International Journal of Contemporary Research and Literacy Works (IJCRL), Vol. 7, No. 1, January 2026, pp. 1–24.

Copyright © 2026 Author(s) retain the copyright of this article. This article is published under the terms of the Creative Commons Attribution Liscense 4.0.

Clock Distribution Optimization Using CCD CTS in a Power-Constrained Design

Harshith K.S[1], Gadipelly Dharanidhar[1]
A.S.V.N Anjaneya Seshasai[2], Vijay Vallabhuni[2,*]

[1] *Silica Launch, Bangalore, India, 560048*
[2] *AstraSilica Technologies, Bangalore, India, 560048*
harshith7ks@gmail.com[1], *dharanidhargadipelly@gmail.com*[1]
seshasai148@gmail.com[2], *vvijay@astrasilica.com*[2,*]

(*Received 11 August 2025; Revised 30 October 2025; Accepted 18 December 2025; Published 01 January 2026*)

Abstract

Clock distribution is among the critical considerations in application specific integrated circuit (ASIC) design nowadays, and it is a significant factor in power consumption and timing performance. Clock tree occupies nearly three quarters of the total dynamic power in power-constrained designs, and this implies that it requires some original techniques to optimize it. This paper will discuss a detailed study of the clock distribution optimization in a 28 nm technology node, which is having stringent power limitations, in the Concurrent Clock and Data (CCD) based Clock Tree Synthesis technique. We run our strategy on a high-complexity design block with 65,946 flattened cells and 12,647 hierarchical instances with a clock frequency of 454.5 MHz and clock period of 2.2ns. Through a systematic use of the CCD-CTS methodology, we can demonstrate a massive improvement in the quality measures of clock trees and at a reasonable power cost. We determined that CCD-based CTS achieves clock repeater counts nearly 48 percent less than conventional techniques with clock latency of 0.26 ns and controlled skew of 0.15 ns. The findings demonstrate that CCD-CTS is a useful methodology to employ in the attempt to apply timing closure in power-constrained settings without affecting the quality measures of designs on multiple optimization objectives.

Keywords: Clock Tree Synthesis; Concurrent Clock and Data Optimization; CCD CTS; Clock Latency; Clock Skew; Low-Power Design; Physical Design; 28nm Technology.

1 Introduction

The clock tree synthesis has become one of the most important phases of contemporary VLSI physical design, which essentially affects the realization of timing closure along with the control of power usage and area overhead. The paper describes an all-inclusive approach to clock distribution network optimization in power-constrained

design using Concurrent Clock and Data (CCD) optimization tools throughout Clock Tree Synthesis stage. We show that CCD-based CTS reduces the number of clock buffers (down to 171 cells) by 47.5% (326 cells to 171 cells), and also controls count of clock buffers with 12,647 instances and operating frequency of 454.5 MHz. The most important innovation is the combination of practical skew optimization and strategic choice of the clock cells to result in multi-objective optimization of the power, performance, and area measures. The work offers practitioners an insight into the current CTS techniques and offers a design flow that can be used in sophisticated technology applications where power consumption is a new first-class constraint in addition to timing performance (Datli, A. *et al.* 2013).

1.1 Role of Clock Distribution in High-Performance ASICs

One of the basic building blocks is clock distribution network, which is one of the building blocks of the physical implementation of high-speed and high-density application-specific integrated circuits. Recent designs Recent designs In recent designs, the clock signal is driven out of a central clock source by an intricate network of buffers, inverters and interconnects to thousands of sequential elements distributed over the silicon die (Han, K*et al.* 2018). The quality of such a distribution is the direct determinant of the clock frequency and power dissipation and overall timing closure capability. The clock tree can be a significant power drain in current ASIC design, and the literature has demonstrated that in certain operating conditions clock networks can be as much as three-quarters of the system dynamic power. This kind of clock power over total power consumption demands a high level of optimization at every level of physical design hierarchy. The importance of clock distribution is further escalated in the high-technology nodes whereby the delays in wire are growing in importance and process variation causes unpredictable timing effects (Vishnu, P. V. *et al.* 2019).

1.2 Challenges of CTS in Power-Constrained Designs

Power-constrained ASIC design is a particular problem in the conventional clock tree synthesis methodologies. Reduction of clock skew and insertion latency is one of the primary objectives that is typically attained through higher buffer insertion that is directly proportional to the power consumption (Shan, W. *et al.* 2019). In traditional methodology, a sequential optimization framework is used where data path optimization is completed in the initial design stage, followed by clock tree synthesis in subsequent design stages.

There are several inherent problems with this method: the clock tree is not constructed in a way that permits one to view the actual timing requirements of the data paths, leading to over-conservative skew targets and excessive buffering; timing surprises discovered in the end of the design flow cannot be fixed with an expensive iteration and timing closure refinement cycle; and the clock tree is fixed once the synthesis has occurred, and can no longer be reconfigured to better utilize

the data paths (Anirudh, S. *et al.* 2022). These sequential optimization constraints are essential bottlenecks in timing closure in designs that are founded on advanced technology nodes with power-constrained designs with aggressive clock frequencies.

1.3 Limitations of Conventional CTS Approaches

The current tools of clock tree synthesis are primarily focused on the minimization of skew by building balanced clock networks (Li, W.*et al.* 2024). This makes implementation easier and deterministic behavior is possible, but in effect this removes any useful exploitation of clock skew as a valuable optimization resource (Lu, Y. C. *et al.* 2023). The conventional approaches also have problems with multi-objective optimization in power-constrained designs where power, area, latency, and skew are required to be optimized with each other and within complex design constraints. The physical design tools used in traditional physical design methods have sequential physical design flows meaning that decisions made in the placement-stage, without a complete understanding of the clock tree, may introduce a poor clock distribution topology, which cannot be optimally fixed in the CTS stage other than by significant rework. Moreover, the conventional methods also have inadequate capabilities to accommodate local variations in timing requirements in the data paths in various regions of the design (Wu, J. *et al.* 2025).

The article is an in-depth case study of clock distribution optimization of a 28 nm and power-constrained ASIC block through Concurrent Clock and Data based Clock Tree Synthesis. Our significant contributions are to have demonstrated that the CCD-CTS methodology can be successfully employed in the implementation of a physical design tool flow that is industry standard in both its practical implementation and, second, we have demonstrated that CCD-CTS methodology can reduce the number of clock repeaters by a factor of about forty-eight percent, compared to the conventional methods, and, finally, we have reported timing analysis results which indicate that CCD-CTS methodology can achieve timing closure at 454.5 MHz and still meet all power constraints. Our experience demonstrates the applicability of the concurrent optimization strategies to the resolution of the major problems of the contemporary, power-limited ASIC design.

