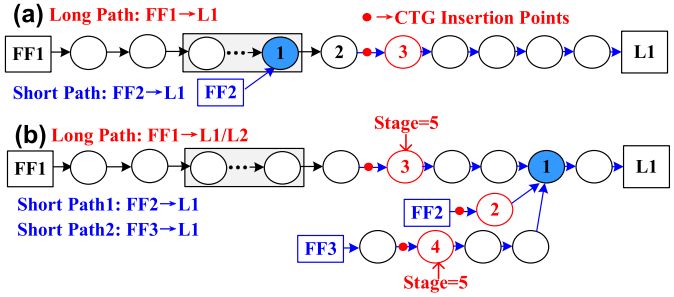


Fig. 6. Illustration of CTG insertion points selection—Case I: critical path and SP share the same endpoint. (a) Overlap node to endpoint latch with stage $> N$. (b) Overlap node to endpoint latch with stage $\leq N$.

across a wide range of VDDs. The yield of CTG is simulated by Nanospice (a yield simulation tool) to obtain the probability of CTG's retention due to leakage at a target clock frequency. A high CTG yield means that there will not be leakage-induced retention in a target clock frequency. Simulation results are combined with those of TD in Section IV, which confirms that the yield of CTG is better than FF's yield for data transmission. Therefore, it is reliable.

B. CTG Insertion Principles

As the CTG-based SPP method is independent of the actual function of the circuit, it can be plugged into any circuit as a generic method. It is complicated to determine the proper positions for CTG insertion since the SPs usually overlapped with some critical paths. In the overlap paths, a node may belong to different paths so it may have a different stage value in each path. Such a node is defined as an overlap node. Therefore, we cannot use the simple rule of inserting one CTG before all the nodes with stage $= N$ in each path because it may lead to multiple CTGs inserted in the same path. Multiple CTGs in the same path lead to timing errors because CTGs are controlled by the clock so that multiple CTGs prevent the input signal from passing to the endpoint registers. Thus, we need to guarantee that inserting CTGs in the SPs do not affect the timing of critical paths. Here are two basic guidelines for the CTG insertion: 1) multiple CTGs in one path are not permitted and 2) critical path insertion points cannot be too far from the endpoint; while for an SP, there is no such restriction.

These two guidelines are used to set the rules for CTG insertion. According to the definition of SP issue in Section II-A, when there is an SP issue, that SP must have one or more overlap nodes with one or more endpoint EDLs. This is because otherwise there will be no SP issue. There are two cases depending on whether the overlap node is shared by one endpoint or two endpoints. As illustrated in Figs. 6 and 7, node ◯ stands for a combinational logic gate, blue nodes are overlap nodes, red dots are CTG insertion positions, FF1–FF4 are startpoint FFs, and L1–L4 are endpoint latches. To be clear, the startpoint registers may also be latches, but the endpoint registers we care about are latches, which are used together with TDs.

By analyzing all the sub-cases in each of the two cases, we obtain the related CTG insertion methods as follows.

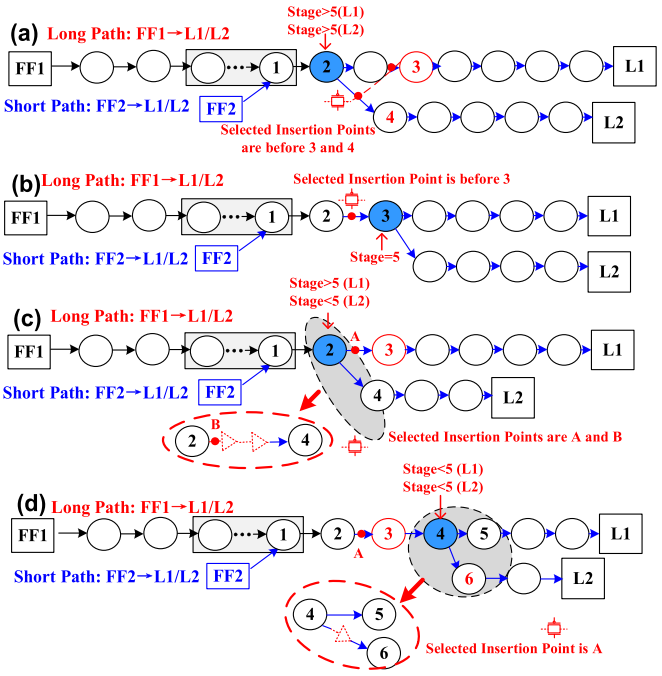


Fig. 7. Illustration of CTG insertion points selection—Case II: critical path and SP end with different endpoints. Define the stage of L1 to the overlap node as P1, and the stage of L2 to the overlap node as P2, we have four sub-cases. (a) P1 $> N$, P2 $> N$. (b) P1 $= $ P2 $= N$. (c) P1 $>$ N && P2 $\leq N$ or P1 $\leq N$ && P2 $> N$. (d) P1 $\leq N$, P2 $\leq N$.

Case I—The Critical Path and the SP Share the Same Endpoint:

This is a simple case as shown in Fig. 6, where FF1 to L1 is the critical path (long path), and FF2 to L1 is the SP. There are two situations in this case, depending on whether the stage between the endpoint latch (L1) and the overlap node is smaller than N [Fig. 6(a), assuming $N = 5$ for example]. If stage $\geq N$, CTG is inserted before the Nth stage, which is node 3 in Fig. 6(a). When overlap node's stage $< N$, as in Fig. 6(b), one CTG is inserted before node 3 in the critical path, and one CTG is inserted in an SP. If the SP's stage $< N$ (as in the path from FF2 to L1), a CTG is inserted after FF2. If it is $\geq N$(as in the path from FF3 to L1), a CTG is inserted before the Nth stage. By doing this, the SPs are extended. Here, the critical path does not have to be inserted by a CTG, but it is easy for realization by using the same rule as shown in Fig. 6(a).

Case II—The Critical Path and the SP Have Different Endpoints:

Case II is quite common in very large-scale integration (VLSI) circuits, where the critical path and the SP have an overlap node but different endpoint latches. As shown in Fig. 7(a)–(d), FF1 → L1 is the critical path and FF2 → L1/L2 are two SPs. Define the stage of L1 to the overlap node as P1, and the stage of L2 to the overlap node as P2, we have four sub-cases: 1) P1 $> N$ && P2 $> N$; 2) P1 $=$ P2 $= N$; 3) P1 $> N$ && P2 <= N or P1 <= N && P2 $> N$; and 4) P1 <= N && P2 <= N. Here are their solutions.

1) *P1 $>$ N, P2 $>$ N:* Insert a CTG before the node of stage $= N$ in each path, which are node 3 and node 4 as in Fig. 7(a).

2) *P1 = P2 = N:* Insert a CTG before the overlap node, that is, node 3 as in Fig. 7(b).
3) *P1 > N && P2 <= N or P1 <= N && P2 > N:* For these two situations, we can use the same solution. We insert some buffers (called virtual buffers because they will not be inserted in the final circuit) after the overlap nodes to balance the stage number between multiple paths, making the minimum stages of overlap node to each endpoint be lengthened to be N.
   After balancing the SPs, insert CTGs before stage N in each path, as point A and point B in Fig. 7(c). In the case when P1 $=$ N or P2 $=$ N, no virtual buffer is needed.
4) *P1 <= N, P2 <= N:* Insert virtual buffers after overlap nodes so they are balanced to have N stage, and then insert a CTG before the stage N point. In the example shown in Fig. 7(d), a virtual buffer is inserted between node 4 and node 6, and a CTG is inserted before common node 3 whose stage is N.

C. CTG Insertion Strategy and Algorithm

The above two cases and six sub-cases cover all the possibilities for CTG-based SPP. According to the above analyses, we define the following CTG insertion strategy: first, find out the minimum stages of each overlap node based on static timing analysis (STA) tools and customized scripts. If an overlap node's stage is less than N, a virtual buffer is added after the node. Repeatedly add virtual buffers until the minimum stages of all overlap nodes are greater than or equal to N. Second, determine the candidate insertion positions to the stages before N in the netlist. Finally, copy the candidate insertion positions to the original netlist without virtual buffer insertion. Therefore, the SPs are extended correctly. Since virtual buffers are not actually inserted in the netlist, they will not affect the timing or increase the circuit area. Moreover, critical paths are weakly affected by only the delay of a CTG at the negative clock cycle.

