Clock Data Compensation Aware Clock Tree Synthesis in Digital Circuits with Adaptive Clock Generation

Taesik Na, Jong Hwan Ko and Saibal Mukhopadhyay
School of ECE, Georgia Institute of Technology, Atlanta, Georgia, USA
{taesik.na, jonghwan.ko}@gatech.edu, saibal@ece.gatech.edu

***Abstract*—Adaptive clock generation to track critical path delay enables lowering supply voltage with improved timing slack under supply noise. This paper presents how to synthesize clock tree in adaptive clocking to fully exploit the clock data compensation (CDC) effect in digital circuits. The paper first provides analytical proof of ideal CDC effect for ring oscillator based clock generation. Second, the paper analyzes non-ideal CDC effect in a gate dominated critical path and wire dominated clock tree design. The paper shows the delay sensitivity mismatch between clock tree and critical path can degrade CDC effect by analyzing timing slack under power supply noise (PSN). Finally, the paper proposes simple but efficient clock tree synthesis (CTS) technique to maximize timing slack under PSN in digital circuits with adaptive clock generation.**

Keywords—clock data compensation; clock tree synthesis; adaptive clock

I. Introduction

Reducing voltage margin is one of key enablers for energy efficient integrated circuit (IC) design. The voltage margin is added in digital circuits to avoid potential timing error caused by voltage droop due to the limited power delivery network (PDN) resources. One approach to reduce voltage margin is to reduce the voltage drop itself from optimal design of the PDN, adding decoupling capacitors, and using on-chip voltage regulators [1]–[2]. The complementary approach is to implement circuit level techniques that can guarantee correct circuit operation under supply noise [3]–[6]. Among them, adaptive clock generation to track critical path delay has emerged as an efficient approach [5]–[6]. It generates elongated (contracted) clock period matched with critical path delay under a negative (positive) power supply noise (PSN). Many works proposed various clock generation schemes to *accurately* mimic the critical path delay under PSN. However, there is a lack of understanding of how the clock tree design needs to change to maximally exploit the advantages of adaptive clocking.

This paper presents how to synthesize the clock tree in adaptive clocking. In particular, we show that delay sensitivity matching between the clock tree and the adaptive clock generation is necessary to fully benefit from clock data compensation (CDC) effect where clock distribution modulates clock period favorably for timing slack under PSN [1]. We consider ring oscillator (RO) based adaptive clock generation as an example to analytically show the ideal CDC effect. We show that the clock period of RO exactly follows the critical path delay under PSN if and only if the voltage-delay sensitivities of RO, critical path and clock tree are the same *regardless of the clock distribution delay or critical path delay*.

Next, we analyze timing slack under PSN to show that ideal CDC cannot be achieved for designs where the voltage delay sensitivities of the clock tree are not the same with that of critical path. Finally, we propose simple but efficient clock tree synthesis (CTS) technique to maximize CDC effect. As a case study, we implement JPEG encoder engine [7] with Synopsys 28nm PDK and apply CTS changing available buffer list while maintaining the same transition delay. Simulation results show that we can achieve up to 85ps timing slack (equivalent 40mV voltage margin) improvement under 300mV voltage droop with derating factor of 0.9.

The rest of the paper is organized as follows: Section II briefly presents behavioral model of timing slack variation due to PSN; Section III analytically proves ideal CDC effect for RO as a clock generation; Section IV shows non-ideal CDC effect for the practical design; Section V presents timing slack analysis for several design choices; Section VI shows our proposed CTS technique; and Section VII summarizes the paper.

II. Background

Previous work [8] proposed behavioral modeling of timing slack variation due to PSN. We briefly review the previous work which will be used for the proof of ideal CDC for the RO in section III. Given a circuit having relationship between delay variation (Δd) and supply voltage difference (Δv) from nominal voltage, delay sensitivity $\alpha(\Delta v)$ [1/V] can be obtained by [8]:

$$\alpha(\Delta v) = \begin{cases} \frac{1}{D_0}\lim_{\Delta v \to 0}\frac{\Delta d(\Delta v)}{\Delta v}, & \Delta v = 0 \\ \frac{1}{D_0}\frac{\Delta d(\Delta v)}{\Delta v}, & otherwise. \end{cases} \quad (1)$$

where, D_0 is nominal circuit delay from the nominal voltage for the target circuit. The power supply noise induced jitter (PSNIJ) $J(t)$, then can be calculated by [8]:

$$J\big(t; \alpha(\Delta v), D_0, \Delta v(t)\big) = \int_{t-D_0}^{t} \alpha(\Delta v(\tau))\Delta v(\tau) d\tau. \quad (2)$$

where, $\Delta v(t)$ is a transient supply voltage difference from nominal voltage. For brevity, we use $J(t; D_0)$ instead of using

