Design Methodology for Voltage-Scaled Clock Distribution Networks

Can Sitik, *Student Member, IEEE*, Weicheng Liu, Baris Taskin, *Senior Member, IEEE*, and Emre Salman, *Member, IEEE*

***Abstract*— A low-voltage/swing clocking methodology is developed through both circuit and algorithmic innovations. The primary objective is to significantly reduce the power consumed by the clock network while maintaining the circuit performance the same. The methodology consists of two primary components: 1) a novel D-flip-flop (DFF) cell that maximizes power savings by enabling low-voltage/swing operation throughout the entire clock network and 2) a novel clock tree synthesis algorithm to ensure that the same timing constraints (i.e., clock frequency, skew, and slew) are satisfied. The proposed methodology is integrated within an industrial design flow. Experimental results on ISCAS'89 benchmark circuits demonstrate that the overall power consumed by the clock tree can be reduced by up to 27% and 44% in, respectively, 32- and 45-nm technologies while satisfying the same timing constraints. Furthermore, the proposed low-swing DFF cell maintains the clock-to-Q delay the same while achieving up to 32% and 15% power savings in the overall flip-flop power of the benchmark circuits at, respectively, 1- and 1.5-GHz clock frequencies.**

***Index Terms*— Clock network, clock tree synthesis (CTS), D-flip-flop (DFF), low power, low-voltage/swing clocking.**

I. Introduction

POWER consumption has become one of the primary concerns for almost any application due to increased design complexity, higher integration, and difficulty in scaling the power supply voltage [1]. A clock distribution network consumes ∼20%–45% of total on-chip power, and ∼90% of this power is consumed by the flip-flops and last branches of a clock tree [2], [3]. This power dissipation is the result of increased pipelining in an IC, which has led to an increase in the number of flip-flops, and hence the total interconnect length of the clock network [4]. A well-known approach to minimize the overall on-chip power dissipation is to reduce the supply voltage [5]. For example, near-threshold computing has received considerable attention to achieve optimal energy efficiency [6]. A reduction in supply voltage, however, degrades IC performance, particularly when the nominal supply voltages are low [5]. Low-swing (LS) signaling has also been investigated to reduce dynamic power consumed by long interconnects [7]. This approach has been extended to clock networks due to high clock net capacitance [8]–[11]. Clock networks operating at near-threshold voltages have also been investigated [12]–[14].

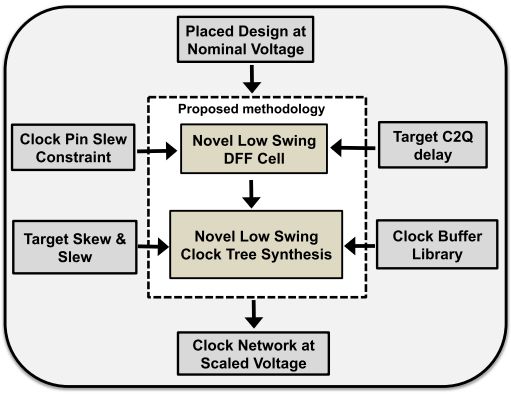


Fig. 1. Summary of the proposed methodology to achieve LS clocking while maintaining the performance requirements.

The existing works on LS/voltage clock networks, however, are primarily effective for low-power applications that do not demand high performance. Achieving a reliable LS clock network without sacrificing performance is challenging due to the following issues: 1) clock buffers operating at a lower voltage increase the insertion delay along the clock path, causing higher clock skew; 2) the drive ability of the clock buffers is degraded, producing higher clock slew; and 3) the interface between an LS clock signal and a flip-flop may increase clock-to-Q delay, thereby reducing the timing slack within the data paths while also increasing power consumption. To alleviate the first two issues, a larger number of clock buffers are required, which sacrifice the power savings. To alleviate the third issue, a common approach is to restore full-swing (FS) operation before the clock signal reaches flip-flops [8], [9]. This approach significantly reduces power savings, since the last stage of a clock network has high switching capacitance.

In this paper, these three primary issues are simultaneously addressed through both circuit and algorithmic innovations, making LS clocking a practical power reduction strategy for both low-power and high-performance applications. Furthermore, the proposed methodology is implemented within a standard design flow for feasible integration into the existing automation tools. As shown in Fig. 1, the methodology

Manuscript received September 30, 2015; revised January 12, 2016 and February 12, 2016; accepted February 12, 2016. Date of publication March 25, 2016; date of current version September 23, 2016. This work was supported by Semiconductor Research Corporation under Grant 2013-TJ-2449 and Grant 2013-TJ-2450.

C. Sitik and B. Taskin are with the Department of Electrical and Computer Engineering, Drexel University, Philadelphia, PA 19104 USA (e-mail: cansitik@outlook.com; taskin@coe.drexel.edu).

W. Liu and E. Salman are with the Department of Electrical and Computer Engineering, Stony Brook University, Stony Brook, NY 11794 USA (e-mail: weicheng.liu@stonybrook.edu; emre.salman@stonybrook.edu).

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

Digital Object Identifier 10.1109/TVLSI.2016.2539926

1063-8210 © 2016 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

consists of a novel D-flip-flop (DFF) cell and a novel clock tree synthesis (CTS) methodology. The proposed DFF cell enables reliable LS operation at the clock sinks while maintaining the timing constraints the same. The proposed CTS algorithm ensures that the same skew and slew constraints as in FS operation are satisfied. In particular, the proposed methodology has the following characteristics.

1) The proposed LSDFF cell achieves similar clock-to-Q delay as traditional FSDFF topology while consuming less power. Reliable operation is ensured despite an LS clock signal and an FS data signal.
2) The proposed slew-aware LS CTS methodology considers not only capacitance, but also resistance to efficiently utilize clock tree resources at different performance constraints and transistor/interconnect technologies.
3) The increase in the insertion delay due to LS operation on the clock tree is methodically compensated by embedding a buffer insertion/wire snaking scheme within the CTS for skew minimization.

The output of the proposed methodology is an LS clock tree, running at the same frequency, satisfying the same clock skew and clock slew constraints as the conventional FS operation while saving significant power.

The rest of this paper is organized as follows. The existing approaches on LS/voltage clocking and related limitations are summarized in Section II. Background information on the effect of voltage level on flip-flop operation and clock signal characteristics is provided in Section III. The proposed methodology is described in Section IV. Experimental results are presented in Section V. Finally, this paper is concluded in Section VI.

II. Summary of Previous Work

Pangjun and Sapatnekar [8] developed a low-voltage/swing clock network by utilizing level converters. Both single voltage and dual voltage converters were considered. A theoretical framework was proposed to appropriately position the low-to-high level converters throughout the clock tree. The primary limitation of this approach is the conversion of the clock signal back to FS at the last stage of the clock tree. This practice significantly reduces the power savings due to high switching capacitance at the sink nodes. In addition, the slew constraint is considered as a secondary design objective after the merging points are determined during CTS. As observed in this paper, this approach generates a nonoptimal LS clock tree with reduced power savings.

Asgari and Sachdev [9] proposed an LS clock network design methodology using a single supply voltage. In this approach, single voltage buffers are used to adjust the clock swing throughout the clock network. Similar to [8], clock voltage is restored to FS at the last stage, thereby significantly reducing the overall power savings. In addition, the clock swing is tuned by relying on the delay of an inverter chain. Thus, the clock swing is highly dependent upon the output load capacitance, limiting the proposed approach to only highly symmetric clock networks, such as H-trees.

More recently, low voltage clock networks have been investigated for near-threshold systems that aim enhanced energy efficiency. Seok *et al.* [13] investigated the skew characteristics of various clock networks operating at low voltages. The primary emphasis is on symmetric networks, such as H-trees. Automated CTS algorithms were not considered. Tolbert *et al.* [12] and Zhao *et al.* [14] proposed a deferred merge embedding (DME)-based CTS method for low-voltage clock networks with an emphasis on clock slew. The proposed technique relies on a computationally expensive procedure of storing multiple solutions in a bottom–up fashion, followed by selecting an optimum solution for each node in a top–down fashion. Clock frequencies of <10 MHz are considered, limiting the proposed approach to only ultralow-power systems, where performance requirements are low.

Recently, a preliminary CTS algorithm and a flip-flop have been proposed for LS operation [15], [16] that target high performance. The authors, however, considered a constant reduced voltage for the clock network. The effects of different clock voltages, interconnect parasitic impedances, process corners, and technology node have not been investigated. Furthermore, the results rely on schematic-based simulations with coarse interconnect models. The limitations of the methodology, such as area overhead, have not been quantified.

The proposed methodology in this paper alleviates all of these issues while targeting low-power systems that still require high-performance capability. Furthermore, the methodology is demonstrated using the existing design automation tools for feasible integration into typical design flows.

III. Background and Motivation

Scaling the supply voltage of an entire clock network affects both the sink cells (flip-flops) and the clock tree characteristics. The effects of clock swing on the conventional DFF cells and timing characteristics of clock trees are described, respectively, in Sections III-A and III-B.

A. Effect of Low-Swing Operation on Flip-Flop

A conventional DFF cell designed for FS operation cannot be used when the clock voltage swing is reduced due to degradations in reliability and power consumption, as described in Sections III-A1 and III-A2.

1) Reliability: In a typical DFF cell, clock signals drive both nMOS and pMOS transistors. If the same DFF topology is used with an LS clock signal (whereas the data signal is still at FS to maintain performance), the pMOS transistors driven by the clock signal fail to completely turn OFF when the clock signal is high. For example, consider a 45-nm technology with a nominal V_{DD} of 1 V. If the clock swing is reduced to $0.7 \times V_{DD}$, the gate-to-source voltage of the pMOS transistors is -0.3 V, since the data signal is at FS and the inverters within the flip-flop are connected to nominal (FS) V_{DD}. Since -0.3 V is sufficiently close to the threshold voltage of pMOS transistors in this technology, this behavior significantly affects the operation reliability of a traditional DFF cell driven by an LS clock signal. As an example, consider a rising-edge triggered master–slave flip-flop. When the clock signal is high, the master latch should be turned OFF. However, due to the LS clock signal, the transmission gate (or tristate inverter)

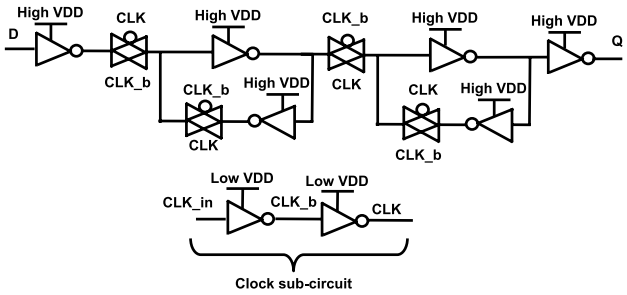


Fig. 2. Typical transmission gate-based DFF topology driven by an LS clock signal.