To show the effectiveness of our CTG insertion strategy, we use a non-trivial example that includes more than one sub-case as shown in Fig. 8(a). There are seven SPs: 1) FF1 to L1; 2) FF2 to L1; 3) FF2 to L2; 4) FF3 to L1; 5) FF3 to L2; 6) FF3 to L3; and 7) FF4 to L4, while FF3 to L4 is a critical path. Assuming $N = 5$, if we simply insert CTGs before all the nodes with stage $\leq N$, the following nodes should be selected corresponding to the above nine cases when we count five stages from the endpoints of L1/L2/L3: nodes ①, ②, ⑤, ④, ③, and ⑦, respectively. However, inserting multiple CTGs (nodes ③, ④, and ⑤) in one path (FF3 to L1/L2/L3) causes a timing error. Therefore, we insert virtual buffers to solve the overlap node's problem. As illustrated in Fig. 8(a), the overlap nodes are nodes 3 and 10. After inserting virtual buffers, these SPs are balanced, as shown in Fig. 8(b). The final insertion positions for CTGs are selected as A, B, C, and D, with equal stage length to the endpoint registers.

According to the above virtual buffer-based insertion strategy, a complete CTG insertion algorithm is developed to find the appropriate CTG insertion points automatically. This algorithm contains three steps as shown in Fig. 9.

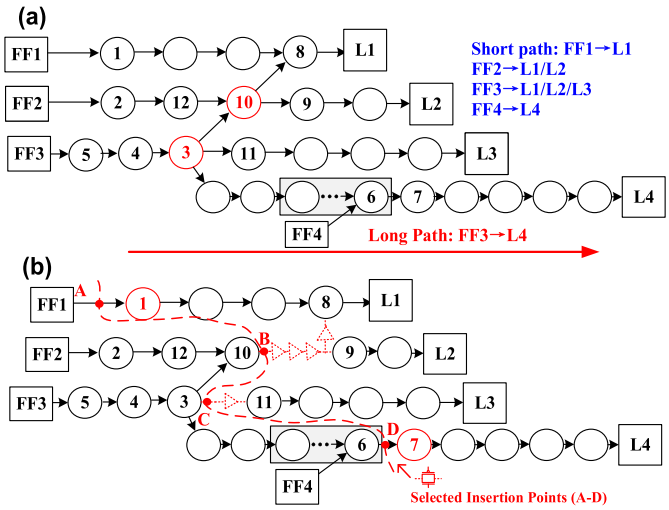


Fig. 8. Example of CTG insertion points selection. (a) Complex paths. (b) Adding virtual buffers to help find the insertion points.

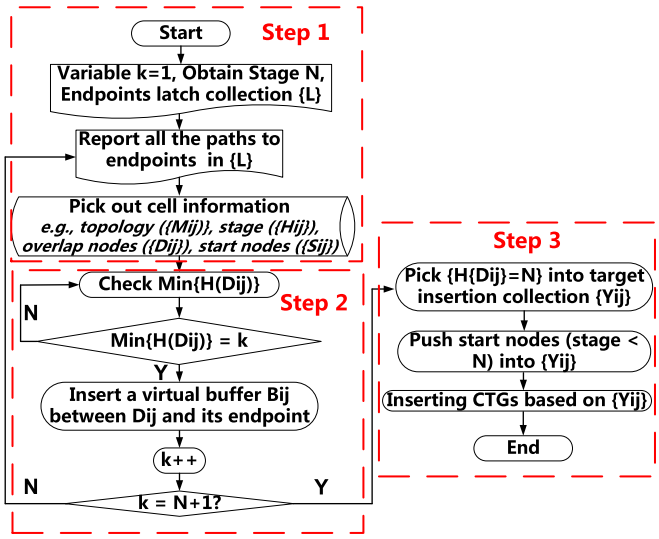


Fig. 9. Automatic CTG insertion algorithm.

Step 1: Initialization. Report paths by STA tools and generate cell information by customized scripts. Cell information includes topology {Mij}, stage {Hij}, overlap nodes {Dij}, and start nodes {Sij}. Here Mij refers to node topology (e.g., node A is before node B and to endpoint C). Hij refers to the node's stage (e.g., the stage of node A to endpoint C). Dij refers to the overlap nodes (e.g., nodes belong to both endpoints C and D). Sij refers to the start nodes (e.g., node E is the startpoint of a path to endpoint C). Yij is the target insertion positions.

Step 2: Balance each path by inserting virtual buffers, as described in Fig. 7. First, find the minimum stage (k) of overlapping nodes Dij. Then, insert virtual buffers before Dij and repeat until k equals N.

Step 3: Select the nodes, {Yij}, with stage $= N$ for CTG insertion. Finally, insert CTGs based on {Yij}.

TABLE I
DETAILED INFORMATION OF STUDY CIRCUITS IN THIS WORK AT SS, 0.5 V, 0 °C

Circuits	Total gates	Total FFs	Critical FFs (Insertion rate)	Logic length (Maximum)	Short paths number (<TCK/2)	Short paths' WNS* (Norm to a FO4 buffer)
S5378	2958	179	19 (10.61%)	12	244	778
S9234	5808	211	67 (31.75%)	22	1550	4806
S15850	10306	534	112 (20.97%)	24	7113	16208
S13207	8589	638	29 (4.55%)	21	547	1737
S38584	20679	1426	459 (32.19%)	22	7187	22071
S38417	23815	1636	384 (23.47%)	23	24008	53800
S35932	17793	1728	170 (9.84%)	10	1601	4887
SHA-256	38481	4897	607 (12.4%)	31	21648	154504
Cortex-M0	14276	841	126 (14.9%)	32	15206	37524

* Short paths' WNS refers to the worst-case negative slack, which represents the overhead for short-path padding. It is obtained by calculating the accumulated short-path delay slack to TCK/2.

It should be noted that the procedure of inserting virtual buffers causes some iterations, but its computational cost is small and acceptable. The main computational cost comes from the position selection of virtual buffers and timing information update of cell topology. In our application circuit of SHA-256, it only takes less than half an hour for all the iterations.

D. TG-SPP Applicability Discussion

Our TG-SPP is a generic method that can be plugged into circuits to solve the SP problem, just like buffer-based SPP, because CTG-insertion is based on the STA results and our algorithm. It is independent of the circuit's functionalities. However, its effect in reducing the area overhead of SPP depends on how severe the SP issue is. More precisely, if there are a lot of SPs whose lengths are quite short, TG-SPP usually outperforms buffer-based SPP in area overhead.

In order to evaluate its applicability and effectiveness in reducing area overhead, we study various circuits in the International Symposium on Circuits and Systems (ISCAS) benchmark. After excluding those small-scale circuits with less than 100 registers, we select the following seven sample circuits: S5378, S9234, S15850, S13207, S38584, S38417, and S35932. We also include the SHA-256 circuit and Cortex-M0 circuit. SHA-256 circuit is a computation-intensive encryption circuit, and Cortex-M0 is a classic microprocessor. These circuits have different SP issues where some are quite severe and others are not.

We synthesize these nine circuits using the same 28-nm CMOS and obtain their path delay distributions by STA. They have different FF numbers and total gates, as well as different structures in terms of SPs. Table I summarizes the detailed information of those circuits, including insertion rate, SPs' delay distribution, total SP worst-case negative slack (WNS), and so on.