978-3-9815370-8-6/17/$31.00 ©2017 IEEE

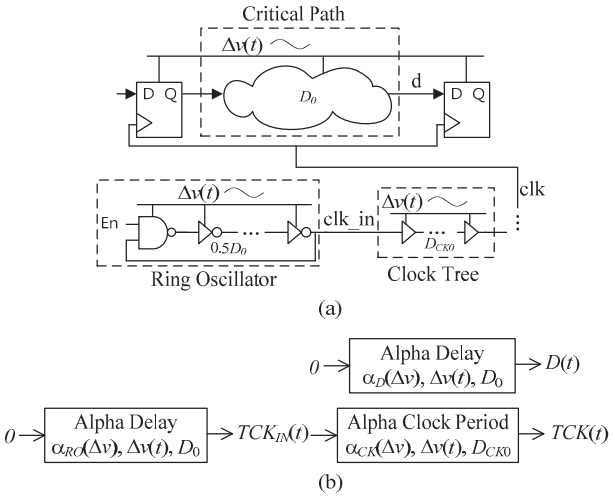

Fig. 1: (a) Block diagram and (b) behavioral modeling [8] for ring oscillator as a local clock generator.

the whole term in (2) and we will use the whole term only when necessary. Power supply noise induced delay (PSNID) $D(t)$ can be calculated by adding PSNIJ ($J(t; D_0)$), propagation of the input delay ($D_{IN}(t)$) and nominal delay (D_0) as follows [8]:

$$D(t) = J(t; D_0) + D_{IN}(t - D_0) + D_0. \quad (3)$$

Now, consider the clock period for a leaf node of the clock tree with the clock distribution delay D_{CK0} from the nominal voltage and the clock period at the source node $TCK_{IN}(t)$. Power supply noise induced clock period (PSNICP) $TCK(t)$ is obtained by adding PSNIJ of the current time and subtracting PSNIJ of the previous rising edge to the propagated input clock period. The PSNICP can be written as [8]:

$$TCK(t) = TCK_{IN}(t - D_{CK0}) + J(t; D_{CK0}) - J(t - TCK_{IN}(t - D_{CK0}); D_{CK0}). \quad (4)$$

Resulting PSN induced slack (PSNIS) $S(t)$ can be written as follows [8]:

$$S(t) = TCK(t) - D(t). \quad (5)$$

III. Ideal Clock Data Compensation

When the clock period exactly follows critical path delay under PSN, $S(t)$ becomes zero and we call this ideal CDC. In this section, we will show inherent ideal CDC effect for a RO as a clock generator. Suppose we have a circuit as shown in fig. 1(a). In this figure, clock is generated from the RO which has half of the critical path delay D_0, making clock period of D_0. The clock propagates through the clock tree with the clock distribution delay of D_{CK0}. Then, the clock meets with critical path delay of D_0 making zero timing slack.

Now we consider PSN, $\Delta v(t)$, and assume on-chip PDN is well designed so that every circuit element is experiencing the same PSN. Fig. 1(b) represents behavioral modeling of this circuit for timing slack variation due to PSN. RO, clock tree and critical path can be modeled as alpha delay, alpha clock period and alpha delay [8] respectively as shown in this figure. We assume every circuit element has the same delay sensitivity $\alpha(\Delta v)$ in this example ($\alpha_D(\Delta v)$, $\alpha_{RO}(\Delta v)$, and $\alpha_{CK}(\Delta v) = \alpha(\Delta v)$). In the following, we use α, α_D, α_{RO} and α_{CK} instead of using $\alpha(\Delta v)$, $\alpha_D(\Delta v)$, $\alpha_{RO}(\Delta v)$ and $\alpha_{CK}(\Delta v)$ for simpler notation.

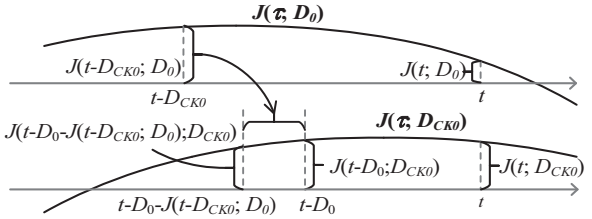

Fig. 2: Waveforms showing $|J(t\text{-}D_0; D_{CK0})\text{-}J(t\text{-}D_0\text{-}J(t\text{-}D_{CK0};D_0);D_{CK0})| \ll D_0$ at a low frequency noise.