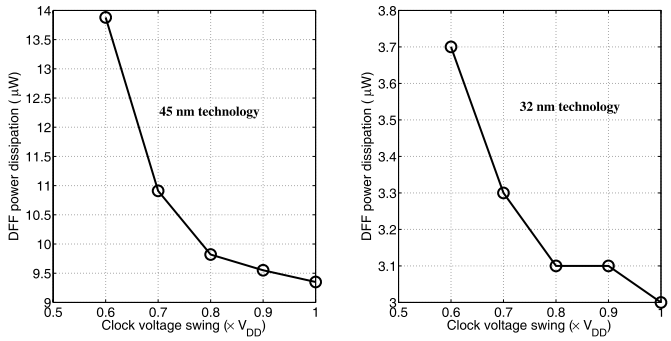


Fig. 3. Increase in power consumption when a conventional DFF is driven with an LS clock signal while the clock subcircuit is connected to a nominal V_{DD}.

within the master latch cannot completely turn OFF. If the data signal is in a different state than the stored data within the master latch, a race condition occurs, which can produce a metastable state.

To better illustrate the unreliability of the conventional DFF cells operating with an LS clock signal, a traditional transmission gate-based DFF, as shown in Fig. 2, is simulated with a 45-nm technology node when the clock swing is 0.7 V. Note that the clock signal and the inverted clock signal are internally generated using two inverters. This circuit is referred to as the clock subcircuit, as also shown in Fig. 2. Note that the inverters within the clock subcircuit are connected to a low supply voltage to provide LS clock signals. Since the pMOS transistors driven by the clock signals are not completely turned OFF, internal nodes experience a glitch as high as 400 mV. Furthermore, at the slow corner, the DFF cell fails to correctly latch the data signal. Thus, a new topology is required that can reliably operate with an LS clock signal and an FS data signal.

Note that an alternative solution is to integrate a level shifter within the DFF cell to restore an FS clock signal [17]. Thus, the clock signal is restored to FS operation before reaching transmission gates. This approach is similar to the existing level shifting DFF cells for dual voltage systems [18], but the level of the clock signal is shifted rather than the data signal. This approach, however, significantly increases the overall power consumption of the DFF cell due to the integrated level shifter. Thus, the power saved at the last stage of the clock network is lost within the DFFs, making this approach impractical for the primary objective of this paper.

2) Power Consumption: The reliability issue described in Section III-A1 can be fixed by connecting the inverters within the clock subcircuit of a conventional DFF to the nominal V_{DD}, producing a single voltage flip-flop driven by an LS clock signal. In this case, these inverters also function as single voltage, low-to-high level shifters, and the transmission gates receive FS clock signals. The primary limitation of this approach is an unavoidable increase in power consumption due to significant static current drawn by the inverters within the clock subcircuit. To better illustrate this behavior, a conventional DFF is simulated when an LS clock signal is applied to the clock pin while the clock subcircuit is connected to a nominal V_{DD}. The overall power consumption is shown in Fig. 3 as a function of clock swing for both 45- and 32-nm technologies. As shown in Fig. 3, DFF power increases by ∼48% and 23% when the clock swing is reduced to $0.6 \times V_{\mathrm{DD}}$ in, respectively, 45- and 32-nm technologies. Thus, a conventional flip-flop designed for an FS clock signal suffers from a prohibiting tradeoff between reliability and power consumption.

B. Effect of Low-Swing Operation on Clock Tree

When developing an LS clock tree, the clock buffers throughout the network are connected to a low supply voltage. A lower power supply voltage on the clock buffers increases the clock path latency (insertion delay) and the switching time (slew) of the clock signals propagating throughout the clock network. An increase in the clock latency potentially increases the clock skew, whereas an increase in the clock slew degrades the clock-to-Q delay of the DFF cells. Fixing these skew and slew issues typically requires buffer insertion, which in turn increases the overall power consumption and, therefore, reduces the power savings obtained through LS clocking. Thus, the implications of voltage scaling on timing and power characteristics are investigated in this section.

To emphasize the requirement for a low-swing-aware CTS methodology, a sample clock tree is synthesized for the largest circuit (s35932) of ISCAS'89 benchmarks. Four different cases are generated by combining two interconnect technologies (wire 1 [19]: $R = 2\ \Omega/\mu$m and $C = 0.1$ fF/μm, and wire 2 [20]: $R = 8\ \Omega/\mu$m and $C = 0.2$ fF/μm) with two frequency/slew constraints.

Case 1: Wire 1, 1 GHz, 150-ps slew constraint.
Case 2: Wire 2, 1 GHz, 150-ps slew constraint.
Case 3: Wire 1, 1.5 GHz, 100-ps slew constraint.
Case 4: Wire 2, 1.5 GHz, 100-ps slew constraint.

The experiments are performed at the slow process–voltage–temperature (PVT) corner of two transistor technologies (slow-slow (SS), 0.95 V, 125 °C for 32-nm SAED and SS, 0.9 V, 125 °C for 45-nm FreePDK). The power supply voltage is scaled to 65% of the nominal value with 5% decrements to observe the effect of voltage scaling.

1) Clock Skew: The effect of voltage scaling on clock skew is shown in Figs. 4 and 5. According to Fig. 4, in a 45-nm technology, the clock skew slightly increases when the clock voltage is scaled. The increase in clock skew does not introduce a violation, assuming a 50-ps skew constraint. In a 32-nm technology, however, the clock skew increases as clock voltage is reduced, reaching and even exceeding the

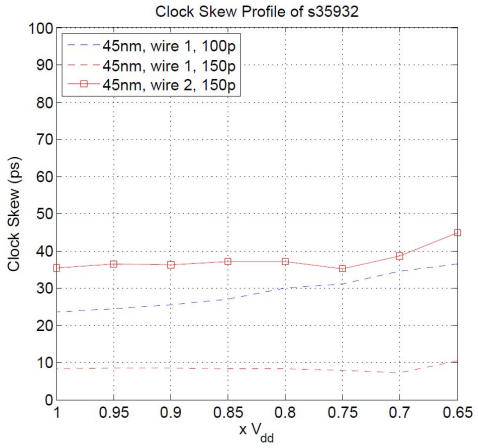


Fig. 4. Clock skew profile of s35932 at various clock swings in 45-nm technology.

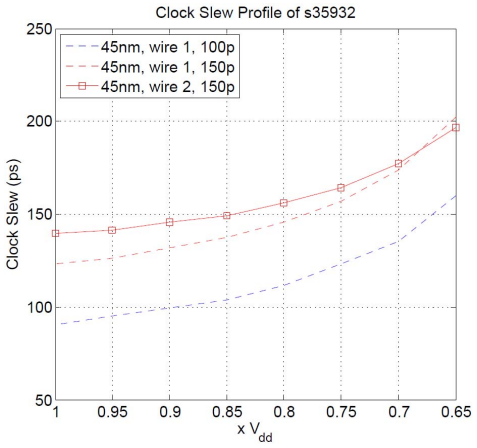


Fig. 6. Clock slew profile of s35932 at various clock swings in 45-nm technology.

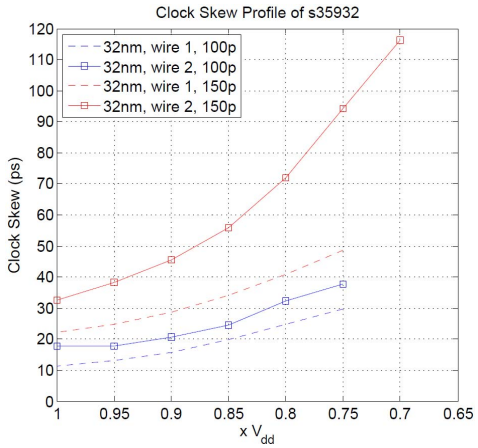


Fig. 5. Clock skew profile of s35932 at various clock swings in 32-nm technology.

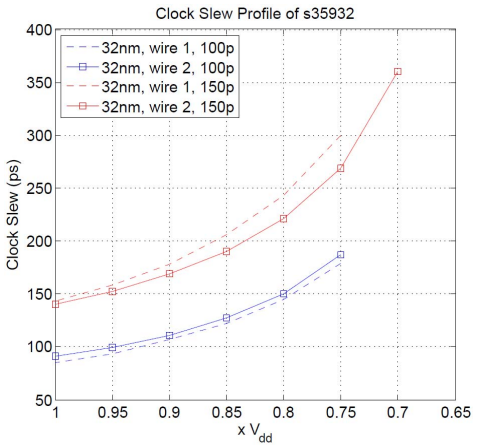


Fig. 7. Clock slew profile of s35932 at various clock swings in 32-nm technology.

skew constraint, as shown in Fig. 5. This violation can be fixed with a post-CTS optimization using in-place buffer sizing [10]. A more challenging issue in LS clocking is the significant increase in the clock slew.

2) Clock Slew: Lower voltage degrades the drive ability of clock buffers, which significantly increases clock slew, particularly in nanoscale technologies where interconnect resistance is dominant. This deleterious effect is shown in Figs. 6 and 7 for, respectively, 45- and 32-nm technologies. According to Figs. 6 and 7, in LS operation, the clock slew increases by ∼50% in a 45-nm technology, and ∼100% in a 32-nm technology. Note that the same slew constraint as in FS clocking can be satisfied at LS operation through buffering the existing topology. This approach, however, causes significant power dissipation due to the necessity of high number of clock buffers to satisfy the slew constraint at each clock sink. Furthermore, the increase in the number of clock buffers increases the insertion delay of clock tree, potentially increasing the clock skew more than what is shown in Figs. 4 and 5. Due to these reasons and the failure to efficiently fix these violations through buffering, a clock swing-aware synthesis of the clock tree is required at LS operation.

3) Power Consumption: The effect of clock voltage on power consumption is investigated when the clock tree is resynthesized (using the proposed CTS algorithm) to satisfy both skew and slew constraints. The power consumption of the clock tree (clock buffers and interconnects) at each voltage level is shown in Figs. 8 and 9 for, respectively, 45- and 32-nm technologies. Note that the lowest voltage in the graphs represents the minimum achievable voltage without introducing any slew violations.

According to Fig. 8, for 45-nm technology, at a 100-ps slew constraint using wire 1, the power savings reach a maximum of ∼33% when the clock swing is at 75% of $V_{\rm dd}$. Note that no feasible clock tree can be synthesized for a 100-ps slew constraint case using wire 2 (due to high resistance). Thus, this case is omitted in Fig. 8. At a 150-ps slew constraint using wire 1, the power savings reach ∼46% when the clock swing is at 65% of $V_{\rm dd}$. Alternatively, at a 150-ps slew constraint using wire 2, power savings reach ∼44% when clock swing is at 75% of $V_{\rm dd}$.

According to Fig. 9, for 32-nm technology, at a 100-ps slew constraint, the power savings are ∼6% when the clock swing is at 90% and 95% of $V_{\rm dd}$ for, respectively, wire 1 and wire 2. For the 150-ps slew constraint, the power savings reach ∼18%

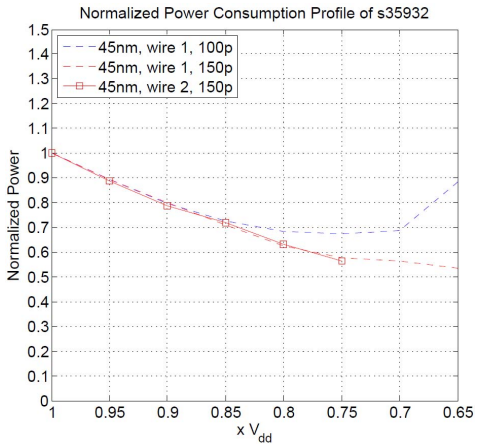


Fig. 8. Power consumption profile of s35932 (clock buffers and interconnects) at various clock swings in 45-nm technology.