As given in Table I, first, they have logic lengths of 10–32, which makes it tolerable when inserting one CTG gate's delay in some critical paths. Second, EDL replacements of endpoint FFs are also different in each circuit depending on their structure, which is selected as those paths whose slacks are less than 15% of the clock cycle (the size of detection window) as commonly set in EDACs [27]. Thus, the endpoints of detecting paths of each circuit are obtained. Based on the insertion of EDLs, the SPs with intersection to the monitored critical paths can be found. Third, some circuits have very severe SP issue, such as S15850, S38417, and SHA-256, that they not only have a large number of SPs with delay less than half of a clock cycle (TCK) but also have severe WNS of SPs represented by accumulating the SP delay slack to TCK/2.

For circuits with severe WNS, our TG-SPP outperforms the traditional buffer-based SPP in terms of area overhead. To show the effectiveness, we normalize those SP WNS to an FO4 buffer delay and estimate the normalized area overhead of each circuit based on our TG-SPP and buffer-based SPP, respectively. The results are shown in Fig. 10. Here, the area overhead is estimated by multiplying the cell (CTG or buffer) area by the number of such cells and then divided by the total circuit area. As one can see from this figure, our method outperforms the buffer-based SPP on all circuits. Its advantage varies in different circuit structures. It is especially useful for S15850 and S38417, which have severe SP issues. Its effect can be further improved by using our proposed CTG-insertion algorithm. The SHA-256 circuit also has severe SP issues and it has the third-highest improvement among these circuits.

E. Circuit Application and Circuit Overhead Comparison

To verify the effectiveness of our proposed method, we apply it to a large VLSI circuit that implements SHA-256. It is an encryption circuit with over 4500 registers and more than 30 000 logic cells. According to the selected critical paths, endpoint registers are replaced by EDLs first. Then, we choose the insertion points and insert CTGs.

To evaluate the effectiveness of our method in reducing the area overhead, we compare with the following three approaches: (I) no resilient technique which is used as the

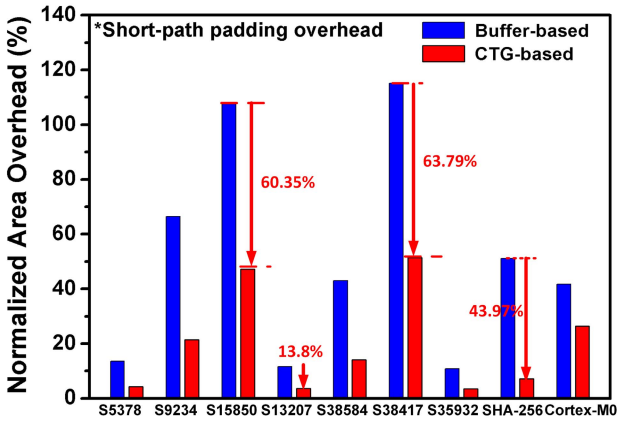


Fig. 10. Normalized area overhead comparisons between conventional buffer-based padding method and our TG-SPP, for 7 ISCAS circuits, HAS-256, and Cortex-M0.

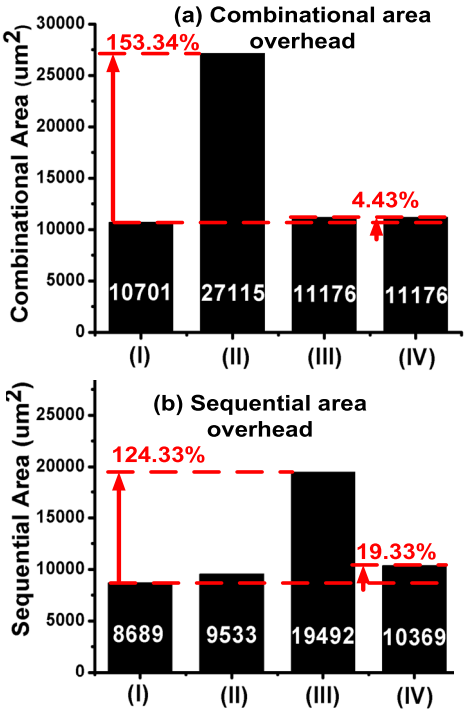


Fig. 11. SPP area overheads of four EDACs on SHA-256 at NTV. (a) Combinational overhead. (b) Sequential overhead. Abbreviations: (I) baseline without EDAC, (II) flop-based resilient circuit, (III) two-phase latch-based resilient circuit, and (IV) our TG-SPP method.

baseline, (II) a conventional resilient technique with the flop-based sequencing logic Razor [18], and (III) the Bubble Razor error detection technique with two-phase latch-based sequencing logic [11]. We refer to our proposed transmission gate with an error-detecting technique as (IV). To obtain a fair comparison, we use the same EDL as that in technique (III) of Bubble Razor, which is composed of a main latch, a shadow latch with an opposite phase, and an XOR gate [11]. Unlike the two-phase latch design, our proposed method replaces only critical paths with latches partially. Thus, it simplifies the design complexity and leads to potentially small sequential logic overhead.

The above four techniques (I)–(IV) are applied on the same SHA-256 circuit for energy-efficiency design working at NTV of 0.55 V using a 28-nm CMOS process. Techniques (III) and (IV) use the same EDL as shown in Fig. 2(a) in Section A, composed of the main latch, a shadow latch with an opposite phase, and an XOR gate. Fig. 11(a) shows the results of their area overhead comparisons for combinational logics. Technique (II) incurs more than 1.5× combinational area overhead due to the excessive amount of SPP buffers. Technique (III) solves the SP issue by two-phase latches and thus needs much less combinational logic than (II). Compared to technique (II), our proposed method (IV) solves the SP issue while reducing overhead on combinational logic from 153.34% to 4.43%.

On the other hand, sequential area overheads are shown in Fig. 11(b). Technique (II) increases 9.7% on sequential area over the baseline (I). Technique (III) has an overhead of up to 124.33% on sequential area because of its higher EDL replacement rate. As many as twice of the latches are used in (III) as the number in the flop-based design (II), causing severer SP problem. Since our proposed technique (IV) only replaces selected endpoint FFs by EDLs, its sequential area overhead is only 19.33%. Therefore, our TG-SPP achieves the same padding effect as the two-phase latch but reduces the sequential area overhead from 124.33% to 19.33%, which is a factor of 6× compared to that of (III).

The other benefit of TG-SPP is the decrease in glitch power. That is because CTGs synchronize multiple SPs, which reduces the invalid flips of combinational logic. Compared with the baseline circuit without using CTGs, it saves 46.06% glitch power when 1000 CTGs are inserted in the circuit.

IV. Lightweight EDL and Resilient Circuit Implementation

In this section, we propose a lightweight EDL and implement it with TG-SPP in a 28-nm resilient circuit. The fabrication technology is 28-nm LP CMOS. The NMOS transistor's threshold voltage varies between 474 and 560 mV (for slow-slow (SS) corner, −20 °C, 0.55 V) based on HSPICE simulation of an inverter.

A. Error-Detecting Latch Design

To reduce the overhead of resilient circuits, we also need a lightweight EDL that can work reliably across a wide voltage range. Earlier error-detecting units [4], [5], [8], [11], [25] usually had a significant area overhead with more than ten extra transistors over a standard FF. Razor-lite (with eight additional transistors over an FF) [9] and iRazor [10] (three extra transistors over a latch) effectively reduce the area overhead; however, they suffer from threshold loss issue, which makes them unsuitable for NTV applications.

Motivated by this, we propose a new lightweight EDL for wide-voltage-range (from NTV to normal VDD) operations. As shown in Fig. 12(a), the proposed EDL consists of the main latch (transparent high) and a shadow TD. During the positive clock phase (detection window), both the main latch and the shadow TD sample the input data. The TD detects

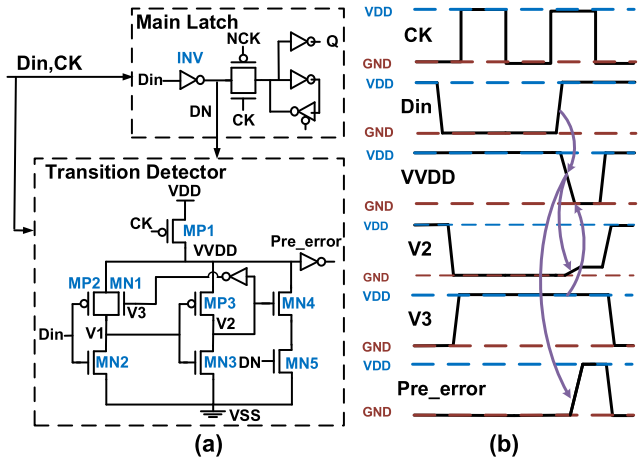


Fig. 12. Proposed EDL. (a) Schematic. (b) Operation example.