The clock period and the critical path are no longer constants under PSN. From fig. 1(b), $TCK_{IN}(t)$ and $D(t)$ can be written as:

$$TCK_{IN}(t) = D(t) = D_0 + J(t; D_0). \quad (6)$$

Making clock period at the leaf node by (4):

$$TCK(t) = D_0 + J(t - D_{CK0}; D_0) + J(t; D_{CK0}) - J(t - TCK_{IN}(t - D_{CK0}); D_{CK0}). \quad (7)$$

We assume the following condition is true for all $t > 0$:

$$|J(t - D_0; D_{CK0}) - J(t - TCK_{IN}(t - D_{CK0}); D_{CK0})| \ll D_0$$

At a low frequency noise, even though PSNIJ itself can be large, $|J(t\text{-}D_0; D_{CK0})\text{-}J(t\text{-}D_0\text{-}J(t\text{-}D_{CK0};D_0);D_{CK0})|$ is small due to the slow supply voltage change as shown in fig. 2. This assumption is also true at a higher noise frequency since the peak values of PSNID and PSNICP become negligible at a higher frequency. Note that integration over a fixed interval on a sin function becomes smaller at a higher noise frequency. Now, $TCK(t)$ can be re-written as:

$$TCK(t) = D_0 + J(t - D_{CK0}; D_0) + J(t; D_{CK0}) - J(t - D_0; D_{CK0}). \quad (8)$$

Ideal CDC can be proved by showing (6) and (8) are the same.

A. Critical Path Delay < Clock Distribution Delay

First, we consider the case where critical path delay (D_0) is shorter than clock distribution delay (D_{CK0}). By additivity principle of integration on intervals, $J(t; D_{CK0})$ in (8) can be written as:

$$J(t; D_{CK0}) = \int_{t-D_0}^{t} \alpha\Delta v(\tau)d\tau + \int_{t-D_{CK0}}^{t-D_0} \alpha\Delta v(\tau)d\tau = J(t; D_0) + J(t - D_0; D_{CK0} - D_0).$$

Similarly, $J(t\text{-}D_0; D_{CK0})$ in (8) can be decomposed as:

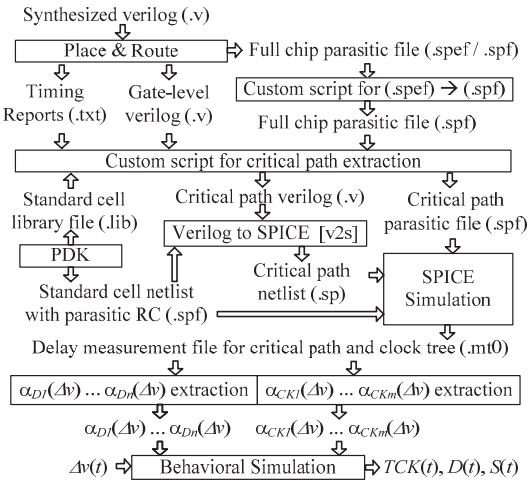

Fig. 3: Design flow.

$$J(t - D_0; D_{CK0}) = J(t - D_0; D_{CK0} - D_0) + J(t - D_{CK0}; D_0).$$

Substituting the above two terms for (8) gives:

$$TCK(t) = D_0 + J(t; D_0). \quad (9)$$

which is same as (6) resulting in ideal CDC effect.

B. Critial Path Delay > Clock Distribution Delay

Secondly, we consider the case where critical path delay (D_0) is longer than clock distribution delay (D_{CK0}). We apply additivity principle of integration on intervals again for $J(t; D_{CK0})$ in (8):

$$J(t; D_{CK0}) = J(t; D_0) - J(t - D_{CK0}; D_0 - D_{CK0}).$$

Note that the integration for an interval $[t\text{-}D_{CK0}, t]$ can be done by integration for $[t\text{-}D_0, t]$ minus $[t\text{-}D_0, t\text{-}D_{CK0}]$ since $D_0 > D_{CK0}$. Similarly, $J(t\text{-}D_0; D_{CK0})$ in (8) can be re-written as:

$$J(t - D_0; D_{CK0}) = J(t - D_{CK0}; D_0) - J(t - D_{CK0}; D_0 - D_{CK0})$$

Substituting the above two terms for (8) gives:

$$TCK(t) = D_0 + J(t; D_0). \quad (10)$$

resulting in the same function with (6).