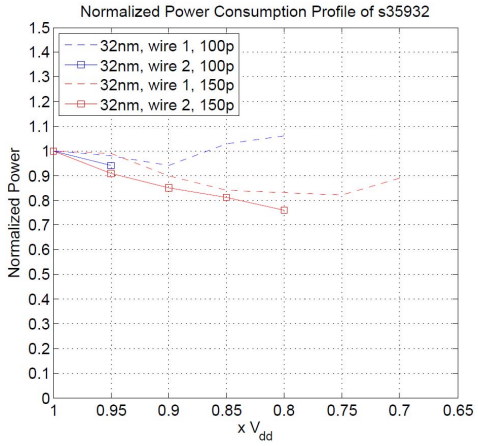


Fig. 9. Power consumption profile of s35932 (clock buffers and interconnects) at various clock swings in 32-nm technology.

(when the clock swing is at 75% of V_{dd}) and 24% (when the clock swing is at 80% of V_{dd}) for, respectively, wire 1 and wire 2.

As shown in Figs. 8 and 9, the minimum achievable clock swing varies depending upon the interconnect technology (wire 1 versus wire 2) and the slew constraint. In the same cases, power consumption starts to increase, if the clock voltage is reduced more than a critical level (such as 0.7 V in 45-nm technology and a 100-ps slew constraint with wire 2). In these cases, the power consumed by the large number of clock buffers (required to satisfy the slew constraint) exceeds the power savings achieved by voltage scaling.

According to this experimental setup, depending upon the specific case (as described above), power savings vary between ∼6% and ∼46%. Thus, a slew-aware CTS methodology is required to efficiently use the resources at different interconnect and transistor technologies, and clock frequencies. The proposed methodology for LS clocking at high performance is described in Section IV.

IV. Proposed Methodology

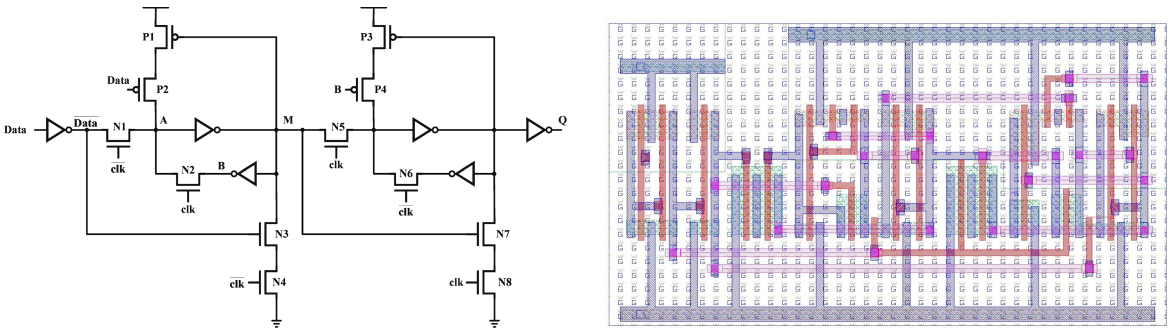
As summarized in Fig. 1, the proposed methodology consists of an LSDFF cell (described in Section IV-A) and an LS-aware CTS algorithm (described in Section IV-B). The proposed LSDFF cell enables LS operation throughout the entire clock tree without introducing power and delay overhead, even though the incoming data signal is at FS. Similarly, the proposed CTS algorithm utilizes this DFF cell and generates an LS clock tree that maximizes the power savings while satisfying the original timing constraints.

A. Low-Swing D-Flip-Flop Cell

1) Proposed Topology: The proposed DFF topology, shown in Fig. 10, is based on the most commonly used static DFF shown in Fig. 2. Rather than using transmission gates, however, pass gates with nMOS transistors (N1, N2, N5, and N6) are utilized as the switches in both master and slave latches. Thus, when the LS clock signal is at logic high, N1 and N6 can completely turn OFF. Pass gates, however, cannot transfer a full voltage to the output. This issue is critical, since the incoming data signal operates at FS. Thus, node A cannot reach a full V_{DD}, thereby increasing the short circuit and leakage current in the following stages in addition to increasing the clock-to-Q delay. Furthermore, pass transistors are less robust to process variations. To alleviate these issues, a pull-up network consisting of two pMOS transistors is added to both master and slave latches (P1–P4). When the master node M transitions to logic low, P1 turns ON. If the data signal is also at logic low, then node A is pulled to full V_{DD} through P1 and P2. Note that P2 (in the master latch) and P4 (in the slave latch) are added to prevent contention current (and, therefore, reduce power consumption) when the data signal is at logic high and the clock signal is at logic low. In this situation, N1 is ON and node A is discharged through N1 and the inverter. If P2 did not exist, a race condition would occur at node A, since N1 should be stronger than P1, which pulls node Y to full V_{DD}. Finally, a pull-down logic (N3, N4, N7, and N8) is added to both master and slave latches to enhance the clock-to-Q delay. In particular, when data and clock signals are at logic low, the pull-down logic is active and pulls master node M to ground, triggering P1. Thus, node A quickly reaches full V_{DD}. Note that the master node does not need to wait for node A to rise through a weak pass transistor and activate the inverter. Instead, the pull-down logic completes this transition relatively faster. Also note that the clock subcircuit (not shown in Fig. 2) is identical to the circuit shown in Fig. 2.

2) Results: The proposed LSDFF topology is designed in both 45- and 32-nm technologies. The DFF cell successfully latches both logic-low and logic-high FS data signals after the rising edge of the LS clock signal. The output signal reaches nominal (FS) V_{DD}, and the DFF cell does not exhibit glitches in any of the internal nodes.

To compare the proposed LSDFF cell with the conventional FSDFF cell, the power and clock-to-Q delay are analyzed as a function of clock swing for both 45- and 32-nm technologies. The overall power consumption is compared in Fig. 11. According to Fig. 11, for both technologies, FSDFF consumes less power than LSDFF at relatively large clock swings. As the clock swing is reduced, however, LSDFF significantly



(a) (b)

Fig. 10. Proposed DFF topology that can reliably work with an LS clock signal whereas the data and output signals are at FS. (a) Transistor level representation. (b) Physical layout in the 45-nm technology node.

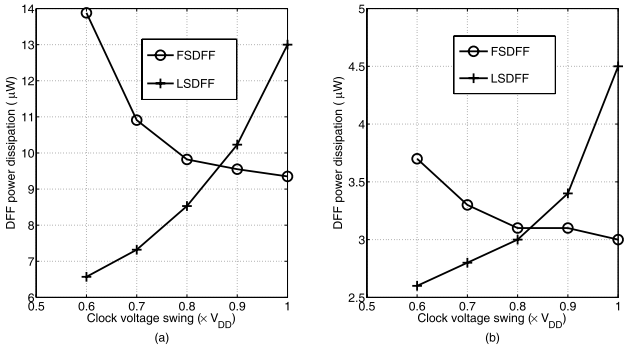


Fig. 11. Power consumption comparison of the proposed LSDFF cell with the conventional FSDFF cell. (a) 45-nm technology. (b) 32-nm technology.

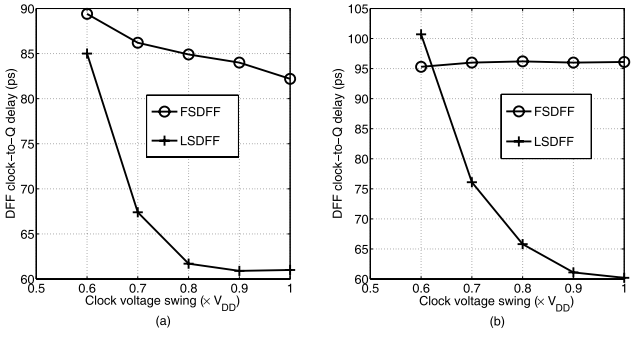


Fig. 12. Clock-to-Q delay comparison of the proposed LSDFF cell with the conventional FSDFF cell. (a) 45-nm technology. (b) 32-nm technology.

outperforms FSDFF. The crossover voltage is ∼0.85 V for 45-nm technology and 0.81 V for 32-nm technology. At a clock swing of 0.6 V, LSDFF consumes 53% and 30% less power than FSDFF in, respectively, 45- and 32-nm technologies.

The clock-to-Q delay of the LSDFF and FSDFF is compared as a function of clock swing in Fig. 12. According to Fig. 12, for both technologies, LSDFF outperforms FSDFF in all clock swings except 0.6 V in 32-nm technology. The clock-to-Q delay of the LSDFF at this point is only 5 ps more than FSDFF. It is important to note that the LSDFF running at 0.65 V or higher exhibits less clock-to-Q delay than FSDFF running at FS. This characteristic is highly important to maintain data path timing the same (or with additional slack) when the conventional flip-flops are replaced with LSDFFs. The clock-to-Q delay in LSDFF is adjusted by sizing the last two inverter stages. Note that the clock pin capacitance remains the same as FSDFF, since the size of the first inverter within the clock subcircuit (see Fig. 2) is kept constant. Also note that, for a fair analysis, the size of the transistors within the flip-flops remains the same at each clock voltage. In the experimental results presented in Section V, however, the LS flip-flop is resized to optimize the clock-to-Q delay and power consumption at the selected clock voltage. For example, at the clock voltage swing of $0.9 \times V_{\mathrm{DD}}$, the significantly lower delay achieved by LSDFF is traded to lower the power consumed by LSDFF.

The timing constraint characterization of the LSDFF and FSDFF demonstrates that the proposed topology has comparable setup and hold times, as listed in Table I [21]. In particular, for hold time, LSDFF slightly outperforms FSDFF. This characteristic is important to ensure that no min-delay constraint violations are introduced in short data paths. Alternatively, for setup time, FSDFF slightly outperforms LSDFF. The difference, however, is sufficiently small as compared with the clock period in multigigahertz designs.

TABLE I

CONSTRAINT CHARACTERIZATION OF THE FSDFF AND THE PROPOSED LSDFF IN BOTH 45- AND 32-nm TECHNOLOGY NODES

Topology		45 nm		32 nm	
		FSDFF	LSDFF	FSDFF	LSDFF
Setup time (ps)	Latch 1	11	23	4	18
	Latch 0	12	28	6	26
Hold time (ps)	Latch 1	-1	-15	-2	-15
	Latch 0	-7	-19	2	-7

The proposed LSDFF has also been simulated at the slow and fast corners to evaluate the robustness of the topology to PVT variations. As listed in Table II, at a clock swing of 0.7 V, the proposed topology achieves reliable operation and outperforms the conventional topology at each corner (nominal, slow, and fast).

Finally, the cell area consumed by the proposed and existing topologies is listed in Table III. According to Table III, LSDFF consumes ∼50% and 55% additional area for, respectively, 45- and 32-nm technologies. The effect of this increase in the overall die area, however, is below 10%, as quantified in Section V.