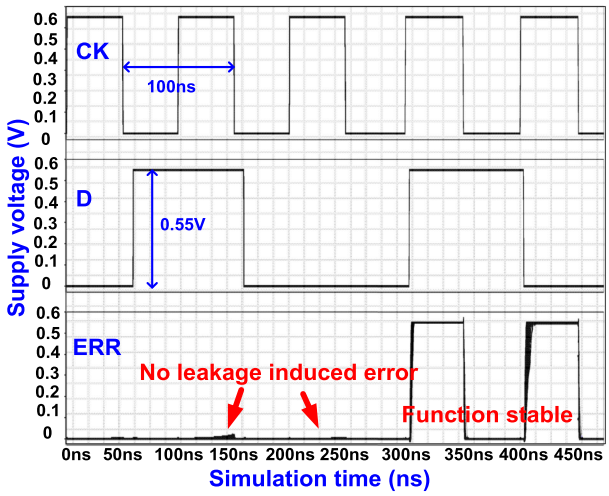


Fig. 13. Monto Carlo simulations of the proposed EDL at NTV of 0.55V.

any data transition during the positive clock phase as a timing error. This offers the following advantages: 1) TD generates a full-output-swing signal without threshold loss because of the full discharge of the virtual power node (VVDD), enabling it to work at NTV without using any skewed output inverter and 2) EDL has low area overhead, LP consumption, and fast response time when compared to a conventional FFs.

Fig. 12(b) shows the timing of our TD. During the negative clock phase, node VVDD is pre-charged and initialized to 1, and it is independent of any input transitions. VVDD becomes a floating node during the positive clock phase. If the input data arrive after the rising clock edge, it is viewed as a late-arriving signal, which makes TD to generate a positive Pre_error pulse.

We now show the working principle by the example of a 0-to-1 data transition in the positive clock phase. As transistor MP1 is off, when the voltage of Din increase, VVDD is first discharged through MN1 and MN2, which postpones V2 from being charged to 1. Hence, it gives sufficient time for V3 to remain at 1 such that VVDD can be fully discharged without any threshold loss. Then the output signal (Pre_error) turns to 1. During the falling-detecting case (when input data transits from 1 to 0), in the positive clock phase, the same output pulse is generated in a similar way. Here, the initial states of inner nodes are: DN = 0 (MN5 turned off), V1 = 0, V2 = 1 (MN4 turned on), and V3 = 0 (MN1 turned off). When Din drops from 1 to 0, MN5 is turned on. Due to the device delay, MN4 is still on, leading to the discharge of VVDD (through MN4 and MN5). During VVDD discharging, note that there is a threshold loss in MP3 to keep MN4 being turned on in order to ensure that VVDD can be discharged to 0.

We analyze a potential leakage problem that may be triggered by process variations. That is, during the positive clock phase, VVDD will be discharged to some degree through device leakage even when there is no data transition. To prevent this leakage from generating an unwanted error signal, we increase the size of MP1 to resist the discharge. In addition, since VVDD is refreshed after each clock falling edge, the leakage will not be a problem. To verify the stability at NTV, we perform Monte Carlo simulations with 10-MHz clock frequency at 0.55 V. From the results shown in Fig. 13, one can see that there is no error caused by leakage when there is no data transition in the positive clock phase. It also shows that the proposed EDL is functional stable at NTV when there is a data transition in the positive clock phase.

We further test the yield of the proposed EDL by Nanospice Monte Carlo simulation at the FF/SS corner, 25 °C and a wide VDD range (from 0.4 to 1.0 V). Here Nanospice is a yield simulation tool. The yield of EDL implies EDL's ability to detect data transition before the falling edge of the clock. To better evaluate the robustness of the proposed EDL, we choose a conventional DFF in the circuit for comparison. DFF's yield is simulated on a timing yield basis [36], [37] with proper settings of working frequency and D-to-CK to measure the output of Q in a certain CK-to-Q period. Taking the 0.6-V supply voltage as an example, the target operating frequency is set as 20 MHz, which is an appropriate frequency for NTV operation. Considering the length of the critical path in our circuit, we provide adequate CK-to-Q time (20% of clock period, 10 ns) to measure whether the output of DFF is correct while keeping D-to-CK (also set as 10 ns) large enough to avoid violating the setup constraint.

As shown in Fig. 14, the y-axis refers to the yield of each device, and $Y = 3$ sigma means the probability of the device working correctly is 0.9973. A high sigma value indicates a good device yield. Fig. 14 shows that whether in SS or FF corner, the proposed EDL as well as CTG have a higher yield than the conventional DFF, especially at low supply voltages. To be clear, the yield of DFF appears to be low at SS corner and low voltages. That is because the CMOS process we used is the 28-nm PolySiON (PS) process, which has a more severe variation than high-K metal gate (HKMG) process. In addition, since there are many more DFFs used in a circuit than EDLs, the probability of EDL's failure is even lower than that of DFF. This means that the proposed EDL and CTG will not be the bottleneck of the circuit in terms of yield.

The proposed EDL's performance parameters are listed in Table II and compared with a conventional 24-T FF from

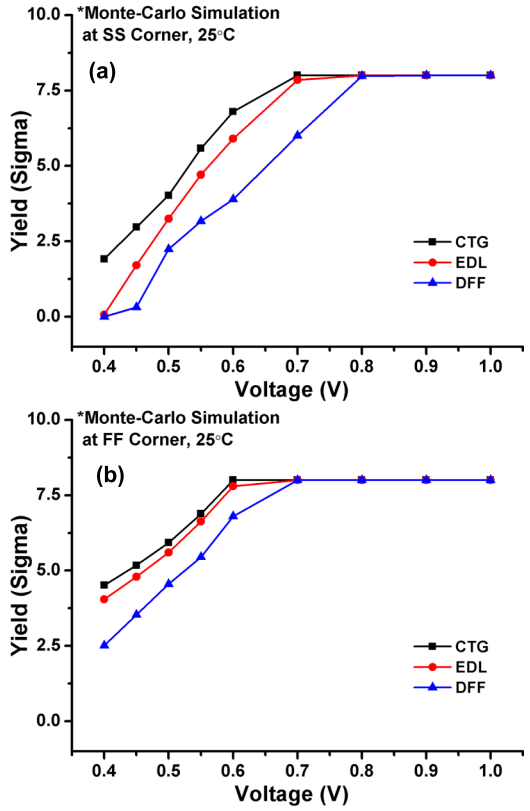


Fig. 14. Yield analysis of the proposed EDL, CTG, and a standard FF cell across wide VDDs (from 0.4 to 1.0 V). Yield at (a) SS corners and (b) FF corners at 25 °C.

TABLE II
COMPARISON BETWEEN OUR EDL AND A CONVENTIONAL FF

	Conventional Flip-flop	Proposed EDL
Transistors	24	26
Area	2.43 um^2	2.64 um^2
Leakage Power	0.26nW@1.1V	0.359 nW@1.1V
	0.023nW@0.55V	0.024 nW@0.55V
Dynamic Power	0.37uW@1.1V	0.437 uW@1.1V
	0.1uW@0.55V	0.06 uW@0.55V
CK-Q Delay	78.8ps@1.1V	70.13 ps@1.1V
	2.452ns@0.55V	1.765 ns@0.55V
Response Time	N/A	2.6 FO4 inverter delay

the standard cell library. Since there are fewer clock nodes in EDL, it has a lower dynamic power as compared to a standard FF, which is 0.79× at normal VDD and 0.7× at NTV with a toggle rate of 50%. In addition, its CK-Q delay and setup time are smaller than those of FF too, giving it advantages to replace the FFs in a critical path. The response time of EDL is 2.6× of an FO4 inverter's delay, which is fast enough to be used in error propagating. In addition, its area overhead is only 8.9% over a general FF. In summary, our proposed EDL has LP and area overhead and is fast for detecting timing errors.