This is notable that ideal CDC effect *does not depend on the relationship between critical path delay and clock distribution delay*.

IV. Non-Ideal Clock Data Compensation

In real implementation, delay sensitivities for the critical path and clock tree are not the same. Normally, clock tree is wire dominated delay and critical path is gate dominated delay, thus, delay sensitivity of the clock tree is smaller than that of

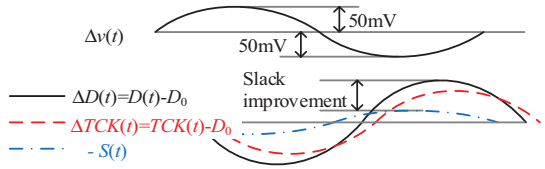

Fig. 4: Waveforms showing slack improvement is measured by max{ΔD(t)}– max{-S(t)}.

the critical path delay [1]. To examine the effect of different delay sensitivities, we consider the same circuit in fig. 1 and set the delay sensitivities α_D, α_{RO} and α_{CK} for critical path, RO and clock tree respectively as shown in fig. 1(b). $TCK_{IN}(t)$, $D(t)$ and $TCK(t)$ now can be written as:

$$TCK_{IN}(t) = D_0 + J(t; \alpha_{RO}, D_0) \quad (11)$$

$$D(t) = D_0 + J(t; \alpha_D, D_0) \quad (12)$$

$$TCK(t) = D_0 + J(t - D_{CK0}; \alpha_{RO}, D_0) + J(t; \alpha_{CK}, D_{CK0}) - J(t - D_0; \alpha_{CK}, D_{CK0}). \quad (13)$$

If we apply similar decomposition and substitution techniques like previous section, we obtain:

$$TCK(t) = D_0 + J(t; \alpha_{CK}, D_0) + J(t - D_{CK0}; \alpha_{RO}, D_0) - J(t - D_{CK0}; \alpha_{CK}, D_0). \quad (14)$$

Note the result in section III is the special form of (14) when $\alpha_D = \alpha_{RO} = \alpha_{CK}$. If we set $\alpha_{RO} = \alpha_{CK}$, then the equation (14) can be modified to:

$$TCK(t) = TCK_{IN}(t) = D_0 + J(t; \alpha_{CK}, D_0). \quad (15)$$

This means that $TCK(t)$ is not dependent on D_{CK0} resulting in no clock period modulation through the clock tree. We refer this effect as *non-clock tree modulation* and ($\alpha_{RO} = \alpha_{CK}$) as the condition for non-clock tree modulation .

V. Frequency Domain Timing Slack Analysis

In this section, we perform frequency domain timing slack analysis for 3 test cases to understand non ideal CDC effect. We use the behavioral modeling in fig. 1(b).

- $\alpha_{RO} = \alpha_{CK} = \alpha_D$: The RO and the clock tree are both gate dominated. This is the ideal CDC case and resulting timing slack can be can be obtained by substituting parameters (12) and (14) for (5):

$$S(t) = 0. \quad (16)$$

- $\alpha_{RO} = \alpha_{CK} = 0.8*\alpha_D$: The RO and the clock tree are both wire dominated. Timing slack for this case is:

$$S(t) = -J(t; 0.2\alpha_D, D_0). \quad (17)$$

- $\alpha_{RO} = \alpha_D$, $\alpha_{CK} = 0.8*\alpha_D$: This is the most conventional case where RO generated clock period tracks critical path delay,