2 Design Specification and Constraints

2.1 Design Specifications and Description

The design in the given work is a high-complexity subsystem with hard time constraints that is a complete functional unit. The core area of the block is a square with an area of 514,863.594 sq. mm and this is a very symmetric starting point of clock tree distribution. It was designed with 2,090 distinct library cells, instantiated in 12,647 hierarchical instantiations and 65,946 flattened cells as is typical of high logic density of modern ASICs (Li, W.*et al.* 2024). The design has 428 input/output ports comprising of thirty six input ports, three hundred and eighty eight output ports and two inout port implying that it has a broad connection

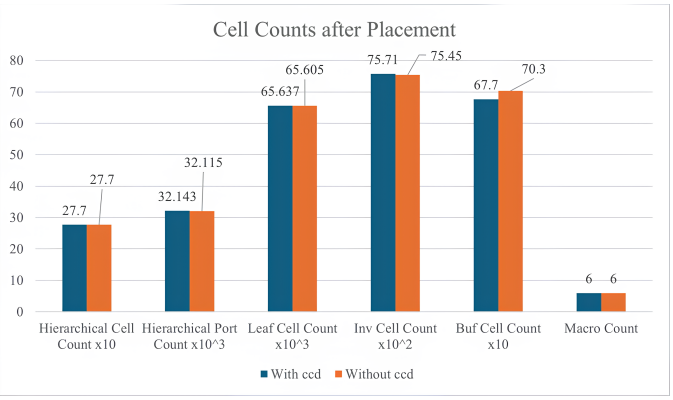

Fig. 1. Cells count after placement.

with other design blocks and principal inputs/outputs. The relatively high number of ports means that the clock tree insertion and the data path timing must be considered in particular so as to prevent excessive peripheral skew.

2.2 Design Parameters $28nm, 2.2ns$

It was a block in a 28 nm CMOS process technology, an advanced node, with well-built design rules, comprehensive cell libraries and production processes that have been well developed. The design target has clock frequency of 454.5 MHz which is equal to 2.2 ns clock period. The principal clock signal was called clk with a period of 2.2 ns and a symmetric waveform with a fifty percent duty cycle (waveform edge defined at 0 ns and 1.1 ns). The timing of the Hybrid timing closure setup has a setup uncertainty (0.33 ns) i.e. the setup timing is on the clock jitter, skew and uncertainty margin, which is required to run well at process voltage and temperature corners. It also had 9 metal layers (M1 to M9) of signal routing and provided much routing capacity to both data and clock networks.

Input timing constraints specified input delay of 1.32 ns compared to output timing constraints out of which there was an output delay of 1.32 ns compared to the primary clock. A timing analysis of combinational paths was done at a virtual clock with the same period (2.2 ns) to time paths and input delay constraints were applied to path timing at a virtual clock of 0.77 ns relative to the primary clock distribution.

2.3 Power Aware Design Requirements

The power constraints of the system as a whole were used to base strict power budgets of the block. The limitations of power above imposed made the clock tree design very critical as the clock network is the highest power consumer in the design. This required the allocation of power budget since the clock distribution network had to consume very minimal dynamic power in order not to impact timing

performance at the aggressive operating frequency of 454.5 MHz. It is this power limit that was the direct reason behind the need of new optimization methods over and above the traditional balanced clock tree synthesis methods since the traditional methods oversize the clock network to provide maximum skew reduction with no consideration of the real timing margins in the data paths (Mo, X.*et al.* 2021).

2.4 Clock Network Design Objectives

The clock specification requirements of the clock network in the block had a few, sometimes, conflicting requirements: to minimize clock delay to minimize timing margin loss; to minimize clock skew to ensure that the timing behavior can be predictable at all corners; to minimize clock network area; to minimize dynamic power consumption. Others were the achievement of design rule compliance that was in terms of maximum transition times, maximum capacitive loads and fanout limits per stage of drivers. The successful achievement of these different objectives came with the need to adopt sophisticated optimization methods that brought about trade-offs among conflicting actions.

3 CCD Based CTS Methodology

3.1 Overview of Concurrent Clock and Data Optimization

Parallel Clock and Data optimization, A radical innovation in the sequential physical design flows of the past. Traditional techniques In traditional techniques, physical design is performed in independent stages: placement optimization is performed to find the locations of cells, without knowledge of the clock tree structure (Shan, W. *et al.* 2019); clock tree synthesis is performed to construct the clock distribution network using known placement; and timing optimization is performed to adjust data paths, and add buffers to meet timing constraints (Anirudh, S. *et al.* 2022). This sequential model possesses the intrinsic potential to restrict the opportunities of optimization because each step has incomplete information about the subsequent step.

Under CCD optimization, the natural interdependence between clock and data path timing is identified and their optimization is viewed within a single framework. In CCD, post-clock tree synthesis timing optimization is a two-way tradeoff between clock network optimization (taking advantage of convenient skew manipulation and buffer sizing) and data path optimization (taking advantage of gate sizing, placement optimization and interconnect optimization). This concurrent example gives the optimization framework the opportunity to take advantage of desirable clock skew relations in which the skewing of clock networks calculated can be purposefully injected to facilitate logic paths otherwise unable to meet timing constraints to do so (Lu, Y. C. *et al.* 2023).

3.2 CTS Constraints and configuration

The implementation of the blocks clock tree synthesis was designed in a manner that it reached a balanced tradeoff of a number of optimization objectives, without going against the technology constraints. The maximum insertion latency limits were configured to 0.30 ns providing decent clock delay budget and permitting insertion of repeaters to be applied to deal with signal integrity. The original targets of clock skews were 0.10ns which is a balanced tree constraint, which would be relaxed by CCD optimization in the event that any useful skew relationships were found. Signal integrity and reliability of power delivery CTS tool would have a limit of thirty two on fanout of the buffer driver stages and a limit of 0.25ns on transition time of the clock networks.

Clock routing was restricted to avoid congested regions that had been identified during the optimization of the placement and to monitor manufacturing design limitations, including minimum spacing limitations between signal lines and between signal and power lines. The clock tree building algorithm will be configured to employ a hierarchical multi-level-method, which would build clock distribution network between the clock origin and multiple buffering levels to all clock sinks that are distributed throughout the design (11,887 clock sinks).