TABLE II

CORNER SIMULATION RESULTS OF THE FSDFF AND THE PROPOSED LSDFF IN BOTH 45- AND 32-nm TECHNOLOGY NODES WHEN THE NOMINAL CLOCK SWING IS 0.7 V

Topology	45 nm		32 nm	
	FSDFF	LSDFF	FSDFF	LSDFF
Nominal corner (TT model, 1.0V and 1.05V supply and 25°C)				
Clock-to-Q delay (ps)	86.16	68.39	95.93	77.50
Dynamic power (μW)	10.75	7.24	3.63	2.77
Worst corner (SS model, 0.9V and 0.95V supply and 125°C)				
Clock-to-Q delay (ps)	183.95	180.44	181.86	175.19
Dynamic power (μW)	6.83	5.86	2.18	2.05
Best corner (FF model, 1.1V and 1.15V supply and -40°C)				
Clock-to-Q delay (ps)	57.02	26.12	55.06	15.69
Dynamic power (μW)	21.62	9.28	9.69	9.78

TABLE III

CELL AREA COMPARISON OF THE FSDFF AND THE PROPOSED LSDFF IN BOTH 45- AND 32-nm TECHNOLOGY NODES

Topology		Area (μm×μm)
45 nm	FSDFF	3.285×2.67=8.77
	LSDFF	4.93×2.67=13.16
32 nm	FSDFF	2.674×1.672=4.47
	LSDFF	4.166×1.672=6.96

B. Low-Swing Clock Tree Synthesis

The timing characterization of clock buffers and interconnects (required only if the related timing libraries are not available) is discussed in Section IV-B1. The proposed CTS algorithm is described in Section IV-B2.

1) Clock Timing Modeling: In industrial design flows, the timing characteristics (delay and slew) of clock buffers and interconnects are provided with a cell library at a range of values in a target technology node. These models, however, may not be available at the LS/voltage operation. Several methods [22]–[27] are discussed in this section for clock buffer and interconnect modeling, which are required only if these models are not available within the cell library.

Based on the target accuracy, the timing (delay and slew) characterization of clock buffers can be achieved using different methodologies [22]–[24]. The first option is to use higher order models [22], [23] in order to estimate timing characteristics with a higher accuracy at the expense of greater computational complexity. The second option [24] is to use simpler first-order models that have sufficient accuracy for a CTS. In [24], it is shown that the first-order model has good fidelity as compared with the SPICE simulations, achieving an error of <4.6% in estimating insertion delay. Thus, the second option [24] is preferred in this paper. However, it is important to emphasize that the algorithm presented in Section IV-B2 is independent of the selected timing characterization methodology.

In [24], the delay $D(B)$ and the output slew $\text{Slew}_{\text{out}}(B)$ of a clock buffer B is modeled as a linear equation of input slew $\text{Slew}_{\text{in}}(B)$ and output capacitance $\text{Cap}_{\text{out}}(B)$ of the buffer. In particular, the delay of a clock buffer $D(B)$ can be written as

$$D(B) = K_{\text{slew}}^{\text{delay}} \times \text{Slew}_{\text{in}}(B) + K_{\text{cap}}^{\text{delay}} \times \text{Cap}_{\text{out}}(B) + K_{\text{delay}} \quad (1)$$

where $K_{\text{slew}}^{\text{delay}}$ and $K_{\text{cap}}^{\text{delay}}$ are, respectively, the coefficients for the input slew $\text{Slew}_{\text{in}}(B)$ and output capacitance $\text{Cap}_{\text{out}}(B)$ for delay computation. K_{delay} is the intrinsic delay of the buffer. As for the output slew $\text{Slew}_{\text{out}}(B)$, it is observed that the input slew does not have a significant effect; therefore, the output slew of a buffer B can be estimated as

$$\text{Slew}_{\text{out}}(B) = K_{\text{cap}}^{\text{slew}} \times \text{Cap}_{\text{out}}(B) + K_{\text{slew}} \quad (2)$$

where $K_{\text{cap}}^{\text{slew}}$ is the coefficient of the output capacitance for slew computation and K_{slew} is the intrinsic output slew. Given the clock slew constraint (for instance, 150 ps), these coefficients are obtained by sweeping the input slew and the output capacitance around the slew constraint.

The wire delay can be estimated with the well-known Elmore delay [25] with sufficient accuracy. The slew degradation on a wire segment T can be estimated using the Bakoglu [26] metric simplified for an ideal wire input with zero input slew

$$\text{Slew}_{\text{ideal}}(T_i, T_f) = \ln 9 \times D(T_i, T_f) \quad (3)$$

where $D(T_i, T_f)$ is the Elmore delay of the wire segment T from its initial point T_i to final point T_f. This result can be extended for wires with nonzero input slews, by using the PERI model estimation [27]. In this estimation, the output slew of the wire segment T is estimated at its final position T_f as

$$\text{Slew}_{\text{wire}}(T_f) = \sqrt{\text{Slew}_{\text{wire}}(T_i)^2 + \text{Slew}_{\text{ideal}}(T_i, T_f)^2} \quad (4)$$

where $\text{Slew}_{\text{wire}}(T_i)$ is the input slew of the wire segment T estimated at its initial position T_i. In a buffered RC network, the output of a buffer is the input of a wire segment and vice versa. Thus, (1)–(4) are used to estimate the slew and the delay propagation along the clock tree.

2) Proposed Algorithm: In this section, the proposed slew-aware LS CTS algorithm is introduced. Algorithm 1 adopts the well-known zero-skew-tree DME method [28] to merge two nodes into one at each step. The merging cost is inspired by [29], which considers both the capacitance and the delay as the cost metric. In this paper, this cost is modified to consider the slew and the delay, in order to accurately capture the impact of voltage scaling (LS operation) and higher wire resistance for sub-45-nm technologies.

In Algorithm 1, lines 5–15 identify whether a merging pair is feasible. If a feasible pair is identified, lines 16–30 describe the merge process, including a novel embedded skew minimization scheme. If a feasible pair is not identified, a buffering process is proposed in lines 31–36 to help satisfy the slew constraint (that causes the infeasible merge process). The feasibility is monitored by checking if the slew constraint can be satisfied with the presence of a buffer at the temporary merging point $T_{i,j}$ of child nodes i and j by calculating the maximum slew $T_{i,j}$ produces, using (2)–(4) (lines 7–10). If no feasible point is found, the buffers are inserted at the unmerged nodes, and their capacitance, delay, and slew constraint parameters are updated (lines 32–35). If a feasible pair is available (i.e., satisfying the slew constraint) for merging (line 16), the one with the minimum

Algorithm 1 Slew-Aware LS CTS

Input: Buffer library, Timing models at the voltage level, Skew and slew constraints ($skew_{const}$,$slew_{const}$)
Output: Clock buffer and interconnect locations
1: Initialize nodes, $Num\_of\_unmerged = Num\ of\ sinks$
2: $D^{max}(i)$=$D^{min}(i)$=0, $slew_{const}(i)$=$slew_{const}$ for each node i
3: **while** $Num\_of\_unmerged > 1$ **do**
4: $Cost_{curr}$=∞
5: **for** i *in Nodes* **do**
6: **for** j *in Nodes* **do**
7: $Slew_{out}(T_{i,j}) = K_{cap}^{slew} \times Cap_{out}(T_{i,j}) + K_{slew}$
8: $X = max[Slew_{ideal}(T_{i,j}, i),\ Slew_{ideal}(T_{i,j}, j)]$
9: $Slew_{wire}(T_{i,j}) = \sqrt{X^2 + Slew_{out}(T_{i,j})^2}$
10: **if** $Cost(i,j) < Cost_{curr}$ && $Slew_{wire}(T_{i,j}) < Min[slew_{cons}(i), slew_{cons}(j)]$ **then**
11: $Cost_{curr}$=Cost(i,j)
12: $Temp_{slew}$=$Slew_{wire}(T_{i,j})$
13: **end if**
14: **end for**
15: **end for**
16: **if** $Cost_{curr}$!= ∞ **then**
17: Num_of_unmerged−−
18: Initialize new node k=$T_{i,j}$
19: $D^{max}(k)$=max[$D^{max}(i)$+D(i,k),$D^{max}(j)$+D(j,k)]
20: $D^{min}(k)$=min[$D^{min}(i)$+D(i,k),$D^{min}(j)$+D(j,k)]
21: **while** $D^{max}(k)$-$D^{min}(k) > skew_{const}$ **do**
22: **if** $D^{max}(k)$-$D^{min}(k) > K_{delay}$ **then**
23: Insert a buffer at the lower delay node
24: **else**
25: Apply wire snaking at the lower delay node
26: **end if**
27: Update $D^{max}(k)$ and $D^{min}(k)$ using Eq. (1)
28: Update $Slew_{const}(i)$ and $Slew_{const}(j)$
29: **end while**
30: $Slew_{const}(k)$= $\sqrt{min[Slew_{const}(i), Slew_{const}(j)]^2 - Temp_{slew}{}^2}$
31: **else**
32: **for** i *in Unmerged Nodes* **do**
33: Insert buffer, update $D^{max}(i)$, $D^{min}(i)$ using Eq. (1)
34: $slew_{const}(i)$=$slew_{const}$
35: **end for**
36: **end if**
37: **end while**

cost (recorded at line 11) is initialized as node k to merge child nodes i and j at this new node (lines 17–20). After the maximum and the minimum delays from k to the child nodes are updated, it is checked if the difference between the maximum and the minimum is larger than the skew constraint $\text{skew}_{\text{const}}$ (line 21). Possible skew violation is fixed through moderate wire snaking for small violations and through buffer insertion for larger violations: If the skew violation is larger than the intrinsic buffer delay K_{delay}, buffer insertion is preferred instead of large amounts of wire snaking that would have been necessary (lines 22 and 23). Otherwise, wire snaking is performed (lines 24 and 25). This procedure continues until the number of unmerged nodes is one, which is the source of the clock tree.

An important characteristic of this embedded skew minimization scheme is the ability to build a skew balanced clock tree at each level. Thus, this approach has the potential to minimize the buffering/wiring cost at the upper levels of the clock tree.

TABLE IV
FLOORPLAN AREA, NUMBER OF DFFS, AND NUMBER OF GATES FOR THE BENCHMARK CIRCUITS USED TO EVALUATE THE PROPOSED METHODOLOGY

Benchmark circuits	Floorplan Size (μm×μm)	Number of DFFs	Number of Gates
s1423	83×82	74	657
s9234	101×101	145	5597
s5378	112×113	176	2779
s15850	173×169	513	9772
s13207	169×166	593	7951
s38584	280×272	1274	19353
s38417	285×281	1575	22179
s35932	289×281	1728	16065

In Algorithm 1, each pair of nodes is visited at each iteration of the *while* loop [complexity of $O(n^2)$ for n clock sinks]. Initially, the number of unmerged nodes is equal to the number of clock sinks (n), and it is decreased by one at every iteration of the loop. The *while* loop is repeated until the number of nodes is one, and therefore, this loop is executed n times. Thus, the overall complexity of the algorithm is $O(n^3)$. However, the cost and the feasibility metrics are needed only for the $O(n^2)$ combinations of pairs that are computed in the first iteration. These metrics are stored in 2-D arrays. Thus, after the first iteration, these metrics are looked up without a recomputation. Consequently, the algorithm performs $O(n^3)$ lookups and $O(n^2)$ computations, making the lower order n^2 term dominant. This postulate is also verified by the runtime data that exhibit a quadratic dependence to the number of clock sinks.

V. EXPERIMENTAL RESULTS

A. Simulation Setup

The proposed algorithm is implemented in Perl, and the output circuits are tested using HSPICE of Synopsys with FreePDK 45-nm and SAED 32-nm technology nodes [19], [30]. The proposed LSDFF cell is also designed using these technologies.