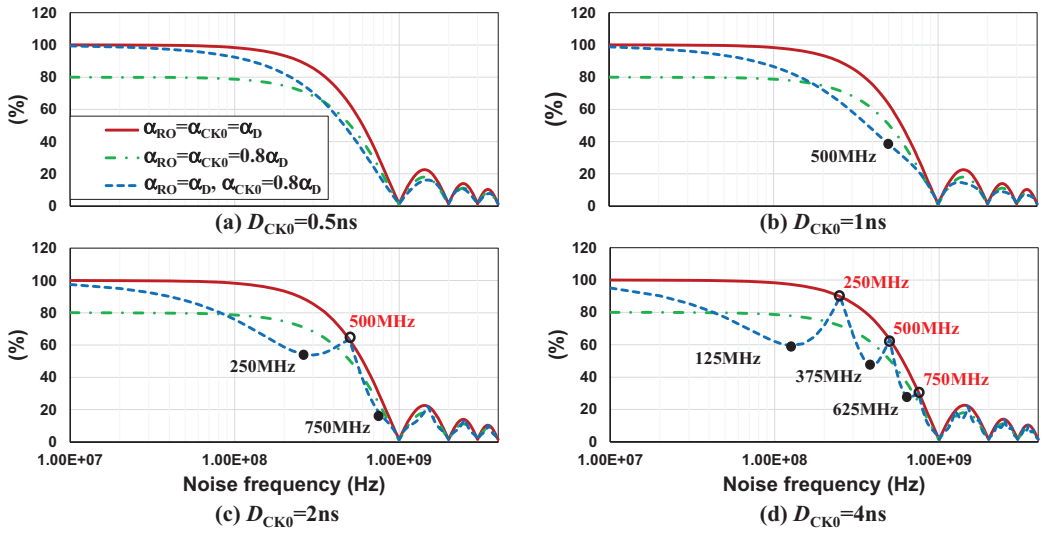

Fig. 5: Normalized slack improvement for D_0=1ns, (a) D_{CK0}=0.5ns (b) D_{CK0}=1ns (c) D_{CK0}=2ns and (d) D_{CK0}=4ns.

but the clock tree is wire dominated and resulting timing slack is:

$$S(t) = -J(t; 0.2\alpha_D, D_0) + J(t - D_{CK0}; 0.2\alpha_D, D_0). \quad (18)$$

A. Simulation Setup

We design JPEG encoder engine [7] with Synopsis 28nm PDK (supply voltage of 1.05V). We use Design Compiler and IC Compiler for synthesis and place & route respectively. After obtaining critical path netlist with our custom scripts, we convert Verilog netlist to SPICE compatible netlist and simulate this with parasitic RC to obtain delay sensitivity $\alpha_D(\Delta v)$ for the critical path as shown in fig. 3. Then, we build behavioral model for each test case and apply 100mV peak-to-peak sinusoidal supply noise $0.05sin(2\pi ft)$ by sweeping noise frequency (f). We perform simulations varying D_{CK0} = 0.5ns, 1ns, 2ns and 4ns while fixing D_0 = 1ns. We measure the timing slack improvement as shown in fig. 4:

$$\text{slack\_impr} = max\{\Delta D(t)\} - max\{-S(t)\} \quad (19)$$

where $\Delta D(t) = D(t) - D_0 = J(t;\alpha_D, D_0)$ and $-S(t)$ is negative timing slack. Then the slack improvement can be maximized when $S(t)$=0 resulting *maximum achievable* slack improvement:

$$\text{max\_achievable\_slack\_impr} = \max\{\Delta D(t)\} \quad (20)$$

We normalize the measured timing slack improvement by dividing delay increase under 50mV (half of the peak-to-peak noise) DC supply voltage drop. The delay increase can be obtained by substituting -50mV for $\Delta v(t)$ in (2) resulting in $-0.05\{\alpha_D(-0.05)\}D_0$. Note that the above value is positive value since the typical delay sensitivity is negative value (delay increases as supply voltage decreases). Then the normalized slack improvement can be the function of f:

$$\text{norm\_slack\_impr}(f) = 100 \times \frac{max\{\Delta D(t)\} - max\{-S(t)\}}{-0.05\{\alpha_D(-0.05)\}D_0}\Bigg|_{\Delta v(t)=0.05sin(2\pi ft)} \quad (21)$$

B. Simulation Results

Fig. 5 shows simulation results.

- $\alpha_{RO} = \alpha_{CK} = \alpha_D$: As expected, this case shows maximum achievable slack improvement for all the noise frequencies since $S(t)$=0 as in (16). Normalized slack improvement in this case is:

$$\text{norm\_slack\_impr}(f) = 100 \times \frac{max\{J(t;\alpha_D,D_0)\}}{-0.05\{\alpha_D(-0.05)\}D_0}. \quad (22)$$

We omit $\Delta v(t) = 0.05sin(2\pi ft)$ for brevity. Note that $J(t; \alpha_D, D_0)$ becomes zero at every $f = n/D_0$ (n is positive integer) since the integration of the sinusoidal function becomes zero between t-D_0 and t for $f = n/D_0$ (n is positive integer). Also, $max\{J(t; \alpha_D, D_0)\}$ decreases as the noise frequency becomes higher since the integration over D_0 on a sin function becomes smaller at a higher noise frequency. In this case, slack improvement is just a function of f, α_D and D_0 not D_{CK0} since there is no clock period modulation through the clock tree (non-clock tree modulation) as $\alpha_{RO} = \alpha_{CK}$. Note that the graphs for this case in fig. 5(a)–(d) are identical regardless of D_{CK0}.