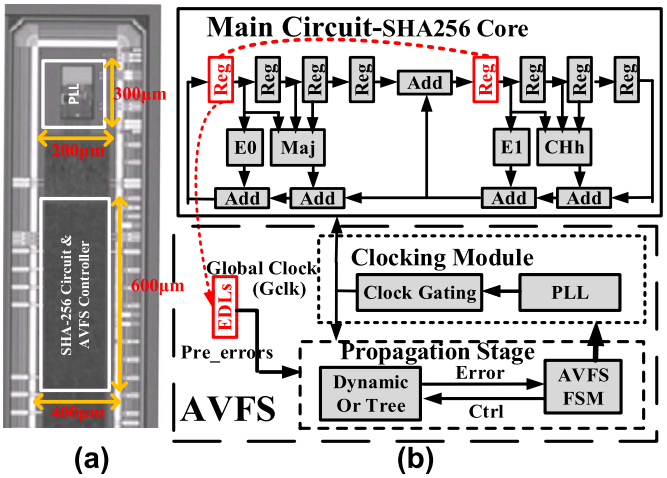


Fig. 15. (a) Chip die photograph. (b) Overall architecture with our proposed EDAC technique.

Table III reports the comparison results of our EDL and other error detection circuits in terms of transistor number, area overhead, metastability, and NTV operation. Compared to Razor II [19], TDTB [4], TD [31], Bubble Razor [12], and Razor-lite [9], our EDL has the fewest transistors as it requires only two more transistors than a conventional 24-T FF. Among the other EDACs, iRazor [10] has the smallest area overhead, but it suffers from a threshold loss on its sensing node since it uses an NMOS to charge its floating node. Thus, its operating voltage cannot be reduced to below 0.6 V. Compared to iRazor [10], Razor-lite [9], and Bubble Razor [12], which cannot operate at NTV, our EDL is able to operate across a wide voltage range (from 0.5 to 1.1 V) without metastability.

B. System Design and Implementation

According to the study in Section III-D, S15850, S38417, and SHA-256 circuits have a very severe SP issue. When SHA-256 works at the NTV region in 28-nm CMOS, the SPs with delay less than 56% of the clock cycle should be padded [27]. In addition, among the nine benchmark circuits in Section III-D, SHA256 circuit has the largest scale with the most logic gates and the most registers. Thus, we select SHA-256 as our implementation circuit.

Fig. 15(a) shows the die photograph of our resilient SHA-256 circuit in a 28-nm CMOS process, with a core area of 0.24 mm^2. The overall architecture of our proposed EDAC technique is shown in Fig. 15(b), which is composed of the main circuit with EDLs replacing the critical endpoint FFs and an AVFS module. AVFS module is composed of Dynamic OR gates, AVFS finite-state machine (FSM), and a clocking module. Most existing AVFS systems decrease the supply voltage (VDD) to the point of the first failure (PoFF) for a given frequency. As for the latch-based design, PoFF does not make an immediate timing error due to the time-borrowing ability of the endpoint latch. However, the borrowed clock shortens the time of the next clock cycle; thus, a recovery mechanism is needed to avoid the accumulated timing error.

TABLE III
COMPARISON TABLE WITH EXISTING ERROR-DETECTING REGISTERS

	Razor II [19]	TDTB [4]	TD [31]	Bubble Razor [12]	Razor-lite [9]	iRazor [10]	Our EDL
Type	Latch	Latch	Latch	Latch	Flip-Flop	Latch	Latch
Extra of Transistor compared to a 24-T FF	31	15	7	11	8	1.46	**2**
Datapath Metastability	No	No	No	No	Yes	No	**No**
FF Area Overhead	-	-	-	-	33%	4.3%	8.6%
NTV Operation	Yes	Yes	Yes	NO	No	No	**Yes**

Thus, we choose to stall the global clock for one cycle by clock gating to ensure that data can be captured in the next cycle when PoFF occurs. When a spontaneous error occurs during VDD decreasing, clock gating is enabled at the negative clock. When two errors happen in a short period, it is considered as PoFF so that the AVFS system will stop decreasing VDD.

In the fabricated chip, among the total 4897 FFs, 607 endpoints FFs are replaced by our proposed EDLs in those critical paths with small slack time under different PVT conditions. For example, paths with a slack less than 3 ns of a 17-ns clock cycle (that is, 17.6% of the clock cycle) are selected at 0.55 V, typical-typical (TT), 25 °C, and some more critical paths are added for other PVT conditions. This results in a 12.4% replacement rate. The CTG insertion number for SPP is 2106, which is extremely small considering that more than 20 000 padding buffers are used by the buffer-based padding method. To see the area and power overhead on the clock tree, the number of buffers/inverters added to the clock tree due to CTG is only 556, which is about 5.71% of the total clock-tree buffers/inverters. The total core area overhead caused by EDLs, dynamic ORs, and CTGs is 8.15%.

Our EDAC technique needs a special design flow as described in the following. After logic synthesis and timing analysis, critical and near-critical paths will be identified. Next, the selected endpoint registers are replaced by EDLs in RTL and the design is synthesized again. Then, CTGs are inserted for SPP by customized automation scripts. Then, after placement and routing, engineering change order (ECO) is performed to insert more EDLs in case new critical paths are generated after layout.

We solve the SP issue by inserting CTGs in paths to make them behave like two-phase-latch paths but without the need for complex retiming. Instead, we replace the registers in critical paths by our EDLs in the normal EDAC ways. Thus, our design complexity as well as the circuit overhead are reduced.

V. MEASUREMENT RESULTS

Our chip is fabricated with the foundry test circuits together and measured in the foundry directly on the wafer mounted on the probe card. The testing platform and probe card on a wafer are shown in Fig. 16(a) and (b), respectively. The benefit of this fabrication and direct testing is that the foundry provides multiple wafers with selected process variations of SS, TT, and FF, which are different from multi project wafer (MPW) chips

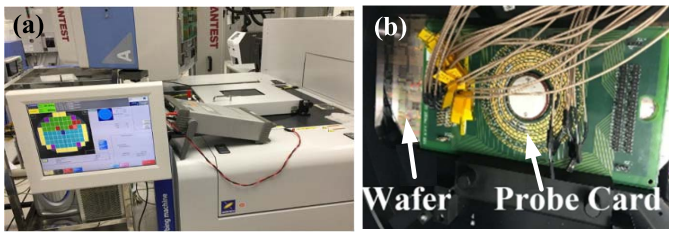


Fig. 16. Chip measurement environment. (a) Testing platform in the foundry. (b) Probe card on a wafer.

that are around the TT corner. Therefore, we obtain chips that represent realistic process variations validated by the foundry. Thus, we are able to obtain realistic power/performance gain of our resilient chip.

To measure the power/performance gain of our resilient chip, first, we measure the baseline frequencies under each supply voltages. The baseline is set to be measured at the worst case conditions, which include a 10% VDD drop, the slowest process corner, and the worst temperature (85 °C at the STV and −20 °C at NTV due to a temperature reverse effect).

For example, when operating at the NTV region of 0.55 V, the baseline frequency is defined as the measured Fmax of the slowest chip at 0.5 V and −20 °C. The measured baselines across the wide voltage range are shown as the black line in Fig. 17(b), which are 24 MHz for NTV and 258.75 MHz for STV. The results also demonstrate that the proposed TG-SPP solves the SP issue properly and its frequency operating range is as wide as the flop-based design, with no erroneous timing.

The resilient chip obtains remarkable power savings or performance gains across a wide voltage range, especially when working at the near-threshold region. To show how much power or performance gains can be obtained due to error-detection adaptive voltage/frequency scaling, we first measure a typical die enabled with across a wide VDD range of 0.55–1.0 V at room temperature, as shown in Fig. 17.