Eight largest circuits from ISCAS'89 benchmarks (floorplan sizes and number of DFF information are listed in Table IV) are selected for experimental analyses. It is important to note that the following holds.

1) ISPD'10 benchmarks cannot be used for this paper, as these benchmarks do not contain any DFF information. Instead, ISPD'10 benchmarks have capacitance information to model the clock pin of DFFs.
2) ISCAS'89 benchmarks have comparable circuit size with ISPD'10 benchmarks (1728 versus 2249 for the largest number of DFFs).

The logic synthesis and the physical placement of the DFF sinks are performed using SoC Encounter of Cadence. The largest buffer from the SAED library (NBUFFX32) [19] of Synopsys is used as the clock buffer. The experiments are performed, and the clock trees are synthesized at the slow corner (for delay, skew, and slew) of each technology in order to verify functionality in reduced noise margins. Thus, the skew and slew constraints are satisfied at the slow corner, where the power supply voltage of the LS clock network is

TABLE V

COMPARISON OF CLOCK TREE POWER (CP IN mW), DFF POWER (DFFP IN mW), CLOCK SKEW (Sk. IN ps), CLOCK SLEW (Sl. IN ps), AND THE CLOCK-TO-Q DELAY (C2Q IN ps) FOR THE BASELINE FS, THE LS IMPLEMENTATION WITH [10], AND THE PROPOSED LS METHODOLOGY FOR 45-nm TECHNOLOGY ALONG WITH WIRE 1 RUNNING AT 1 GHz AND WORST CASE CORNER. LS CLOCK VOLTAGE IS AT 0.65 × V_{dd}

Circuits	Full Swing (FS)					Low Swing (LS) with [10]					Proposed Low Swing (LS)				
	CP	DFFP	Sk.	Sl.	C2Q	CP	DFFP	Sk.	Sl.	C2Q	CP	DFFP	Sk.	Sl.	C2Q
s1423	0.284	0.269	0.2	48.2	166.7	0.116	0.323	0.1	94.8	199.9	0.121	0.210	0.3	101.2	168.8
s9234	0.365	0.528	0.3	63.8	169.3	0.155	0.636	0.3	120.8	208.1	0.165	0.412	0.3	133.7	173.0
s5378	0.421	0.637	0.1	80.4	172.9	0.177	0.786	0.1	144.6	212.8	0.333	0.499	14.3	123.6	172.0
s15850	1.255	1.885	8.3	127.7	180.4	0.527	2.400	16.1	222.7	233.1	0.786	1.456	12.3	138.3	172.8
s13207	1.343	2.176	28.4	138.0	182.4	0.565	2.796	37.9	242.6	237.9	0.834	1.686	20.1	140.8	173.8
s38584	3.189	4.656	37.9	118.6	178.5	1.342	5.872	48.2	210.5	229.1	2.014	3.649	22.5	141.3	172.9
s38417	3.685	5.749	46.9	125.3	179.8	1.499	7.320	38.7	237.2	231.1	2.319	4.519	39.9	140.3	173.0
s35932	3.908	6.318	9.2	121.3	179.9	1.659	7.990	8.8	210.9	229.3	2.581	4.959	44.9	142.6	173.3
Norm.	1.00	1.00	<50ps	<150ps	176.2	0.42	1.27	<50ps	**VIO**	222.7	0.63	0.78	<50ps	<150ps	172.5

TABLE VI

COMPARISON OF CLOCK TREE POWER (CP IN mW), DFF POWER (DFFP IN mW), CLOCK SKEW (Sk. IN ps), CLOCK SLEW (Sl. IN ps), AND THE CLOCK-TO-Q DELAY (C2Q IN ps) FOR THE BASELINE FS, THE LS IMPLEMENTATION WITH [10], AND THE PROPOSED LS METHODOLOGY FOR 45-nm TECHNOLOGY WITH WIRE 2 RUNNING AT 1 GHz AND WORST CASE CORNER. LS CLOCK VOLTAGE IS AT 0.75 × V_{dd}

Circuits	Full Swing (FS)					Low Swing (LS) with [10]					Proposed Low Swing (LS)				
	CP	DFFP	Sk.	Sl.	C2Q	CP	DFFP	Sk.	Sl.	C2Q	CP	DFFP	Sk.	Sl.	C2Q
s1423	0.358	0.270	0.9	88.2	173.9	0.197	0.288	1.2	110.3	191.6	0.191	0.192	1.1	108.5	182.5
s9234	0.873	0.531	17.6	126.3	179.7	0.469	0.571	23.3	158.3	200.8	0.561	0.379	4.0	79.6	176.0
s5378	0.966	0.646	6.4	124.3	180.3	0.522	0.696	7.1	153.4	200.1	0.610	0.468	25.6	132.6	186.0
s15850	2.509	1.873	27.7	117.5	178.7	1.355	2.021	36.2	147.6	198.7	1.628	1.385	49.9	141.6	187.6
s13207	3.204	2.162	31.5	127.6	179.9	1.729	2.350	38.5	163.4	201.6	1.799	1.594	17.1	109.7	182.0
s38584	8.825	4.712	47.6	137.9	181.3	4.623	5.208	43.5	168.1	202.9	4.558	3.468	33.5	124.2	184.5
s38417	9.340	5.815	49.8	133.6	180.5	4.846	6.458	47.0	261.5	216.7	5.200	4.289	38.2	134.2	187.0
s35932	9.556	6.380	37.4	136.9	181.6	5.143	7.015	36.8	173.8	203.4	5.468	4.703	39.2	141.1	188.4
Norm.	1.00	1.00	<50ps	<150ps	179.5	0.53	1.10	<50ps	**VIO**	202.0	0.56	0.74	<50ps	<150ps	184.3

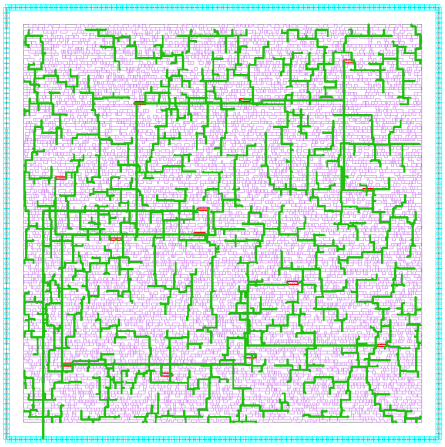


Fig. 13. Physical layout of the synthesized LS clock tree for benchmark circuit s38584 in 32-nm technology.

reduced by 10%. These PVT corners correspond, respectively, to slow–slow device models, 0.9 V (0.9 × 1) and 125 °C for the 45-nm technology node; and slow–slow device models, 0.95 V (0.9 × 1.05), and 125 °C for the 32-nm technology node. Thus, if the power grid of the clock network is designed to ensure that the largest voltage drop does not exceed 10% of the LS clock voltage, the synthesized clock trees operate as intended. As an example, a synthesized LS clock tree is shown in Fig. 13 for benchmark circuit s38584 in 32-nm technology.

In order to highlight the efficacy of the proposed swing- and slew-aware CTS algorithm with the presence of the proposed LSDFF topology, the power consumption of the LS clock tree (CP), the overall power consumption of DFF cells (DFFP), clock skew (Sk.), maximum clock slew (Sl.), and the maximum clock-to-Q delay (C2Q) of the DFFs are compared with an FS clock tree that has traditional FSDFFs. A custom implementation of the LS clock tree using the method in [10] (which performs voltage scaling on an existing FS clock tree with a traditional FSDFF) is also included in the comparison. For each technology node (45 and 32 nm), combinations of wire models (wire 1 and wire 2 described in Section III-B) and frequency constraints (1 and 1.5 GHz with scaled slew constraints of 150 and 100 ps, respectively) are considered. An LS voltage level is selected as the level that has minimum power consumption for each case. The clock skew constraint is set to 50 ps at all cases. The results are listed in Tables V–XI. Note that the last row in these tables is based on average values considering all of the benchmark circuits.

B. 45-nm Technology

The experimental results in 45-nm technology are listed in the following three tables.

1) *Table V:* 45 nm, wire 1, 150-ps slew constraint at 1 GHz.
2) *Table VI:* 45 nm, wire 2, 150-ps slew constraint at 1 GHz.
3) *Table VII:* 45 nm, wire 1, 100-ps slew constraint at 1.5 GHz.

As mentioned in Section III-B, the case with wire 2 and a 100-ps slew constraint is omitted for this technology, since no feasible clock trees can be synthesized, even at full V_{dd}.

TABLE VII

COMPARISON OF CLOCK TREE POWER (CP IN mW), DFF POWER (DFFP IN mW), CLOCK SKEW (Sk. IN ps), CLOCK SLEW (Sl. IN ps), AND THE CLOCK-TO-Q DELAY (C2Q IN ps) FOR THE BASELINE FS, THE LS IMPLEMENTATION WITH [10], AND THE PROPOSED LS METHODOLOGY FOR 45-nm TECHNOLOGY WITH WIRE 1 RUNNING AT 1.5 GHz AND WORST CASE CORNER. LS CLOCK VOLTAGE IS AT $0.75 \times V_{dd}$

Circuits	Full Swing (FS)					Low Swing (LS) with [10]					Proposed Low Swing (LS)				
	CP	DFFP	Sk.	Sl.	C2Q	CP	DFFP	Sk.	Sl.	C2Q	CP	DFFP	Sk.	Sl.	C2Q
s1423	0.419	0.408	0.2	44.4	166.3	0.230	0.427	0.2	67.7	180.7	0.226	0.303	0.1	65.9	172.1
s9234	0.549	0.803	0.3	61.3	169.1	0.308	0.846	0.3	88.6	186.3	0.586	0.589	12.9	76.0	175
s5378	0.628	0.975	0.1	79.0	172.3	0.351	1.038	0.2	112.4	191.5	0.628	0.747	0.6	74.0	174.3
s15850	2.165	2.872	6.2	92.5	174.2	1.199	3.051	5.2	129.9	194.1	1.465	2.198	15.7	84.6	177.1
s13207	2.287	3.316	14.4	90.1	173.3	1.271	3.531	17.7	128.5	194.4	1.535	2.522	9.6	88.2	178.0
s38584	5.062	7.162	25.4	93.6	174.3	2.811	7.643	29.4	131.1	194.4	3.903	5.598	10.9	86.8	177.8
s38417	6.659	8.900	25.6	95.4	174.9	3.677	9.526	34.3	132.8	195.4	4.424	6.935	28.1	87.8	178.2
s35932	7.561	9.751	23.3	90.5	173.5	4.173	10.390	30.9	128.2	194.4	5.084	7.602	18.8	86.5	178.0
Norm.	1.00	1.00	<50ps	<100ps	172.2	0.55	1.07	<50ps	**VIO**	191.4	0.70	0.77	<50ps	<100ps	176.3

TABLE VIII

COMPARISON OF CLOCK TREE POWER (CP IN mW), DFF POWER (DFFP IN mW), CLOCK SKEW (Sk. IN ps), CLOCK SLEW (Sl. IN ps), AND THE CLOCK-TO-Q DELAY (C2Q IN ps) FOR THE BASELINE FS, THE LS IMPLEMENTATION WITH [10], AND THE PROPOSED LS METHODOLOGY FOR 32-nm TECHNOLOGY WITH WIRE 1 RUNNING AT 1 GHz AND WORST CASE CORNER. LS CLOCK VOLTAGE IS AT $0.75 \times V_{dd}$