- $\alpha_{RO} = \alpha_{CK} = 0.8*\alpha_D$: Normalized slack improvement in this case can be obtained by substituting (12) and (17) for (21):

$$\text{norm\_slack\_impr}(f) = 100 \times \frac{max\{J(t;0.8\alpha_D,D_0)\}}{-0.05\{\alpha_D(-0.05)\}D_0}. \quad (23)$$

This shows that we can only achieve 80% of maximum achievable slack improvement as shown in fig. 5. The clock period modulation ($\Delta TCK_{IN}(t) = TCK_{IN}(t) - D_0 = J(t; \alpha_{RO}, D_0)$) from the RO is 80% of $J(t; \alpha_D, D_0)$ due to $\alpha_{RO} = 0.8*\alpha_D$. And the generated clock is, then, delivered to the leaf node without clock tree modulation since $\alpha_{RO} = \alpha_{CK}$. The graphs for this case are also identical regardless of D_{CK0} due to non-clock tree modulation.

- $\alpha_{RO} = \alpha_D$, $\alpha_{CK} = 0.8*\alpha_D$: Normalized slack improvement in this case can be obtained by substituting (12) and (18) for (21):

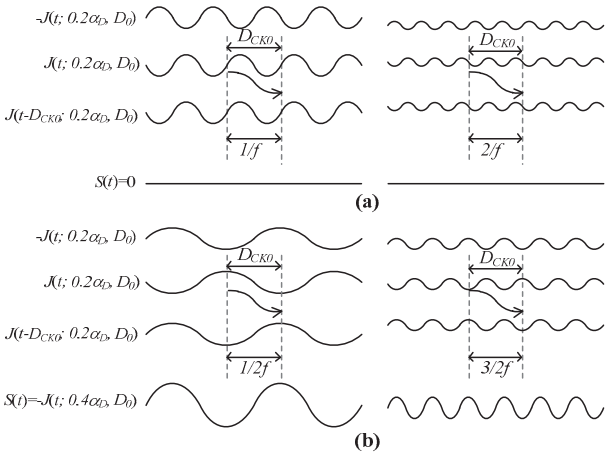

Fig. 6: (a) $S(t) = 0$ at every $f = n/(D_{CK0})$ (n : positive integer) (b) $S(t) = -J(t; 0.4\alpha_D, D_0)$ at every $f = n/(2D_{CK0})$ (n : positive *odd* integer) for $\alpha_{RO} = \alpha_D$, $\alpha_{CK} = 0.8*\alpha_D$.

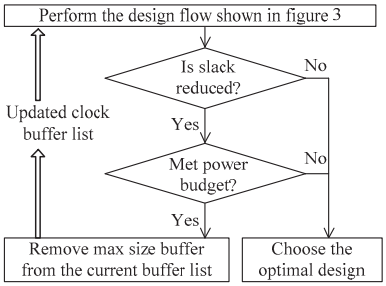

Fig. 7: Decision tree to choose the optimal design.

$$\text{norm\_slack\_impr}(f) = 100 \times \frac{max\{J(t;\alpha_D,D_0)\} - max\{J(t;0.2\alpha_D,D_0) - J(t-D_{CK0};0.2\alpha_D,D_0)\}}{-0.05\{\alpha_D(-0.05)\}D_0}. \quad (24)$$

Interesting observations can be found in this case. First, this case shows 100% slack improvement at DC noise. Since α_{RO} is the same with α_D, PSNICP generated from the RO exactly follows PSNID of the critical path. At DC supply voltage droop, clock period modulation through the clock tree is negligible, thus, it achieves 100% slack improvement.

Second, the slack improvement can be maximized at every $f = n/D_{CK0}$ (n is positive integer) since $S(t)$ becomes 0. The two terms, $J(t; 0.2\alpha_D, D_0)$ and $J(t- D_{CK0}; 0.2\alpha_D, D_0)$ in (18), are cancelled out at these frequencies as shown in fig. 6(a). Empty circles marked in fig. 5 show this phenomenon.

Third, the slack improvement is reduced by 40% from the maximum achievable slack improvement at every $f = n/(2D_{CK0})$ (n is positive *odd* integer). As shown in fig. 6(b), at every $f = n/(2D_{CK0})$ (n is positive *odd* integer), the amplitude of timing slack is maximized, thus, resulting in $S(t) = -J(t; 0.4\alpha_D, D_0)$. Slack improvement, then, becomes $max\{J(t; 0.6\alpha_D, D_0)\}$ which is 60% of maximum achievable slack improvement. The frequencies in this condition are marked with filled circles in fig. 5. This is interesting in that slack improvement can be even lower than $\alpha_{RO} = \alpha_{CK} = 0.8*\alpha_D$ case around these frequencies.