3.3 NDR Selection to Clock Routing

Clock routing was governed by Non-Default Rules (NDR) to impose more rigorous design discipline on signal routing than on general signal routing to provide high-quality signal integrity and timing predictability. The chosen NDR of clock routing used larger spacing of wires than the normal spacing rules minimized the crosstalk coupling between the signal lines. Also, the clock routing NDR imposed a minimum width of clock distribution paths, which decreased parasitic resistance and the resulting clock propagation delay. Clock routing preferred metals were based upon examination of early placement, with lower metal layers (M2 and M3) being used to distribute local clocking around clock sinks, and higher metal layers (M7 and M8) being used to distribute clocking network segments of global clocking over longer distances (Wu, J. *et al.* 2025). The NDR selection process has been keen to trade off better signal integrity with more area consumption and routing congestion since too much routing resources allocated to clock networks may lead to routing congestion issues in data path routing.

3.4 Integration with Power-Aware Cell Selection

The CCD-CTS methodology power-conscious computation was done by selecting clock buffer and inverter cells with care out of the technology library. The offered technology library of 28 nm offered several clock buffer models that were optimized to various drive strengths, threshold voltages and power properties. In non-critical areas of the clock tree, it was possible to select cells having large threshold voltages (both standard Vt and high Vt versions) to minimize subthreshold leakage current,

whereas in other regions of the clock tree (such as critical clock paths) standard Vt cells were used to provide timing margin. The dynamic power reduction the technique was to choose minimum-size clock cell that could satisfy the target drive requirements instead of excessive timing margins by over-sizing clock cells.

The method used to select the clock cell was a method that gave a feedback in terms of timing analysis, and the cell whose size was smaller than needed, was only up-sized where it was needed to remain within timing constraints or drive capability limits. This cell selection scheme was power-conscious, together with convenient skew optimization made possible by the CCD methodology, yielded a middle ground in which the power-consumption of clock trees was actively reduced without reducing timing closure (Mo, X.*et al.* 2021).

4 Clock Cell Selection Strategy

4.1 Characteristics of Clock Buffers and Clock Inverters

The clock buffers and inverters are the major active devices in the clock distribution network and they play important roles in signal regeneration and propagation delays. The main option in the majority of clock tree applications is clock buffers that preserve the phase relationship between the input clock signal. Nevertheless, clock inverters, which reverse the phase of the input signal, can be of great benefit in clock tree design when the phase-inverted clock signals are needed by sequential elements or when they offer better timing or power behavior to particular tree nodes. Clock buffers designed to optimize clock distribution, this is normally the case with clock buffers designed to be symmetric in both rise and fall times so that the clock duty cycle is balanced at each tree stage, important in avoiding loss of timing margin due to clock asymmetry. The technology library of 28 nm in this work offered the clock buffer variants of different drive strengths, with the minimum-strength designs consuming low power and having limited drive capability, and the maximum-strength designs able to drive large capacitive loads, but consuming large power (Pasca, A. *et al.* 2016).

4.2 Criteria for Buffer vs Inverter Usage

The choice to use clock buffers at every clock tree level or inverters was guided by several factors that were considered concurrently. Timing margin requirements were also used as primary criteria, with paths whose timing margins were tightly timed preferentially being assigned non-inverting buffers to ensure the maximum possible control of propagation delay. Secondary criteria were based on capacitive load demands at every tree node; capacitance load capacitance more than the optimum drive range of a single stage of buffer, use a series of inverters in turn to achieve intermediate drive levels.

Phase requirements of the clock tree topology were used as tertiary criteria, with some balanced clock distribution topologies having to invert the phases of certain nodes to balance the path lengths and skew. The use of inverter pairs (two inverters in series) offered a good design trend to obtain the intended timing characteristics and recovery of phase relationship (Vishnu, P. V. *et al.* 2019). In the real-world clock tree design, the tradeoff between the choice of buffers and inverters needs to be evaluated with careful attention to delay properties across process corners and environmental conditions since the introduction of inverters causes a delay to increase that does not necessarily scale in the same way with process variation as paths of entirely buffer-based design (Niranjana, M. I.*et al.* 2023).

4.3 Impact Analysis: Power Consumption, Area Utilization, and Clock Skew

There are three important design measures directly influenced by the choice of clock cells; dynamic power consumption, silicon area usage and clock skew properties. The most important trade-off mechanism between the two metrics is represented by cell sizing decisions. Bigger clock cells have lower delay and better drive but use a similar amount of power and silicon area. Smaller clock cells on the other hand minimize power and area, but might need higher buffering levels to meet timing goals, which can add to the overall area and power consumption due to the additional cell adds (Han, K*et al.* 2018).

The clock cell selection algorithm used in the present work tried to reduce the overall power consumption with timing constraints, and repeated cell size optimization with timing analysis feedback. Cell selection based on power made the clock tree power consumption very low compared to the traditional methods which are more conservative and are likely to over-size. The area of clock cell selection was of special concern in this design because area was fixed at 514,863.594 sq. mm, and any area used in clock distribution network could not be used by logic cells. The effect of cell sizing on the skew was an expression of the basic fact that smaller cell sizes raise propagation delay variance within the clock tree, which might have to be propagated globally in a poorly-designed tree topology.

4.4 Implementation in Synopsys ICC2 Tool Flow

Selection strategy clock cell in the Cadence ICC2 (Integrated Circuit Compiler II) physical design tool was achieved by specifying cell constraints and optimization directives in the CTS flow. The cells in the library were pre-characterized on a timing, power, and area basis and the selection of cells was represented in the CTS configuration. The ICC2 tool was set to repeatedly test clock tree topologies with alternatives of the alternatives having different cell choices and optimizing towards the given objective (minimum power subject to timing closure in this work). The CTS optimization engine received timing feedback based on the results of a

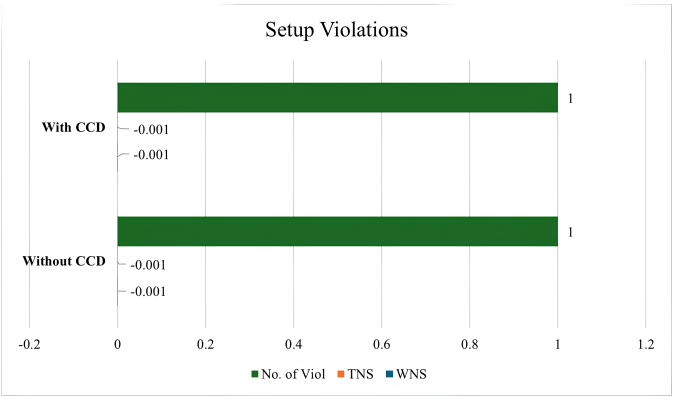

Fig. 2. Setup violations after CTS.