Circuits	Full Swing (FS)					Low Swing (LS) with [10]					Proposed Low Swing (LS)				
	CP	DFFP	Sk.	Sl.	C2Q	CP	DFFP	Sk.	Sl.	C2Q	CP	DFFP	Sk.	Sl.	C2Q
s1423	0.146	0.105	0.3	82.3	109.9	0.080	0.146	0.2	161.1	118.0	0.131	0.102	30.8	137.7	118.7
s9234	0.217	0.217	0.5	115.0	112.2	0.122	0.319	0.4	234.5	120.1	0.199	0.199	28.8	127.3	120.7
s5378	0.265	0.272	0.3	135.3	113.5	0.147	0.414	0.4	283.0	120.6	0.251	0.241	3.3	121.6	120.7
s15850	0.815	0.778	2.6	131.3	113.5	0.451	1.180	4.9	279.7	119.7	0.691	0.703	27.2	141.6	118.8
s13207	0.944	0.881	20.4	130.2	112.6	0.526	1.316	39.9	282.2	120.5	0.732	0.813	24.8	144.6	118.5
s38584	2.161	1.921	28.6	146.7	113.0	1.181	2.906	37.6	304.3	120.7	1.734	1.753	25.6	146.0	119.2
s38417	2.465	2.396	12.9	148.7	113.8	1.359	3.616	13.4	306.7	120.7	2.107	2.165	41.5	144.3	118.7
s35932	2.817	2.617	22.1	144.7	113.0	1.547	3.949	47.2	308.7	120.7	2.301	2.376	49.0	148.5	119.4
Norm.	1.00	1.00	<50ps	<150ps	112.7	0.55	1.51	<50ps	**VIO**	120.1	0.83	0.91	<50ps	<150ps	119.3

TABLE IX

COMPARISON OF CLOCK TREE POWER (CP IN mW), DFF POWER (DFFP IN mW), CLOCK SKEW (Sk. IN ps), CLOCK SLEW (Sl. IN ps), AND THE CLOCK-TO-Q DELAY (C2Q IN ps) FOR THE BASELINE FS, THE LS IMPLEMENTATION WITH [10], AND THE PROPOSED LS METHODOLOGY FOR 32-nm TECHNOLOGY WITH WIRE 2 RUNNING AT 1 GHz AND WORST CASE CORNER. LS CLOCK VOLTAGE IS AT $0.80 \times V_{dd}$

Circuits	Full Swing (FS)					Low Swing (LS) with [10]					Proposed Low Swing (LS)				
	CP	DFFP	Sk.	Sl.	C2Q	CP	DFFP	Sk.	Sl.	C2Q	CP	DFFP	Sk.	Sl.	C2Q
s1423	0.332	0.108	29.7	112.9	111.4	0.205	0.136	46.9	179.7	117.7	0.202	0.094	1.4	128.1	113.7
s9234	0.517	0.207	1.3	90.5	111.0	0.323	0.257	3.7	149.4	116.2	0.351	0.184	20.3	139.5	113.0
s5378	0.651	0.249	4.8	87.6	110.1	0.408	0.309	8.1	144.4	115.7	0.406	0.224	5.6	143.2	113.1
s15850	1.602	0.763	47.1	136.2	112.2	0.985	1.007	40.9	217.0	118.0	1.210	0.650	41.3	148.7	112.9
s13207	1.789	0.880	43.2	141.9	112.4	1.120	1.152	46.7	225.3	118.2	1.313	0.750	12.5	141.7	112.9
s38584	4.518	1.896	41.8	145.7	112.4	2.785	2.488	21.1	229.5	118.4	3.345	1.616	45.7	147.3	113.5
s38417	5.118	2.383	44.5	145.9	112.5	3.183	3.104	24.2	209.0	117.7	3.709	1.999	36.4	145.1	113.6
s35932	5.656	2.585	32.1	141.5	112.3	3.521	3.371	46.9	225.1	118.4	4.282	2.198	26.8	148.0	113.2
Norm.	1.00	1.00	<50ps	<150ps	111.8	0.62	1.30	<50ps	**VIO**	117.5	0.73	0.85	<50ps	<150ps	113.2

TABLE X

COMPARISON OF CLOCK TREE POWER (CP IN mW), DFF POWER (DFFP IN mW), CLOCK SKEW (Sk. IN ps), CLOCK SLEW (Sl. IN ps), AND THE CLOCK-TO-Q DELAY (C2Q IN ps) FOR THE BASELINE FS, THE LS IMPLEMENTATION WITH [10], AND THE PROPOSED LS METHODOLOGY FOR 32-nm TECHNOLOGY WITH WIRE 1 RUNNING AT 1.5 GHz AND WORST CASE CORNER. LS CLOCK VOLTAGE IS AT $0.90 \times V_{dd}$

Circuits	Full Swing (FS)					Low Swing (LS) with [10]					Proposed Low Swing (LS)				
	CP	DFFP	Sk.	Sl.	C2Q	CP	DFFP	Sk.	Sl.	C2Q	CP	DFFP	Sk.	Sl.	C2Q
s1423	0.214	0.154	0.2	76.6	110.3	0.172	0.163	0.2	97.8	112.7	0.287	0.140	19.6	82.1	106.1
s9234	0.474	0.299	10.8	79.7	109.7	0.379	0.316	13.1	100.1	112.0	0.433	0.273	5.2	75.7	106.0
s5378	0.543	0.368	1.0	82.1	110.7	0.435	0.391	1.4	103.2	112.9	0.546	0.331	1.6	73.4	106.5
s15850	1.437	1.036	8.6	87.3	110.1	1.155	1.100	9.8	109.1	112.5	1.456	0.967	33.7	86.6	106.2
s13207	1.705	1.223	4.5	85.9	110.1	1.370	1.291	6.2	108.3	112.3	1.590	1.120	12.4	85.5	106.0
s38584	4.173	2.613	28.0	85.5	109.7	3.346	2.762	35.9	107.2	112.2	3.758	2.400	14.1	87.7	106.1
s38417	4.655	3.256	13.5	87.4	109.9	3.743	3.452	17.2	110.0	112.3	4.485	2.968	36.4	86.8	106.1
s35932	5.036	3.577	11.1	84.8	109.9	4.051	3.795	15.7	107.3	112.6	4.741	3.254	28.9	89.2	106.0
Norm.	1.00	1.00	<50ps	<100ps	110.1	0.80	1.06	<50ps	**VIO**	112.4	0.95	0.91	<50ps	<100ps	106.1

In Table V, it is shown that the methodology in [10] achieves significant 58% power savings in the clock tree. These savings, however, are possible with significantly high clock slew that violates the slew constraint (150 ps). The proposed slew-aware methodology simultaneously satisfies both skew and slew constraints while achieving significant

TABLE XI

COMPARISON OF CLOCK TREE POWER (CP IN mW), DFF POWER (DFFP IN mW), CLOCK SKEW (Sk. IN ps), CLOCK SLEW (Sl. IN ps), AND THE CLOCK-TO-Q DELAY (C2Q IN ps) FOR THE BASELINE FS, THE LS IMPLEMENTATION WITH [10], AND THE PROPOSED LS METHODOLOGY FOR 32-nm TECHNOLOGY WITH WIRE 2 RUNNING AT 1.5 GHz AND WORST CASE CORNER. LS CLOCK VOLTAGE IS AT $0.95 \times V_{dd}$

Circuits	Full Swing (FS)					Low Swing (LS) with [10]					Proposed Low Swing (LS)				
	CP	DFFP	Sk.	Sl.	C2Q	CP	DFFP	Sk.	Sl.	C2Q	CP	DFFP	Sk.	Sl.	C2Q
s1423	0.483	0.154	1.6	79.3	110.0	0.433	0.158	1.7	86.6	110.7	0.432	0.148	1.7	86.7	108.8
s9234	0.766	0.305	1.3	90.9	111.1	0.687	0.313	1.0	99.2	111.8	0.748	0.289	9.7	90.3	108.6
s5378	0.967	0.367	4.9	87.6	110.0	0.867	0.376	5.2	95.7	111.1	0.935	0.351	5.6	88.4	108.9
s15850	2.894	1.065	9.3	90.5	109.9	2.593	1.086	11.2	99.2	110.5	2.723	1.019	8.6	94.3	108.8
s13207	3.124	1.228	13.6	87.6	109.7	2.802	1.260	13.9	96.2	110.9	2.860	1.184	13.3	92.6	108.4
s38584	7.931	2.615	18.8	94.0	110.0	7.110	2.687	21.4	103.0	111.0	7.659	2.520	33.9	92.0	108.7
s38417	8.736	3.263	10.1	92.8	110.4	7.849	3.336	11.8	102.4	110.9	8.557	3.113	32.8	87.7	108.7
s35932	10.130	3.562	17.5	90.7	109.8	9.080	3.638	18.1	99.7	110.9	9.599	3.410	29.2	93.1	109.0
Norm.	1.00	1.00	<50ps	<100ps	110.1	0.90	1.02	<50ps	**VIO**	111.0	0.96	0.96	<50ps	<100ps	108.7

37% power savings. Furthermore, the conventional FSDFF used in [10] has poor performance, increasing power consumption by 27% and clock-to-Q delay by 46.5 ps, on average. Alternatively, the proposed LSDFF achieves 22% overall power savings compared with the traditional DFF running at FS while providing smaller clock-to-Q delay (172.5 versus 176.2 ps).

A similar trend is observed in Table VI. The methodology in [10] achieves 47% power savings with slew violations throughout the clock tree while increasing DFF power consumption by 10% and clock-to-Q delay by 22.5 ps. Alternatively, the proposed slew-aware clock tree methodology satisfies both skew and slew constraints, and achieves 44% power savings within the clock tree. The LSDFF achieves 26% overall power savings compared with the traditional DFF operating at FS, and achieves similar clock-to-Q delay (184.3 versus 179.5 ps, on average).

The experimental results listed in Table VII demonstrate that the proposed slew-aware CTS algorithm achieves 30% savings while satisfying the skew and the slew constraints, whereas the methodology in [10] achieves 37% savings with slew violations. Compared with a more relaxed slew constraint (150 ps), the power savings at a tighter slew constraint (100 ps) is slightly reduced (30% versus 37%). This result is intuitive as there is smaller margin for scaling the clock swing at a tighter slew constraint at the DFF sinks. Thus, as the clock frequency increases (tighter slew constraints), the power savings are reduced, since the clock voltage cannot scale as much. The proposed LSDFF achieves 23% power saving along with a comparable clock-to-Q delay against the traditional DFF (176.3 versus 172.2 ps, on average). In [10], however, the overall power consumption of the DFF sinks and clock-to-Q delay increases by, respectively, 7% and 19.2 ps.

C. 32-nm Technology

The experimental results in 32-nm technology are listed in the following four tables.

1) *Table VIII:* 32 nm, wire 1, 150-ps slew constraint at 1 GHz.
2) *Table IX:* 32 nm, wire 2, 150-ps slew constraint at 1 GHz.
3) *Table X:* 32 nm, wire 1, 100-ps slew constraint at 1.5 GHz.
4) *Table XI:* 32 nm, wire 2, 100-ps slew constraint at 1.5 GHz.