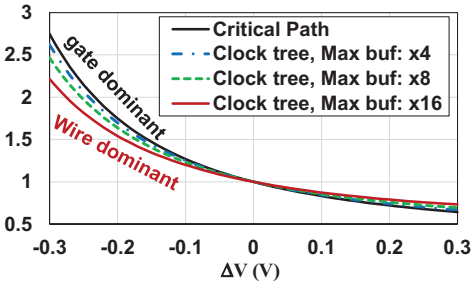

Fig. 8: Normalized delay.

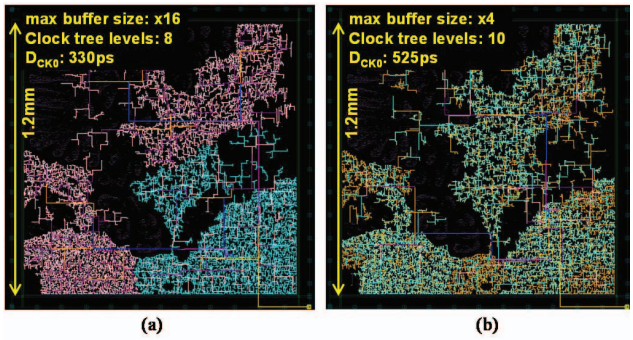

Fig. 9: IC compiler snapshot showing clock tree for JPEG encoder for max buffer size is (a) x16 and (b) x4

Also note that these frequencies become lower when clock distribution delay D_{CK0} becomes larger. This is noteworthy as clock distribution delay tends to increase in today's high frequency microprocessors and significant first droop noise occurs at around 200Mhz [1]. Therefore, it suggests that clock generation only mimicking the critical path delay does not guarantee optimum solution. In order to get fully benefit from CDC effect, delay sensitivity of clock tree should be well matched with that of critical path delay.

VI. CDC Aware Clock Tree Synthesis

In this section, we propose simple CDC aware CTS algorithm to make gate dominant clock tree design in standard design flow. We take the same design with the previous section, but use measured delay sensitivity for clock tree (α_{CK}) instead of assuming $\alpha_{CK} = \alpha_D$ or $\alpha_{CK} = 0.8*\alpha_D$ like previous session. We also use measured D_0 and D_{CK0}. For adaptive clock generation, we assume clock is being generated with the same delay sensitivity of the critical path delay. Note, design of adaptive clock generation is beyond the scope of this paper.

A. CDC Aware CTS

To change delay sensitivity of the clock tree, we propose to control available clock buffer list for CTS while maintaining maximum transition parameter (*max_transition*) unchanged during place and route. We initially perform the design flow as in fig. 3 with normal clock buffer list and check the timing slack and power budget given PSN and derating factor. And we remove maximum strength buffer one by one and perform the design flow again with updated clock buffer list until the timing slack tends to degrade and power consumption for the clock tree violates power budget. Entire decision tree is shown in fig. 7.

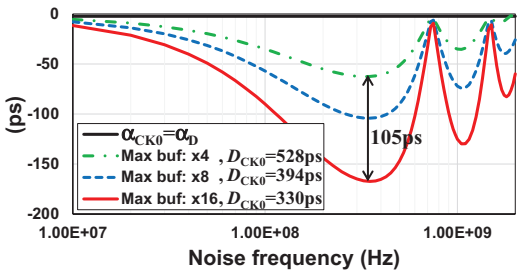

Fig. 10: Timing slack with 300mV peak-to-peak voltage with D_0= T_{CK0}=1.35ns

B. Implementation Results

JPEG encoder is implemented with buffer list {x1, x2, x4, x8, x16} first and then with {x1, x2, x4, x8}, {x1, x2, x4} and {x1, x2} in 28nm Synopsys PDK (supply voltage = 1.05V). The implementation results show D_0 = TCK_0 = 1.35ns. Fig. 8 shows normalized delay for critical path delay and clock tree distribution delay when the maximum buffer size is limited to x16, x8 and x4. We observe that clock tree becomes wire dominated as we use large buffers. CTS with small sized buffers gives longer clock distribution delay since the depth of the clock tree becomes larger as shown in fig. 9.