The baseline power/frequency (blue line with triangle dots) is measured under the worst case, while the resilient chip (black line with square dots) is measured when error detection is enabled. Fig. 17(a) shows the power consumption of our resilient chip versus the margined baseline, where our voltage scaling based on timing monitoring reduces the supply voltage to be lower than the baseline voltage, thus consuming less power. It can be seen that at the STV region, it gains 28.74% power saving over the baseline when operating at the same

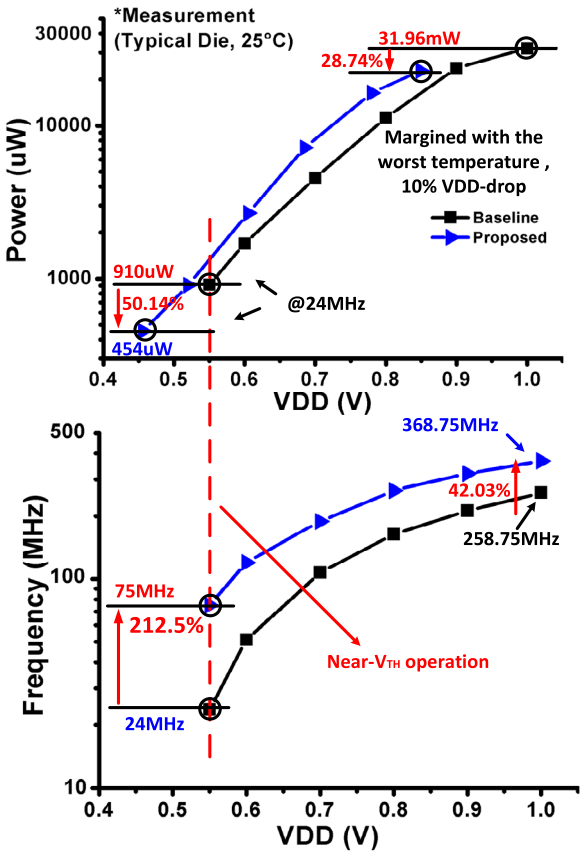


Fig. 17. Measured results of a typical die across VDD of 0.55–1.0 V. (a) Power consumptions and the related power gains of the resilient chip and baseline. (b) Maximum operating frequencies and frequency gains of the resilient chip and baseline.

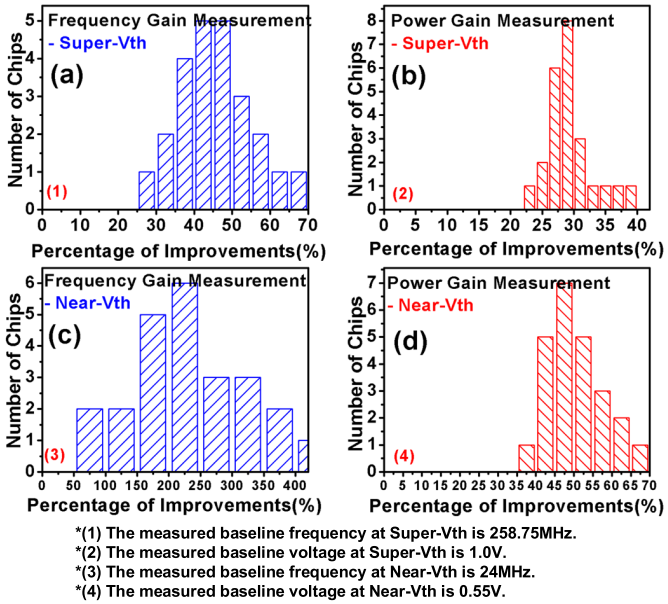


Fig. 18. Measured power and frequency gains of 24 dies at both NTV and STV. Frequency gain at (a) STV and (c) NTV. Power gain at (b) STV and (d) NTV.

frequency, whereas at the NTV region, it gains 50.14% power saving over the 0.55-V baseline. Fig. 17(b) shows the operating frequencies of the resilient chip and baseline frequencies when working at the same supply voltage, where we see a performance gain of 42% (368.75 MHz over 258.75 MHz) at the STV region and 3.125× (75 MHz over 24 MHz) at the NTV region for a typical die. Both power gains and performance gains are higher at NTV than STV because of the large timing margins due to severe PVT variations at NTV.

We also evaluate the overall power and frequency gains at both NTV and STV by measuring 24 dies from 6 representative wafers with process variations from SS to FF. Their frequency-gain and power-gain distributions at STV are shown in Fig. 18(a) and (b) when measured at 1.0 V, 25 °C. The frequency gains are 24.5%–70% and power gains are 22.5%–42% at STV. The near-threshold gains are shown in Fig. 18(c) and (d), where we can see 55%–405% frequency improvement or 38.6%–69.4% of power savings when compared to baseline chips working at the fixed 0.55 V. Our frequency gains are higher than the state of the art because of two reasons. First, the severe variation at NTV makes the worst case frequency of the slowest die quite low (only 24 MHz). Second, as opposed to the MPW chips with limited process variations reported in other works, our dies are fabricated along with foundry testing circuits; thus, the provided SS, TT, and FF wafers are with realistic variations, including die-to-die variations.

In order to evaluate the effectiveness of our proposed method in reducing power or improving performance, we compare our work with the state-of-the-art EDAC techniques, as shown in Table IV. These techniques used different implementations. For example, whether they are FF based or latch-based EDAC strategies, near-Vth operations are enabled or not, different SPP solutions, different EDAC insertion rates, and so on [9]–[12], [28]. Thus, it will not be fair to directly compare their area overheads and adaptive voltage scaling (AVS) power savings since they are affected by multiple factors such as different timing monitors, processes, benchmark circuits, insertion rates, and detection window sizes.

Detailed comparisons and analyses are given as follows.

1) Compared to flop-based designs of Razor II [19] and Razor-lite [9], which used a buffer with/without duty-cycle control to solve the SP issue, our chip is able to work at near threshold, whereas they could not. The NTV operation causes more severe variations than the normal VDD applications, which increase the paths to be monitored as well as the SP issue. Thus, our area overhead may be larger compared to non-NTV resilient circuits.
2) Our application circuit has more SPs compared to other microprocessor-based designs [9]–[12], [28], as explained in Section III-D. Our EDL insertion rate is 12.4%, which is higher than other designs [25], [28]. Compared to the NTV-enabled designs of Tw tuning [25] and TEM [28], they have much lower insertion rates (5.5% and 5.7%) and a little smaller area overhead (7.3% and 7%) than ours. Since our previous work was implemented on the same SHA-256 circuit, we redesign Tw tuning [25] resilient circuit in the same 28-nm process and insert the endpoint timing monitors with

TABLE IV
COMPARISONS OF OUR EDAC SYSTEM AND PREVIOUS EDAC WORKS

	Razor II [19]	**Razor-lite [9]**	**Tw tuning [25]**	**TEM [28]**	**iRazor [10]**	**Bubble Razor [11]**	**R Processor [12]**	**This work (TG-SPP)**
Technology	0.13um	45nm SOI	40nm	40nm	40nm	45nm SOI	65nm	28nm
NTV enabled	Yes	No	Yes	Yes	No	No	Yes	**Yes**
Sequencing	Flip-Flop	Flip-Flop	Flip-Flop	Flip-Flop	Latch	Latch	Latch	Latch
Short path solution	Buffer	Duty-cycle Control+buffer	No**	NA	Buffer+Pulse Generator	Two-phase latch	Two-phase latch+ sparse insertion	**CTG insertion**
EDAC Extra transistors*	31	8	48	46	1.46	20+dynOR +cluster	24	2
Metastability	Yes	Yes	No	No	Yes	No	No	**No**
EDAC Insertion rate	14.64% (121/826)	19.82% (492/2482)	5.5%	5.7% (224/3913)	8.67% (1115/12875)	100%	13%	12.4% (607/4897)
Area overhead	NA	4.42%	7.3%	7%	13.6%	103%	8.3%	8.15%
Typical freq gain	NA	83%	190.8%***	NA	34%	103%	130%	**212.5%**
Typical Power gain	33%	45.4%	33.3%	26%	41%	54%	38%	50.14%

* Compared to a standard 24-T flip-flop;
**No short path issue due to the use of two parallel FFs as the timing monitor;
*** Compared to signoff frequency while others compared to measured baseline frequency.

the same insertion rate of 12.4% as in this article. After synthesis and Primetime simulation, we obtain its overhead as 9.9%, which is higher than 8.15% in this article. As for TEM [28], since it has similar EDAC transistors (46 vs. 48) and a similar insertion rate (5.7% vs. 5.5%) as in [25], it is reasonable to estimate that its area overhead may increase to a similar level when its insertion rate is increased to the same level as in this article.