Table 1. *Clock timing summary report with using CCD after CTS*

Clock Summary	**Value**	**Corner**
Max. setup launch latency	0.26	ssg0p81vm40c_cw
Min. setup capture latency	0.10	ffg0p99vm40c_cb
Min. hold launch latency	0.10	ffg0p99vm40c_cb
Max. hold capture latency	0.26	ssg0p81vm40c_cw
Max. active transition	0.22	ssg0p81v125c_rcw
Min. active transition	0.02	ffg0p99vm40c_cb
Max. setup skew	0.13	ssg0p81vm40c_cw
Max. hold skew	0.13	ssg0p81vm40c_cw

static timing analysis, which was conveyed as clock timing reports, and used to make decisions regarding cell sizing and buffer insertion. The refinement process was typically performed through several CTS executions with gradual changes in the parameters since single-pass optimization of CTS could hardly fulfill all the objectives at the same time without iteration (Niranjana, M. I.*et al.* 2023).

5 CTS Timing Analysis

5.1 Results of Clock Latency Optimization

Clock latency, the prop delay between the primary clock source and each clock sink in the design is a key timing parameter that has a direct relationship with the clock frequency and timing closure that is possible (Ewetz, R. 2023). The un-optimized CTS initial configuration showed maximum setup launch latency of 0.14ns that is indicated in Table 2 and is the propagation delay over the longest path between the clock source and the clock pin of the flip-flop that is launching the critical timing path. Such a baseline latency served as a benchmark of measuring the effect of CCD-based optimization. Using CCD optimization, the maximum setup launch latency was 0.26 ns (Table 1) which is a 0.12 ns higher than with the standard CTS.

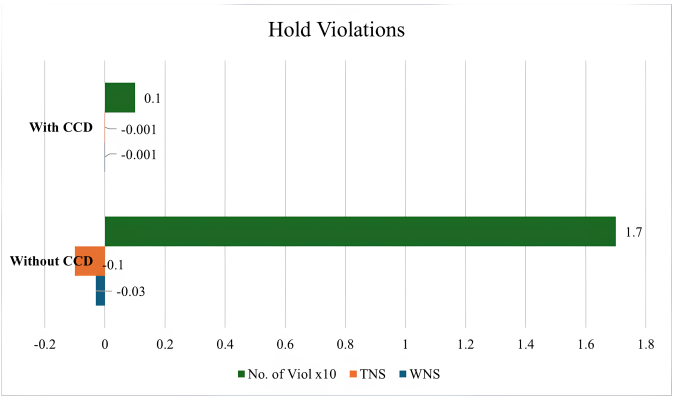

Fig. 3. Hold violations after CTS.

Such a latency increase is an intentional trade-off of the CCD optimization engine in favor of other optimization advantages, namely, to allow more efficient useful skew optimization and less buffering. This rise in latency is, counterintuitive as it appears, actually beneficial in power-constrained designs, in which overall minimization of power is more important than minimization of latency (Lu, Y. C. *et al.* 2023). Figure 2 and Figure 3 show the setup and hold violation trends over the design, and these figures show that, although the latency increased, the timing closure overall was better as a result of simultaneous optimization of both data path delays and clock skew.

5.2 Analysis of Clock Skew Reduction

Clock skew is the variation in the clock propagation to various sink nodes, which is a critical timing closure constraint. Too much skew leads to loss of timing margin and may result in designs that do not close timing constraints despite huge delays on the combinational paths (Kourtev I. S.*et al.* 2009). On the other hand, positive skew, when used appropriately, can also widen timing margins because it allows shorter clock periods in some of the critical paths. The traditional CTS (no CCD) had a maximum setup skew of 0.02 ns (Table 2), which is an indicator of very balanced clock distribution with a very small skew.

Nonetheless, the violent balancing necessary to reach these low skew levels meant that more buffer stages had to be inserted and that the buffer stages had to be carefully laid out to balance the path lengths. The CCD-based CTS method had a maximum setup skew of 0.13 ns (Table 1) and this is the introduced skew that was well controlled to offer desirable timing properties where required. The growth in skew between 0.02 ns and 0.13 ns is a deliberate optimization trade-off in which the total number of buffers in clock tree was altered to reduce power usage and accept skew, and it was realized that useful skew could be used to offset data path timing margins (Shan, W. *et al.* 2017). Such skew increase was confirmed by the timing analysis that was done in detail to ensure that the total setup and hold timing closure were met at all corners.

Table 2. *Clock timing summary report without using CCD after CTS*

Clock Summary	**Value**	**Corner**
Max. setup launch latency	0.14	ssg0p81v125c_rcw
Min. setup capture latency	0.10	ffg0p99vm40c_cb
Min. hold launch latency	0.10	ffg0p99vm40c_cb
Max. hold capture latency	0.14	ssg0p81v125c_rcw
Max. active transition	0.09	ssg0p81v125c_rcw
Min. active transition	0.01	ffg0p99vm40c_cb
Max. setup skew	0.02	ssg0p81vm40c_cw
Max. hold skew	0.02	ssg0p81v125c_rcw

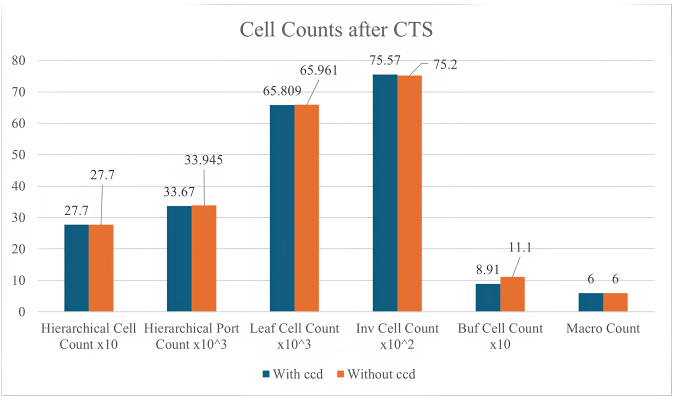

Fig. 4. Cells count after CTS.