As demonstrated in Table VIII, the approach in [10] achieves 45% power savings at the expense of a significantly high clock slew (as much as twice the constraint). The proposed methodology, however, achieves 17% power savings while satisfying both skew and slew constraints. Although the clock-to-Q delay of [10] is comparable to the clock-to-Q delay of a traditional DFF operating at FS, the power consumption increases by 51%. Alternatively, the proposed LSDFF decreases power consumption by 9%. A similar trend is observed in Table IX. The proposed slew-aware CTS methodology achieves 27% savings while satisfying both skew and slew constraints. In [10], the power consumption of DFF increases by 30% whereas the proposed LSDFF decreases the overall DFF power by 15%. According to Table X, the methodology in [10] provides 20% power savings in the clock tree with violations in clock slew, whereas the proposed LS CTS methodology achieves 5% savings within the clock tree without any violations.

Similar to 45-nm technology, the power savings achieved by LS clocking decrease at tighter slew constraints (17% at 150 ps versus 5% at 100 ps). The proposed LSDFF achieves an additional 9% power savings within the DFF, outperforming the conventional DFF of [10] that increases the overall power consumption by 6%.

Note that according to these results, power savings are reduced in a 32-nm technology node. This trend is due to the 32-nm cell library that is optimized for 500 MHz. Thus, in 32-nm technology, the clock voltage does not scale as much as the 45-nm technology node, producing reduced power savings. In general, if the clock buffers have stronger drive ability, it is easier to scale the clock voltage while still satisfying the timing constraints. Thus, in more advanced technologies, the proposed methodology is expected to achieve higher reduction in power.

D. Effect of Process–Voltage–Temperature Corners

As mentioned before, the clock trees are synthesized in the slow (worst) corner. All of the results listed in Tables V–XI are also obtained in the slow corner. The effect of nominal and fast corners on the proposed methodology is investigated in this section by simulating the synthesized clock trees at the nominal (nominal voltage, 27 °C, nominal transistor models)

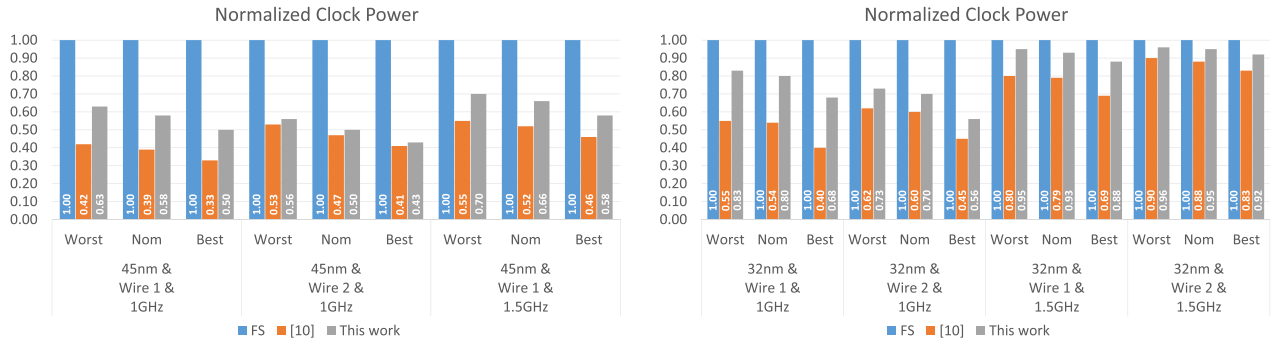


Fig. 14. Corner simulation results of average normalized clock power consumption. (a) 45-nm technology. (b) 32-nm technology.

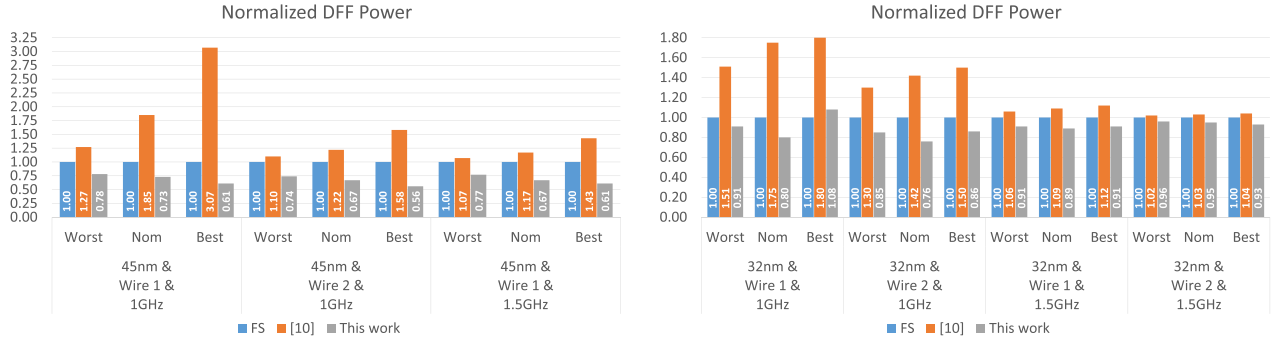


Fig. 15. Corner simulation results of average normalized flip-flop power consumption. (a) 45-nm technology. (b) 32-nm technology.

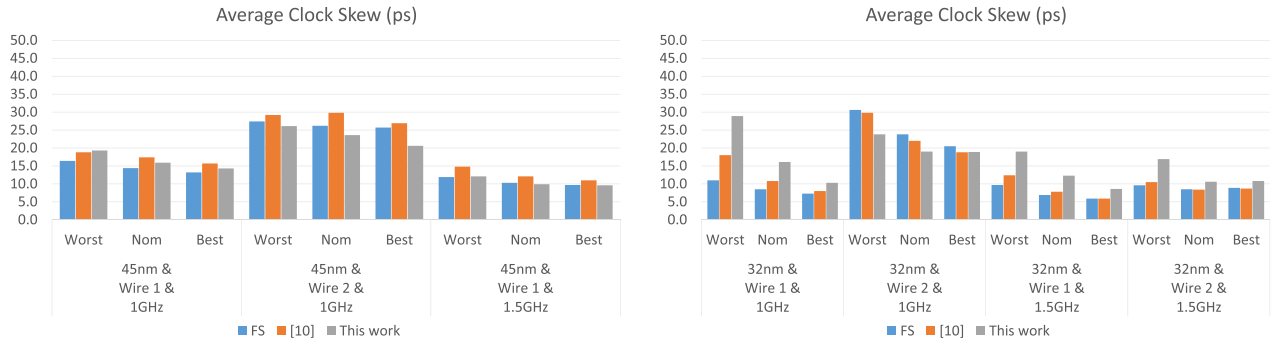


Fig. 16. Corner simulation results of average clock skew. (a) 45-nm technology. (b) 32-nm technology.

and fast ($1.1 \times V_{DD}$, -40 °C, fast transistor models) corners. Results are presented as the average over all of the benchmark circuits.

Normalized power consumed by the clock tree and flip-flops is shown, respectively, in Figs. 14 and 15. An important observation is that at the nominal corner, the power savings achieved by the methodology increase. For example, for 45-nm technology with wire 1 running at 1 GHz, the clock power savings at the slow corner are 37% [see Fig. 14(a)]. At the nominal corner, however, the power savings increase to 42%. The savings in the flip-flop power for the same case are 22% for the slow corner and 27% for the nominal corner [see Fig. 15(a)].

The clock skew results at each corner are shown in Fig. 16. These results represent the average of the highest skew in each benchmark circuit. According to these results, the clock skew is reduced at the nominal and fast corners. Furthermore, the corner-to-corner skew variation is sufficiently small. For example, the maximum skew variation in 45- and 32-nm technologies is, respectively, 5 and 18.6 ps. As compared with the conventional, FS operation, corner-to-corner skew variation increases, at the most, by 3.8 and 14.9 ps in, respectively,

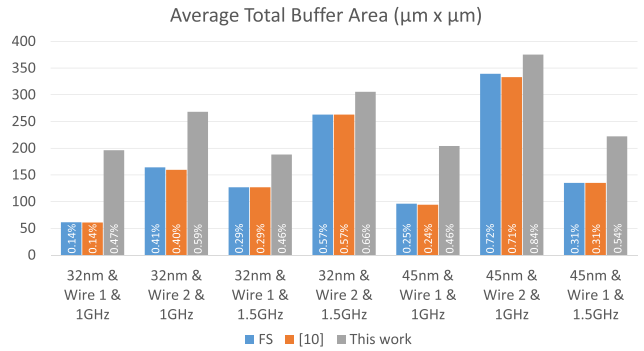


Fig. 17. Average overall clock buffer area comparison in 45- and 32-nm technology nodes.

45- and 32-nm technologies. In general, there is higher skew variation in 32-nm technology, primarily due to larger process variation ranges in the transistor model files.

E. Discussion on the Effect of Interconnect Resistance

An interesting observation is that LS clocking achieves more reduction in power with wire 2 technology (more resistive interconnect) than wire 1. For example, in 45-nm technology, 44% reduction is achieved with wire 2 (see table VI) as compared with 37% reduction with wire 1 (see Table V), despite operating at a higher voltage swing ($0.75 \times V_{dd}$ versus $0.65 \times V_{dd}$). This outcome demonstrates that although high wire resistance (of wire 2) limits the feasibility of clock voltage scaling, the swing- and slew-aware CTS algorithm maximizes the power savings when the interconnect resistance dominates (and, therefore, makes slew awareness more critical). A similar behavior is observed for the 32-nm technology node, as demonstrated in Table VIII (17% power savings with wire 1) and Table IX (27% power savings with wire 2).

When the interconnect resistance dominates, the increase in the device resistance (due to voltage scaling) has a smaller negative impact, thereby favoring LS clocking as an efficient power reduction strategy. This behavior is highly critical, since interconnect resistance further dominates device resistance in sub-32-nm FinFET technologies. Thus, the proposed slew-aware CTS algorithm is highly applicable in scaling the clock voltage of future technology nodes and maximizing power savings while satisfying the required skew and slew constraints.

F. Area Overhead

The area overhead of the proposed methodology is characterized by quantifying the area consumed by the proposed LS flip-flops and clock buffers as a percentage of the overall die area. These results are reported as the average of all of the benchmark circuits. Specifically, in the conventional FS operation, in 45-nm technology, the entire flip-flops consume, on average, 14.84% of the overall die area. Alternatively, the proposed LS flip-flops consume, on average, 22.27% of the overall area, exhibiting 7.4% increase. In 32-nm technology, this increase is only 4.2%.

Area overhead due to the clock buffers is shown in Fig. 17. As expected, the area consumed by the clock buffers is significantly less than the flip-flop area. In particular, in the conventional FS operation, in 45-nm technology, the entire clock buffers consume, on average, 0.43% of the overall die area. Alternatively, in the proposed methodology, the clock buffers consume, on average, 0.61% of the die area, exhibiting only 0.18% increase. In 32-nm technology, this increase is 0.19%.