Since it is not affected by threshold loss, the proposed technique is well suited for the voltage-scalable resilient design from NTV to normal VDD. As compared with Razor-lite, iRazor, and Bubble Razor, our proposed TG-SPP efficiently reduces the area overhead brought by SPP. In addition, we do not need a duty-cycle-correcting circuit or a carefully controlled clock tree as used in other latch-based resilient circuits. Our EDL circuit also avoids the possible data path metastability problem that occurred in the two-phase latch-based design. With an acceptable area overhead, we obtain the near-Vth EDAC operation with a relatively large power saving and the best frequency gain.

VI. CONCLUSION

In order to solve the SP issue in the resilient circuit, we propose a CTG-based method, which extends SPs with only one CTG gate in a path. Our resilient circuit incorporates the benefits of both single latch-based and two-phase latch-based techniques, in which it solves the SP issue as the two-phase latch-based technique while having a low replacement ratio and small sequential overhead as the single latch-based error detection. Our CTG-based SPP method can be plugged into circuits as a universal method because the insertion of CTGs is based on STA results and independent of the circuit function. However, its effectiveness in reducing the area overhead due to SPP depends on how severe the SP issue is.

Applied to an SHA-256 algorithm circuit, the proposed method substantially reduces the combinational/sequential area overhead. It also reduces the invalid flipping of combinational logic and thus saves glitch power. Furthermore, we design a lightweight EDL circuit and apply it with our proposed insertion mechanism on the SHA-256 test chip in a 28-nm CMOS process. Our chip measurement results demonstrate that our EDAC technique solves the difficulty of using a limited resource to extend SPs in the resilient circuit and has better gains in performance/power as compared to the margined baseline.

ACKNOWLEDGMENT

The authors would like to thank Semiconductor Manufacturing International Corporation (SMIC) for providing the fabrication and the wafer measurement and ProPlus Electronics for providing Nanospice tool for yield analysis.

REFERENCES

[1] K. A. Bowman, S. G. Duvall, and J. D. Meindl, "Impact of die-to-die and within-die parameter fluctuations on the maximum clock frequency distribution for gigascale integration," *IEEE J. Solid-State Circuits*, vol. 37, no. 2, pp. 183–190, Feb. 2002.

[2] S. Borkar, T. Karnik, S. Narendra, J. Tschanz, A. Keshavarzi, and V. De, "Parameter variations and impact on circuits and microarchitecture," in *Proc. DAC*, Jun. 2003, pp. 338–342.

[3] M. Seok, G. Chen, S. Hanson, M. Wieckowski, D. Blaauw, and D. Sylvester, "CAS-FEST 2010: Mitigating variability in near-threshold computing," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 1, no. 1, pp. 42–49, Mar. 2011.

[4] K. A. Bowman *et al.*, "Energy-efficient and metastability-immune resilient circuits for dynamic variation tolerance," *IEEE J. Solid-State Circuits*, vol. 44, no. 1, pp. 49–63, Jan. 2009.

[5] D. Bull, S. Das, K. Shivashankar, G. S. Dasika, K. Flautner, and D. Blaauw, "A power-efficient 32 bit ARM processor using timing-error detection and correction for transient-error tolerance and adaptation to PVT variation," *IEEE J. Solid-State Circuits*, vol. 46, no. 1, pp. 18–31, Jan. 2011.

[6] Y.-M. Yang, I. H.-R. Jiang, and S.-T. Ho, "PushPull: Short-path padding for timing error resilient circuits," *IEEE Trans. Comput.-Aided Design Integr.*, vol. 33, no. 4, pp. 558–570, Apr. 2014.

[7] W. Dai, W. Shan, X. Shang, X. Liu, H. Cai, and J. Yang, "HTD: A light-weight holosymmetrical transition detector for wide-voltage-range variation resilient ICs," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 65, no. 11, pp. 3907–3917, Nov. 2018.

[8] J. Zhou *et al.*, "HEPP: A new in-situ timing-error prediction and prevention technique for variation-tolerant ultra-low-voltage designs," in *Proc. IEEE Asian Solid-State Circuits Conf. (A-SSCC)*, Singapore, Nov. 2013, pp. 129–132.

[9] S. Kim, I. Kwon, D. Fick, M. Kim, Y. P. Chen, and D. Sylvester, "Razor-lite: A side-channel error-detection register for timing-margin recovery in 45nm SOI CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2013, pp. 264–265.

[10] Y. Zhang *et al.*, "iRazor: 3-transistor current-based error detection and correction in an ARM Cortex-R4 processor," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2016, pp. 160–162.

[11] M. Fojtik *et al.*, "Bubble razor: Eliminating timing margins in an arm cortex-m3 processor in 45 nm CMOS using architecturally independent error detection and correction," *IEEE J. Solid-State Circuits*, vol. 48, no. 1, pp. 66–81, Jan. 2013.

[12] S. Kim and M. Seok, "Variation-tolerant, ultra-low-voltage microprocessor with a low-overhead, within-a-cycle in-situ timing-error detection and correction technique," *IEEE J. Solid-State Circuits*, vol. 50, no. 6, pp. 1478–1490, Jun. 2015.

[13] A. Rahimi, L. Benini, and R. K. Gupta, "Variability mitigation in nanometer CMOS integrated systems: A survey of techniques from circuits to software," in *Proc. IEEE*, vol. 104, no. 7, pp. 1410–1448, Jul. 2016.

[14] K. Bowman *et al.*, "Dynamic variation monitor for measuring the impact of voltage droops on microprocessor clock frequency," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, San Jose, CA, USA, Sep. 2010, pp. 1–4.

[15] M. Eireiner, S. Henzler, G. Georgakos, J. Berthold, and D. Schmitt-Landsiedel, "*In-Situ* delay characterization and local supply voltage adjustment for compensation of local parametric variations," *IEEE J. Solid-State Circuits*, vol. 42, no. 7, pp. 1583–1592, Jul. 2007.

[16] C. Wang *et al.*, "Near-threshold energy- and area-efficient reconfigurable DWPT/DWT processor for healthcare-monitoring applications," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 62, no. 1, pp. 70–74, Jan. 2015.

[17] Z.-H. Li, T.-T. Zhu, Z.-J. Chen, J.-Y. Meng, X.-Y. Xiang, and X.-L. Yan, "Eliminating timing errors through collaborative design to maximize the throughput," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 2, pp. 670–682, Feb. 2017.

[18] S. Das *et al.*, "A self-tuning DVS processor using delay-error detection and correction," in *Dig. Tech. Papers Symp. VLSI Circuits*, Jun. 2015, pp. 258–261.

[19] S. Das *et al.*, "RazorII: In situ error detection and correction for PVT and SER tolerance," *IEEE J. Solid-State Circuits*, vol. 44, no. 1, pp. 32–48, Jan. 2009.

[20] K. A. Bowman *et al.*, "A 45 nm resilient microprocessor core for dynamic variation tolerance," *IEEE J. Solid-State Circuits*, vol. 46, no. 1, pp. 194–208, Jan. 2011.