5.3 Setup and Hold Timing After CTS

Setting up timing analysis determines the timing margin of data signals to travel across combinational logic and stabilize at the input of the data signal into the receiving flip-flops before the clock edge. In setup timing closure, the arrival time of the data signal, in addition to a clock-to-output delay at the launching flip-flop, and all the propagation delays of the combinational logic, and a set-up time requirement must all be met before the clock edge reaches the receiving flip-flop. The implementation of CCD optimization altered the clock latency and useful skew properties, which have a direct impact on the available set-up timing margin. Violation of setup timing in each design phase was analyzed and it was found that CCD optimization had caused some initial setup violations in the data path (Figure 2 indicates timing violations in CTS stage), which had been corrected by systematic data path optimization and successive refinement of CTS.

Hold timing analysis is used to measure the shortest time in which data signals can be steady following the clock edge. Hold violations are caused by the data signals varying too fast following the clock edge and may be captured by latch elements in the flip-flop before it reaches a steady state. The timing of hold

Table 3. *Clock QoR timing after cts*

Clock QoR	**With CCD**	**Without CCD**
Sinks	11887	11887
Levels	8	8
Clock Repeater Count	171	326
Max Latency	0.26	0.14
Global Skew	0.15	0.02
Wire Length	38935.72	42617.52

is a very sensitive area to clock skew because less clock skew on the flip-flop on the receiving end can bring the hold timing closer to the margins (Kourtev I. S.*et al.* 2009). Figure 3 illustrates the trends of the hold violations and there we can see that optimization of CCD did not lead to significant shifts in the hold timing properties in spite of the higher clock skew.

5.4 Global Timing Validation

Timing analysis was done on a comprehensive basis and at every stage of design (placement, CTS, routing) to ensure that global timing closure was maintained throughout the design flow. Timing analysis of the primary clock was enabled on the primary clock to check all paths in the primary clock through the combinational logic to the flip-flop inputs and ensure that they would finish within the primary clock cycle time of 2.2 ns with input and setup delays applied. Hold timing analysis was used to analyze all the flip-flop-to-flip-flop paths to ensure there were no hold violations against the constraints that were applied. The block took the longest combinational paths in the design, which were about 0.7 to 0.9 ns, and provided sufficient margin compared to the 2.2 ns cycle time. Multi-corner analysis (slow-slow at low voltage and high temperature, fast-fast at high voltage and low temperature and nominal corner) showed timing closure was preserved in all corners, which confirmed the resilience of the design (Ewetz, R. 2023).

6 Trade-Off Analysis and Optimization Balance

6.1 Latency vs Power Trade-Off

A basic trade-off between power consumption and clock latency was one of the main influences in CCD-based optimization choices in the design. The standard clock tree synthesis is trying to minimize the latency as it is implemented using aggressive buffering, where a series of buffer stages are added to the clock tree to minimize the total wire delay and to provide rapid propagation of the clock signal. Nevertheless, every insertion of a buffer contributes to dynamic power consumption (because of switching activity), and static leakage power (Lu, Y. C. *et al.* 2023).

The CCD optimization strategy explicitly traded-off the goals of latency optimization to the goal of reducing the total power consumption, tolerating higher latency in case the higher latency could be compensated by large power savings

Table 4. *Utilization after cts.*

After CTS	With CCD	Without CCD
Utilization Ratio	0.2274	0.2280
Total Area	514863.594	514863.594
Total Capacity Area	261566.2980	261621.7380
Total Area of Cells	59488.6320	59654.4480

due to reduced buffering. The quantitative effect of this trade-off was also clear in the results: in the absence of CCD optimization, the maximum setup launch latency was 0.14 ns with the use of 326 clock repeaters (Table 3); in the presence of CCD optimization, maximum setup launch latency was 0.26 ns with the use of only 171 clock repeaters (Table 3), a 47.5 percent decrease in the number of repeats. In the assumption that each repeater is drawing around 2 to 4 micro amperes in dynamic switching activity and extra leakage, the power savings of 326 to 171 repeater count mean a lot of power saved in the clock distribution network. This 0.12 ns latency increase took about 5.5 percent of the 2.2 ns clock period (timing budget) but the power savings associated with it was much larger than the loss of timing margin in the power-constrained design environment.

6.2 Skew vs Area Trade-Off

Another trade-off related to clock tree design was clock skew and silicon area usage. Reduction of clock skew by employing balanced tree construction means that the buffers should be placed with care and that several iterations should be performed to balance the path lengths between the essential clock source and all sinks. This balancing can frequently necessitate the addition of buffer stages in shorter paths in order to compensate delays in longer paths, and in effect pad shorter paths to meet skew targets. This type of padding wastes silicon area, which is inefficient in terms of area. The CCD optimization strategy tolerated greater clock skew (0.13 ns compared to 0.02 ns) at the cost of having a smaller area cost of the clock tree itself. But it was not only area trade-off that occurred in clock tree footprint, the lowering of clock latency by traditional CTS implied that further placement-level optimization was necessary to accomplish timing closure in the data paths, which could raise placement area overhead (Semeraro, G. *et al.* 2002).

The real values of area utilization (Table 4 and Table 3 indicate that total area of cells used post-CTS was 59486.32 sq. mm with CCD and 59654.448 without CCD) indicate that the area consumption with CCD optimization was slightly less (0.28 percent lower) than with data path buffering additions, implying that power savings due to decreased clock buffering counterbalance the area consumption due to data path buffering additions. The skew vs area trade off in this design had shown that skew acceptance at the price of enhanced skew might in fact decrease the overall area footprint when applied in combination with effective skew optimization at the data path level.