Fig. 10 shows timing slack when we apply 300mV peak-to-peak sinusoidal PSN. We observe that over 100ps of timing slack is improved by simply synthesizing clock tree with limited available clock buffer list.

Fig. 11 shows minimum timing slack over the noise frequencies varying peak-to-peak voltage of sinusoidal PSN with various derating factors. We see that timing slack for limited buffer list degrades more than normal case with increased derating factor since the clock distribution delay for limited buffer list is longer than that for the normal buffer list. Thus, optimal design should be chosen based on PSN and derating factor. CTS with normal buffer list gives optimum design for lower PSN while CTS with limited buffer list give better design for larger PSN given derating factor.

Table I summarizes clock tree implementation results for various clock buffer list cases. As we limit maximum buffer size, clock tree distribution delay, skew, clock tree level and number of clock tree buffers tend to increase as expected. We also observe changing clock buffer size doesn't impact much on the power consumption except x2 case. Even if we use large number of clock buffers with limited clock buffer list, power consumption doesn't change much since unit power consumption of small size buffer is lower than that of large size buffer offsetting increased number of clock buffers. This shows we can make gate dominant clock tree design without additional power overhead by carefully using our proposed technique.

TABLE I
CLK TREE IMPLEMENTATION SUMMARY

Max buffer size	x16	x8	x4	x2
Longest Path (ps)	330	394	528	703
Skew (ps)	68	114	107	153
# total level	8	8	10	19
# CT Buffers	472	475	826	2039
Power (mW)	19.8	19.4	19.3	21.5

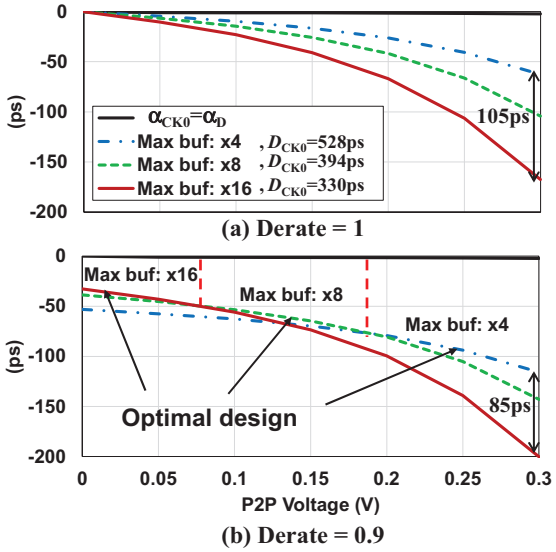

Fig. 11: Timing slack varying peak-to-peak voltage with derating factor is (a) 1 and (b) 0.9

VII. Conclusion

This paper addresses careful clock tree design is needed for adaptive clock generation to maximize CDC effect. To this end, first, we analytically show how the timing slack is impacted by the clock tree design in adaptive clocking for both ideal case and generalized case. Second, frequency domain timing slack analysis for several cases is provided. And finally, we propose simple and efficient way to control delay sensitivity for clock tree to maximize the CDC effect. Synopsys 28nm implementation results for JPEG encoder show 85ps timing slack (equivalent 40mV voltage margin) improvement given 300mV voltage droop and derating factor of 0.9. Our observation is important in that the clock tree distribution delay is becoming longer and it is wire dominated while most of the critical path delay is gate dominated.

Acknowledgment

This material is based on work supported in part by National Science Foundation under grant CCF#1417323.

References

[1] K. L. Wong et al., "Enhancing microprocessor immunity to power supply noise with clock-data compensation," *JSSC*, 2006.

[2] N. Kurd et al., "Haswell: A family of IA 22 nm processors," *JSSC*, 2015.

[3] S. Das et al., "RazorII: In Situ Error Detection and Correction for PVT and SER Tolerance," *JSSC*, 2009.

[4] K. Chae et al., "A Dynamic Timing Error Prevention Technique in Pipelines With Time Borrowing and Clock Stretching," *TCAS*, 2014

[5] A. Grenat, et al., "Adaptive Clocking System for Improved Power Efficiency in a 28nm x86-64 Microprocessor," *ISSCC*, 2014.

[6] J. Kwak and B. Nikolic, "A Self-Adjustable Clock Generator With Wide Dynamic Range in 28 nm FDSOI," *JSSC*, 2016

[7] OpenCores [Online]. Available: http://opencores.org/

[8] T. Na and S. Mukhopadhyay, "Behavioral Modeling of Timing Slack Variation in Digital Circuits Due to Power Supply Noise", *DATE*, 2016.