[21] P. N. Whatmough, S. Das, and D. M. Bull, "A low-power 1-GHz razor FIR accelerator with time-borrow tracking pipeline and approximate error correction in 65-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 49, no. 1, pp. 84–94, Jan. 2014.

[22] S. Kim, J. P. Cerqueira, and M. Seok, "A 450mV timing-margin-free waveform sorter based on body swapping error correction," in *Proc. IEEE Symp. VLSI Circuits (VLIC)*, Jun. 2016, pp. 1–2.

[23] J.-S. Wang and S.-N. Wei, "Process/voltage/temperature-variation-aware design and comparative study of transition-detector-based error-detecting latches for timing-error-resilient pipelined systems," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 10, pp. 2893–2906, Oct. 2017.

[24] W. Jin, S. Kim, W. He, Z. Mao, and M. Seok, "*In Situ* error detection techniques in ultralow voltage pipelines: Analysis and optimizations," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 3, pp. 1032–1043, Mar. 2017.

[25] W. Shan, X. Shang, L. Shi, W. Dai, and J. Yang, "Timing error prediction AVFS with detection window tuning for wide-operating-range ICs," *IEEE Trans. Circuits Syst., II, Exp. Briefs*, vol. 65, no. 7, pp. 933–937, Jul. 2018.

[26] C. Wang, J. Luo, and J. Zhou, "A 1-V to 0.29-V sub-100-pJ/operation ultra-low power fast-convergence CORDIC processor in 0.18-μm CMOS," *Microelectron. J.*, vol. 76, pp. 52–62, Jun. 2018.

[27] W. Dai, P. Liu, and W. Shan, "Short-path padding method for timing error resilient circuits based on transmission gates insertion," in *Proc. Great Lakes Symp. VLSI (GLSVLSI)*, May 2018, pp. 105–110.

[28] H. Reyserhove and W. Dehaene, "Margin elimination through timing error detection in a near-threshold enabled 32-bit microcontroller in 40-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 53, no. 7, pp. 2101–2113, Jul. 2018.

[29] M. Seok, D. Jeon, C. Chakrabarti, D. Blaauw, and D. Sylvester, "Pipeline strategy for improving optimal energy efficiency in ultra-low voltage design," in *Proc. 48th ACM/EDAC/IEEE Design Automat. Conf. (DAC)*, Jun. 2011, pp. 990–995.

[30] D. Bol, D. Flandre, and J. Legat, "Technology flavor selection and adaptive techniques for timing-constrained 45nm subthreshold circuits," in *Proc. ACM/IEEE Int. Symp. Low-Power Electron. Design*, Aug. 2009, pp. 21–26.

[31] F. Botman, D. Bol, J.-D. Legat, and K. Roy, "Data-dependent operation speed-up through automatically inserted signal transition detectors for ultralow voltage logic circuits," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 22, no. 12, pp. 2561–2570, Dec. 2014.

[32] D. Bol *et al.*, "SleepWalker: A 25-MHz 0.4-V Sub-mm^2 7-μW/MHz microcontroller in 65-nm LP/GP CMOS for low-carbon wireless sensor nodes," *IEEE J. Solid-State Circuits*, vol. 48, no. 1, pp. 20–32, Jan. 2013.

[33] K. A. Bowman, "Adaptive and resilient circuits: A tutorial on improving processor performance, energy efficiency, and yield via dynamic variation," *IEEE Solid State Circuits Mag.*, vol. 10, no. 3, pp. 16–25, Aug. 2018.

[34] 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, Jul. 2013.

[35] D. Ernst *et al.*, "Razor: A low-power pipeline based on circuit-level timing speculation," in *Proc. 36th Annu. IEEE/ACM Int. Symp. Microarchitecture (MICRO)*, Dec. 2003, p. 7.

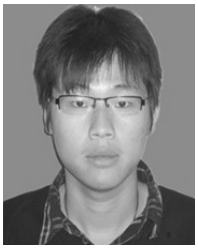
[36] L. Marco, R. De Rose, F. Frustaci, S. Perri, and P. Corsonello, "Comparative analysis of yield optimized pulsed flip-flops," *Microelectron. Rel.*, vol. 52, no. 8, pp. 1679–1689, Aug. 2012.

[37] H. Mostafa, M. Anis, and M. Elmasry, "Comparative analysis of timing yield improvement under process variations of flip-flops circuits," in *Proc. IEEE Comput. Soc. Annu. Symp. VLSI*, May 2009, pp. 133–138.

Weiwei Shan (M'09) received the B.S. degree in microelectronics from Tianjin University, Tianjin, China, in 2003, and the Ph.D. degree in microelectronics from Tsinghua University, Beijing, China, in January 2009.

From 2018 to 2019, she was a Visiting Professor with Columbia University, New York, NY, USA. She is an Associate Professor with the National ASIC Center, Southeast University, Nanjing, China. She has authored or coauthored more than 50 technical articles in conferences and journals and authorized over 25 invention patents. Her research mainly focuses on variation resilient adaptive very large-scale integration (VLSI) circuits, ultra-low power system-on-a-chip (SoC) design, and countermeasure techniques of security circuits.

Dr. Shan was a recipient of the State Scientific and Technological Progress Award in 2014, the A-SSCC Distinguished Design Award in 2017, and the GLSVLSI Best Paper Candidate in 2018.

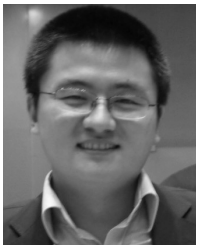


Wentao Dai (S'17) received the B.S. degrees in microelectronics from Yangzhou University, Nanjing, China, in 2013. He is currently pursuing the Ph.D. degree with the National ASIC Center, Southeast University, Nanjing.

His current research focuses on static random-access memory (SRAM) design, in-memory computing, and near-threshold design.

Chuan Zhang (S'07–M'13) received the B.E. degree (*summa cum laude*) in microelectronics and the M.E. degree (Hons.) in very large-scale integration (VLSI) design from Nanjing University, Nanjing, China, in 2006 and 2009, respectively, and the M.S.E.E. and Ph.D. degrees from the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN, USA, in 2012.

He is currently the Excellence Professor and the Purple Mountain Professor with Southeast University, Nanjing. He is also with the LEDAS, National Mobile Communications Research Laboratory, Quantum Information Center, Southeast University, and with the Purple Mountain Laboratories, Nanjing. His current research interests include low-power high-speed VLSI design for digital signal processing and digital communication, bio-chemical computation and neuromorphic engineering, and quantum communication.



Hao Cai (M'15) received the Ph.D. degree in electrical engineering from Télécom Paristech, Université Paris-Saclay, Paris, France, in 2013.

From 2012 to 2014, he was involved in the European EUREKA Program CATRENE-RELY for high-reliability nanoscale integrated circuits and systems. He is currently a Faculty Member with the National ASIC System Engineering Center, Southeast University, Nanjing, China. His current research interests include circuit techniques for emerging technologies, ultralow-power very large-scale integration (VLSI), and reliability-aware design.

Peiye Liu received the B.S. degree from the Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2015, where he is currently pursuing the Ph.D. degree with the Beijing Key Laboratory of Intelligent Telecommunication Software and Multimedia.

Now, his research interests are about deep learning/machine learning and its applications in computer vision.

Jun Yang (M'13) received the B.S. and Ph.D. degrees in electronic engineering from Southeast University, Nanjing, China, in 1999 and 2004, respectively.

He is currently a Professor with the National ASIC System Engineering Research Center, Southeast University. His current research focuses on static random-access memory (SRAM) design, in-memory computing, and near-threshold design.

Longxing Shi (M'09–SM'18) received the Ph.D. degree in microelectronics from Southeast University, Nanjing, China, in 1988.

He is currently a Professor and the Dean of the Electrical Science and Technology School, Southeast University. His research interests include system-on-chip design and RF and mixed-signal integrated circuit design. He has authored or coauthored more than 120 technical articles in conferences and journals. He holds more than 200 Chinese invention patents and 12 USA patents.

Dr. Shi was a recipient of several State Scientific and Technological Progress Awards.