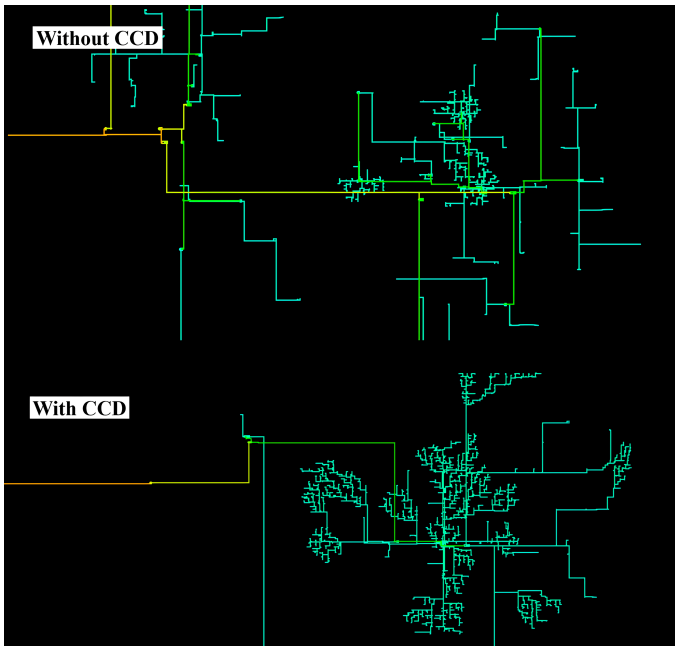

Fig. 5. Clock path from level 1 to level 3.

6.3 CTS Impact on Data Path Timing

Clock tree synthesis essentially influences the performance of data paths that can be achieved in terms of timing . The timing margin that might be utilized by data paths to pass through combinational logic is also directly established by the clock latency and skew properties established by CTS . Unnecessary clock latency decreases timing window available, which can cause multi-cycle path constraints or may need to cut functional frequency in case not covered in the design frequency target (Semeraro, G. *et al.* 2002).

Timing margin loss at flip-flops receiving late clock edges is caused by excessive clock skew, especially positive clock skew whereby some clock sinks are fed with delayed clock edges. On the other hand, positive negative skew at critical flip-flops may be used to extend timing margins by enabling longer combinational path delays. The CCD optimization framework was explicit in the timing requirements of the data path during the construction of the clock tree and allowed the optimization engine to introduce beneficial skew at critical data path nodes on demand. This clock path and data path co-optimization is fundamentally different than the traditional sequential methodologies, where clock tree construction is done without any knowledge of how important data paths are in reality (Li, Z. *et al.* 2025).

This effect on the data path timing was captured in the violation statistics: with no CCD optimization, the number of set up violations in the post-CTS was

Table 5. *Utilization after placement*

Utilization	**With CCD**	**Without CCD**
Utilization Ratio	0.2266	0.2265
Total Area	514863.594	514863.594
Total Capacity Area	261566.2980	261621.7380
Total Area of Cells	59260.824	59258.43

low due to the balanced clock tree providing conservative timing margins; with CCD optimization, there were initial set up violations (Figure 2) but which were addressed systematically by concurrent data path optimization. The resulting design has timing closure with good setup margins at all the endpoints, which validates that CCD approach was effective in optimizing the coupled system of clock and data paths.

6.4 Observed Optimization Balance

The results obtained showed that CCD-CTS methodology was effective in balancing optimization of various competing goals in the power-constrained design scenario. The clock network design that had CCD optimization had good balance in the metrics of latency, skew, power, and area than the traditional methods. Such a quantitative advantage as the decrease in the number of clock repeaters (47.5 percent between 326 and 171) is the most dramatic, and it is directly related to significant power and area savings. The 0.12 ns additional latency also cost about 5.5 percent of the timing budget which was an acceptable trade-off considering the power savings.

The skew increment of 0.13 ns was also carefully controlled by co-optimizing the data path to ensure that skew relationships could be used beneficially to transform what could otherwise have been a timing disadvantage, into a timing benefit. The ratio of overall utilization was slightly higher than 0.2280 to 0.2274 in CTS (Table 4), which means that there was very little area overhead due to CCD-based cell selection and placement corrections. Such an outcome of balanced optimization could not have been attained using the traditional methods of sequential optimization because neither clock-oriented optimization nor data-path-oriented optimization could have been used to achieve the desired power, area, latency, and skew goals at the same time (Wu, J. *et al.* 2025).

7 Summary of Findings

7.1 CTS Quality Metrics Summary

The overall clock tree synthesis results are summarized in given tables, that offer quantitative measures that allow a detailed comparison between the traditional CTS and CCD-based CTS methods. Table 5 shows the utilization metrics right after the placement stage, which defines a baseline utilization of areas before clock

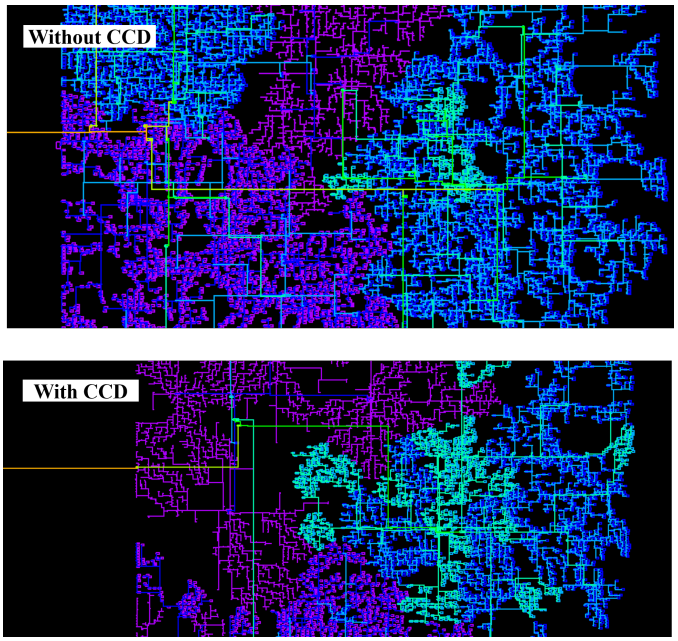

Fig. 6. Clock path from level 1 to level 6.

tree synthesis. Utilization ratio is constant at around 0.226 with and without CCD which shows that the placement stage decisions were the same in both the optimization paths. Table 4 shows the utilization measures after CTS completion. In CTS stage, the design was enhanced with clock cells, which raised the total cell area to some 59,260 sq. mm to some 59,488 or 59,654 sq. mm with or without CCD optimization.

Utilization ratio also rose slightly to 0.2274 with CCD and 0.2280 without CCD, meaning that there was no significant difference in overall area consumption between the two methods, but the cell distribution between clock network and data path optimization was different. Table 2 shows the clock timing statistics without CCD optimization and the highest setup launch latency is 0.14 ns, minimum setup capture latency is 0.10 ns and highest setup skew is 0.02 ns. Table 1 shows the same metrics with CCD optimization used with maximum setup launch latency of 0.26 ns, minimum setup capture latency of 0.10 ns (no change) and maximum setup skew of 0.13 ns. This comparison shows that CCD optimization introduced intentionally larger latency and skew in order to achieve lower repeater requirements in general (Pouiklis, G. *et al.* 2016).