VI. Conclusion

A design methodology is proposed for voltage-scaled clock networks operating at a reduced swing. The primary objective is to achieve significant reduction in power consumption without degrading circuit performance. The proposed methodology consists of a novel LSDFF cell and a novel swing- and slew-aware CTS algorithm. The proposed DFF cell can reliably operate with an LS clock signal, thereby enabling LS operation throughout the entire clock network. Thus, power savings are maximized. The proposed CTS algorithm ensures that the same clock frequency, skew, and slew as in FS/voltage operation are satisfied. Furthermore, the slew-aware CTS algorithm is sufficiently flexible to target various performance constraints while adapting to the differences in the transistor and interconnect technologies. The entire methodology is integrated into an industrial design flow for automation. Experimental results on largest ISCAS'89 benchmark circuits demonstrate significant reduction in clock power for both 45- and 32-nm technology nodes while satisfying all of the timing constraints.

References

[1] E. Salman and E. G. Friedman, *High Performance Integrated Circuit Design*. New York, NY, USA: McGraw-Hill, 2012.

[2] E. G. Friedman, "Clock distribution networks in synchronous digital integrated circuits," *Proc. IEEE*, vol. 89, no. 5, pp. 665–692, May 2001.

[3] H. Kawaguchi and T. Sakurai, "A reduced clock-swing flip-flop (RCSFF) for 63% power reduction," *IEEE J. Solid-State Circuits*, vol. 33, no. 5, pp. 807–811, May 1998.

[4] K. Chulwoo and S.-M. Kang, "A low-swing clock double-edge triggered flip-flop," *IEEE J. Solid-State Circuits*, vol. 37, no. 5, pp. 648–652, May 2002.

[5] R. Gonzalez, B. M. Gordon, and M. A. Horowitz, "Supply and threshold voltage scaling for low power CMOS," *IEEE J. Solid-State Circuits*, vol. 32, no. 8, pp. 1210–1216, Aug. 1997.

[6] R. G. Dreslinski, M. Wieckowski, D. Blaauw, D. Sylvester, and T. Mudge, "Near-threshold computing: Reclaiming Moore's law through energy efficient integrated circuits," *Proc. IEEE*, vol. 98, no. 2, pp. 253–266, Feb. 2010.

[7] H. Zhang, V. George, and J. M. Rabaey, "Low-swing on-chip signaling techniques: Effectiveness and robustness," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 8, no. 3, pp. 264–272, Jun. 2000.

[8] J. Pangjun and S. S. Sapatnekar, "Low-power clock distribution using multiple voltages and reduced swings," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 10, no. 3, pp. 309–318, Jun. 2002.

[9] F. H. A. Asgari and M. Sachdev, "A low-power reduced swing global clocking methodology," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 12, no. 5, pp. 538–545, May 2004.

[10] C. Sitik and B. Taskin, "Skew-bounded low swing clock tree optimization," in *Proc. 23rd ACM Int. Conf. Great Lakes Symp. VLSI (GLSVLSI)*, May 2013, pp. 49–54.

[11] C. Sitik, E. Salman, L. Filippini, S. J. Yoon, and B. Taskin, "FinFET-based low-swing clocking," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 12, no. 2, Aug. 2015, Art. no. 13.

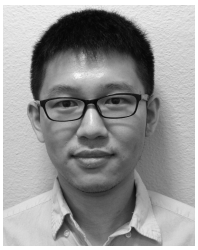
[12] J. R. Tolbert, X. Zhao, S. K. Lim, and S. Mukhopadhyay, "Analysis and design of energy and slew aware subthreshold clock systems," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 30, no. 9, pp. 1349–1358, Sep. 2011.

[13] M. Seok, D. Blaauw, and D. Sylvester, "Clock network design for ultra-low power applications," in *Proc. ACM/IEEE Int. Symp. Low-Power Electron. Design*, Aug. 2010, pp. 271–276.
[14] X. Zhao, J. R. Tolbert, C. Liu, S. Mukhopadhyay, and S. K. Lim, "Variation-aware clock network design methodology for ultra-low voltage (ULV) circuits," in *Proc. ACM/IEEE Int. Symp. Low-Power Electron. Design*, Aug. 2011, pp. 9–14.
[15] C. Sitik, L. Filippini, E. Salman, and B. Taskin, "High performance low swing clock tree synthesis with custom D flip-flop design," in *Proc. IEEE Comput. Soc. Annu. Symp. VLSI*, Jul. 2014, pp. 498–503.
[16] M. Rathore, W. Liu, E. Salman, C. Sitik, and B. Taskin, "A novel static D-flip-flop topology for low swing clocking," in *Proc. ACM/IEEE Great Lakes Symp. VLSI*, May 2015, pp. 301–306.
[17] W. Liu, E. Salman, C. Sitik, and B. Taskin, "Enhanced level shifter for multi-voltage operation," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2015, pp. 1442–1445.
[18] H. Mahmoodi-Meimand and K. Roy, "Dual-edge triggered level converting flip-flops," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2004, pp. II-661–II-664.
[19] *Synopsys 32 nm Generic Library*, Synopsys Inc., Mountain View, CA, USA, 2012.
[20] S. Natarajan *et al.*, "A 32 nm logic technology featuring 2^{nd}-generation high-k + metal-gate transistors, enhanced channel strain and 0.171 μm^2 SRAM cell size in a 291 Mb array," in *Proc. IEEE Int. Electron Devices Meeting*, Dec. 2008, pp. 1–3.
[21] A. Dasdan, E. Salman, F. P. Taraporevala, and K. Kucukcakar, "Characterizing sequential cells using interdependent setup and hold times, and utilizing the sequential cell characterizations in static timing analysis," U.S. Patent 7 506 293, Mar. 17, 2009.
[22] P. Li and E. Acar, "A waveform independent gate model for accurate timing analysis," in *Proc. IEEE Int. Conf. Comput. Design (ICCD)*, Oct. 2005, pp. 363–365.
[23] S. Raja, F. Varadi, M. Becer, and J. Geada, "Transistor level gate modeling for accurate and fast timing, noise, and power analysis," in *Proc. 45th ACM/IEEE Design Autom. Conf. (DAC)*, Jun. 2008, pp. 456–461.
[24] C. Sitik, S. Lerner, and B. Taskin, "Timing characterization of clock buffers for clock tree synthesis," in *Proc. IEEE Int. Conf. Comput. Design (ICCD)*, Oct. 2014, pp. 230–236.
[25] W. C. Elmore, "The transient response of damped linear networks with particular regard to wideband amplifiers," *J. Appl. Phys.*, vol. 19, no. 1, pp. 55–63, 1948.
[26] H. B. Bakoglu, *Circuits, Interconnections, and Packaging for VLSI*. Reading, MA, USA: Addison-Wesley, 1990.
[27] C. V. Kashyap, C. J. Alpert, F. Liu, and A. Devgan, "Closed-form expressions for extending step delay and slew metrics to ramp inputs for *RC* trees," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 23, no. 4, pp. 509–516, Apr. 2004.
[28] K. D. Boese and A. B. Kahng, "Zero-skew clock routing trees with minimum wirelength," in *Proc. 5th Annu. IEEE Int. ASIC Conf. Exhibit*, Sep. 1992, pp. 17–21.
[29] R. Chaturvedi and J. Hu, "Buffered clock tree for high quality IC design," in *Proc. IEEE 5th Int. Symp. Quality Electron. Design*, Mar. 2004, pp. 381–386.
[30] *FreePDK 45 nm Library v1.4*, North Caroline State Univ., Raleigh, NC, USA, 2011.

Can Sitik (S'11) received the B.S. degree in electrical and electronics engineering from Middle East Technical University, Ankara, Turkey, in 2011, and the M.S. and Ph.D. degrees in computer engineering from Drexel University, Philadelphia, PA, USA, in 2013 and 2015, respectively.

He is currently a Software Engineer with Intel Corporation, Hillsboro, OR, USA, where he is involved in integrated circuit power and timing estimation methodologies.

Mr. Sitik received the Best Paper Nomination at the Great Lakes Symposium on VLSI, Paris, France, in 2013.



Weicheng Liu received the B.S. degree in automation from Tsinghua University, Beijing, China, in 2009. He is currently pursuing the Ph.D. degree with Stony Brook University, Stony Brook, NY, USA.

He was with Freescale Semiconductor, Austin, TX, USA, in 2015, where he was involved in low power clock tree synthesis methodologies. His current research interests include custom digital circuit design and VLSI physical design with an emphasis on clock networks and low power clocking.

Baris Taskin (S'01–M'05–SM'12) received the B.S. degree in electrical and electronics engineering from Middle East Technical University, Ankara, Turkey, in 2000, and the M.S. and Ph.D. degrees in electrical engineering from the University of Pittsburgh, Pittsburgh, PA, USA, in 2003 and 2005, respectively.

He was a Staff Engineer with MultiGiG Inc. (now part of Analog Devices), Scotts Valley, CA, USA, from 2003 to 2004, where he was involved in electronic design automation of integrated circuit (IC) timing and clocking. He joined the Department of Electrical and Computer Engineering, Drexel University, Philadelphia, PA, USA, as an Assistant Professor, in 2005, where he is currently an Associate Professor. His current research interests include electronic design automation for VLSI, low-power circuits, resonant clocking, clock network synthesis, wireless IC interconnects, and networks-on-chip for chip multiprocessors.

Dr. Taskin was a recipient of a number of awards for his research and professional contributions, including the ACM SIGDA A. Richard Newton Award in 2007, the National Science Foundation Faculty Early Career Development (CAREER) Award in 2009, an ACM SIGDA Distinguished Service Award in 2012, and the Delaware Valley Young Electrical Engineer of the Year Award from the IEEE Philadelphia Section in 2013. He served on the organizational committees of various conferences that include the Great Lakes Symposium on VLSI (GLSVLSI), the International Symposium on Circuits and Systems, the International Symposium on Low Power Electronics and Design, and the System Level Interconnect Prediction, and technical committees that include GLSVLSI, the International Conference on Computer-Aided Design, and the International Conference on Computer Design. He is an Associate Editor of the *Journal of Circuits, Systems and Computers* and the *Microelectronics Journal* (Elsevier).

Emre Salman (S'03–M'10) received the B.S. degree in microelectronics engineering from Sabanci University, Istanbul, Turkey, in 2004, and the M.S. and Ph.D. degrees in electrical engineering from the University of Rochester, Rochester, NY, USA, in 2006 and 2009, respectively.

He was with STMicroelectronics, Geneva, Switzerland, Synopsys, Mountain View, CA, USA, and Freescale Semiconductor, Austin, TX, USA, where he conducted research in the fields of custom circuit design, timing, and noise analysis. Since 2010, he has been an Assistant Professor with the Department of Electrical and Computer Engineering, Stony Brook University, Stony Brook, NY, USA, where he is currently the Director of the Nanoscale Circuits and Systems Laboratory. He is the Leading Author of a comprehensive tutorial book entitled *High Performance Integrated Circuit (IC) Design* (McGraw-Hill, 2012), which unifies interconnect-centric design methodologies for nanoscale ICs. His current research interests include analysis, modeling, and design methodologies for high performance and energy efficient ICs with an emphasis on power, signal, and sensing integrity.

Prof. Salman received the National Science Foundation Faculty Early Career Development (CAREER) Award in 2013 and the Outstanding Young Engineer Award from the IEEE Long Island, Long Island, NY, USA, in 2014. He serves as the Americas Regional Editor of the *Journal of Circuits, Systems and Computers* and on the organizational/technical committees of various IEEE and ACM conferences. He served on the Editorial Board of the IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION SYSTEMS.