Table 3 shows clock quality of results metrics right after CTS, which sets the detailed comparison between the two optimization strategies. The legacy balanced clock tree was configured with 326 clock repeaters spread over eight levels of hierarchy to access all 11,887 clock sinks and with the maximum latency

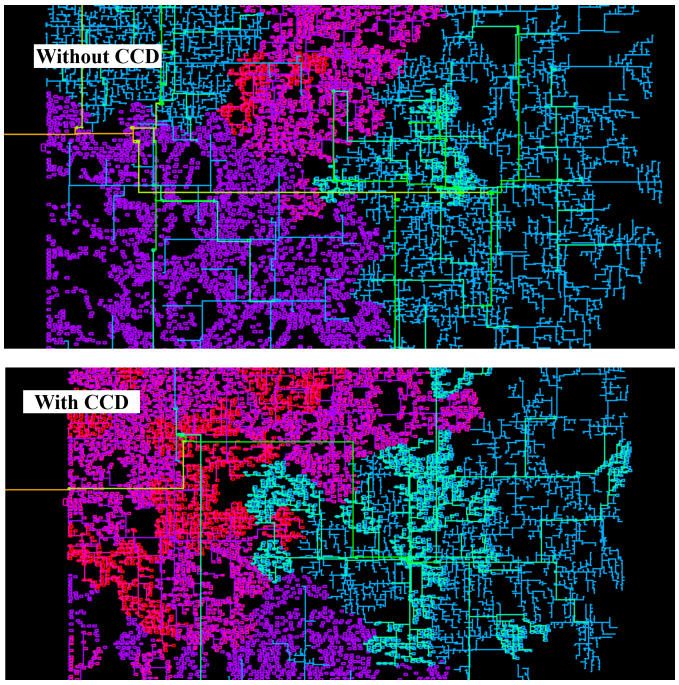

Fig. 7. Clock path from level 1 to level 9.

of 0.14 ns and a global skew of 0.02 ns, and the total clock wire length of 42, 617.52 micrometers, without CCD optimization. Using CCD optimization, the clock tree used only 171 repeaters (47.5 percent reduction), which were spread out across the same eight hierarchical levels, had a maximum latency of 0.26 ns and global skew of 0.15 ns, and a total clock wire length of 38,935.72 micrometers (8.6 percent reduction). The clock network power saving is directly proportional to the dramatic decrease in the number of repeaters. Table 6 shows the design rule violation after routing, a key phase at which parasitic extraction and last validation takes place (Chakraborty, A. *et al.* 2010).

In the absence of CCD optimization, four of the maximum transition time violations and five of the maximum capacitive load violations occurred during routing (Table 6), and it can be deduced that not all clock nets were routed to meet design rule requirements on signal integrity. Zero design rule violations were seen after routing with CCD optimization (Table 6) showing better compliance with design rules by use of more thoughtful clock network topology (Chakraborty, A. *et al.* 2010). Table 7 shows final utilization measures after completion of routing. The utilization ratios were 0.2283 with CCD and 0.2287 without CCD, which implies final designs with basically the same area utilization. Final clock quality of results metrics after routing and final parasitic extraction are shown in table 8.

Table 6. *DRV violations after routing*

DRV violations	**With CCD**	**without CCD**
Total no. of i/p nets	67811	67966
Max tran violations	0	4
Max Cap violation	0	5

Table 7. *Utilization after routing*

After Routing	**With CCD**	**Without CCD**
Utilization Ratio	0.2283	0.2287
Total Area	514863.594	514863.594
Total Capacity Area	261566.2980	261621.7380
Total Area of Cells	59715.684	59830.974

The measurements after CTS confirm the trends, CCD optimization is maximum latency of 0.28 ns, global skew of 0.16 ns, with only 171 repeaters and 38,655.30 micrometers of clock wire length, compared to conventional CTS with maximum latency of 0.15 ns and global skew of 0.05 ns with 326 repeaters and 42,578.26 micrometers of clock wire length.

7.2 Practical Implementation Challenges

The use of the CCD based CTS methodology in the design has had its share of practical problems that had to be overcome during the design. The first significant issue was convergence of CCD optimization engine to solutions that are achievable and satisfy all the constraints. The CCD framework can be used to optimize multiple conflicting objectives (latency, skew, power, area) simultaneously and naive optimization algorithms can lead to oscillation or divergence during optimization when a single objective is changed, resulting in conflicts in other objectives (Lee, K.*et al.* 2006). They have solved this challenge by ensuring that prioritization of optimization targets in the CCD tool configuration is put into great consideration to ensure that primary objectives (timing closure and power minimization) and secondary objectives (area and skew) are set so that realistic values are achieved as per the set primary objectives. The second issue was the one of validation of

Table 8. *Clock QoR after routing*

Clock QoR	**With CCD**	**Without CCD**
Sinks	11887	11887
Levels	8	8
Clock Repeater Count	171	326
Max Latency	0.28	0.15
Global Skew	0.16	0.05
Wire Length	38655.30	42578.26

the proposed beneficial skew benefits against the actual timing analysis .

The suggested CCD optimization engine introduced additional skew to the basis of mathematical optimization but this was required to be justified by observing that such skew actually provided timing advantages of the skew as determined by the skew. The initial results were that the proposed skew on some sections of the design did not achieve the intended timing gains, and as such, there was the necessity to repeat and optimize skew distribution (Pouiklis, G. *et al.* 2016). This challenge was one of the reasons behind the creation of timing feedback loop of the static timing analysis to the CCD optimizer that enables the tool to actively adjust skew targets to actual timing validation rather than pure mathematical optimization. The third one was the combination of CCD-optimized clock tree and the downstream routing step. The CCD optimization tool was able to adjust the clock network topology and the clock cell location, but they had to be aligned with routing stage constraints like congestion limits and design rule viability. The initial CCD proposals suggested clock topology that created routing congestion of critical data paths, and had to impose restrictions on the CCD formulation to guarantee that the topology was not congestion causing.

A fourth problem was the problem of control of a number of design corners (slow-slow, fast-fast, nominal) within the CCD optimization framework. Preferably, optimization of CCD should be done to ensure that there is a beneficial skew in all corners, although the skew benefit in a corner and hurt in a corner may occur depending on the variation of the process. This was an issue that had to be studied more closely concerning corner-specific timing quantities and establish powerful skew aims that provided equal timing improvements in all corners (Sitik, C. *et al.* 2016).

7.3 Benefits of CCD-Based CTS in Power-Constrained Designs

The CCD-based CTS technique had demonstrated several significant benefits that may be specifically utilized on the power-constrained ASIC design flows. The greatest benefit was that the repeats in clock distribution were reduced by 47.5 percent (326 to 171 repeats) (Table 3), which was a very large saving in clock distribution network power. At a power consumption of 2 to 4 micro amperes per clock repeater as an average, the elimination of 155 clock repeaters would lead to a current loss of approximately 310 to 620 micro amperes of dynamic current in the clock network. This is among the most significant possibilities of power optimization in the designs whose clock frequencies are in the gigahertz range and numerous clock distribution cells. Besides saving absolute power, the CCD method also enabled the choice of cells according to power in the clock network, and prefer high-threshold voltage cells in non-critical parts of the clock tree (Dai, Y. Y. *et al.* 2018). The power conscious design also reduced the leakage power consumption in the clock network that is also beneficial

even at the idle operating conditions when dynamic power consumption is minimal.

The second significant benefit was improvements in design rule compliance post routing whereby CCD optimization achieved zero design rule violation and five capacitive load violation versus four transition time and five capacitive load violation in conventional CTS (Table 6). Signal integrity errors that are caused by such violations of design rules have to be either iterated or tolerated in design (Lee, K.*et al.* 2006). The adherence to design rules was more appropriate in the CCD methodology due to the low risk of verification, and the possibility to emend the design quicker than to the CTS stage. The third benefit involved the reduction in total clock wire length (8.6 percent reduction of 42, 617.52 to 38, 935.72 micrometers) (Table 3), which contributed to a reduction in power consumption in clock routing and a reduction in the congestion in the chip routing. The lower buffers, lower interconnect and better routing design rules were converted to more powerful and efficient clock distribution network.

The fourth benefit was the establishment of a foundation on which optimization could be carried out in future. The CCD method and practical skew analysis enabled by the simultaneous optimization provides a platform on which more sophisticated optimization techniques, such as adaptive clock stretching (Shan, W. *et al.* 2017), dynamic voltage and frequency scaling, and clock gating integration can be found. Optimized designs based on CCD methodology are better aware of skew relationships and timing margin distribution and the advanced techniques can be realized with less risk than the conventional balanced clock trees.

8 Interpretation and Implications

8.1 Applicability to Larger and Faster Designs

The CCD-CTS scheme, illustrated in the block, is a power-constrained design with a complexity of much but not immensely large (approximately 66,000 flattened cells) complexity. The uncertainty about the ability of it to be extended to much larger designs (million of cells) and faster designs (millions of hertz) is also a major consideration to whether the CCD route should be incorporated into the mainstream of design practice. The scalability of CCD optimization engine may be a serious problem in larger designs; optimization structures founded on heuristics of global optimization (simulated annealing or genetic algorithms) may become computationally infeasible in very large designs (Li, Z. *et al.* 2025). The CCD optimization scheme applied in this work relies upon an iterative refinement scheme with localized optimization updates, implying a sensible scale to bigger designs, nevertheless, on designs with millions of cells the overall capability of CCD remains unexplored in the literature .

The less latency budget per cycle is more important in higher frequency designs of multi-gigahertz speed. The timing closure of the block CCD-CTS

implementation was approximately 5.5 percent of the clock period; and can be scaled to a gigahertz implementation with an equivalent reduction in period. However, the absolute values of the latency of the number of ns would not proportion as clock tree latency is inherently limited by the RC propagation through interconnects and circuit delays are not directly proportional to the higher frequency. The issue at gigahertz design is fewer clock signals are allocated with even more stringent latency and skew specifications, and the CCD optimization at latency reduction is even more welcome (Mathur, U.*et al.* 2022).

8.2 Scalability to Advanced Technology Nodes

The design was prepared in the 28 nm CMOS technology which is an established node of technology with well established design rules and large cell libraries. The problem of the applicability of CCD-CTS to the more developed technology nodes (20 nm, 16 nm, 7 nm, 5 nm) is also the most relevant as advanced nodes are regarded as the border of the ASIC design. The fundamental principles of CCD optimization Co-optimization of clock and data paths, leverage of favorable skew and power-sensitive cell selection are technology-neutral and should apply to advanced nodes. However, technology-related problems are a few, which happen in advanced nodes. First, process variation in advanced nodes is becoming increasingly harder to predict and control, and clock variation is harder to predict and manage between manufacturing variation. CCD optimization techniques must be experimented in larger ranges of variation of the process, and perhaps additional skew distributions than experienced in mature nodes (Kuzmin, P. A. 2024).

Second, the concentration of power and thermal is more significant in more advanced nodes and may require thermal optimization and temperature-dependent delay model to be included in the CCD framework. Third, the improved nodes can add more metal layers as well as routing resources, however, routing congestion is also more in the signal and power networks **??**. The CCD method approach needs to be adjusted to monitor the more congestion limits of developed nodes. However, the fundamental value proposition of the CCD-based optimization, or the capacity to optimize a group of objectives in consideration of inter dependencies concomitantly, needs to endure and will be more pronounced with the advancement of the process technology (Mathur, U.*et al.* 2022).

8.3 Design Guidelines for Power-Aware CTS

Based on the experiences of this work, a set of design rules can be formulated that is applicable to the practitioners who are using power-aware clock tree synthesis with power-constrained ASIC design. First of all, it is recommended to establish distinct priorities of design objectives at the start of physical design. The primary objective of the design was made to be power minimization, timing closure was a binding constraint and latency/skew optimization was a secondary

objective. This prioritization was clear enough to make the tools configured and the optimization frameworks to be selected in a manner that directly tackles the intended optimization goal. Second, give complete timing feedback of the output of the timing analysis of the static analysis to the clock tree optimization engine. The benefits of CCD optimization can only be attained when timing analysis is capable of assuring that the suggested clock skew configurations are capable of delivering the timing gains as they were predicted (Prasad, D. *et al.* 2016).

Third, reason about iterative CTS optimization, rather than thinking of single-pass optimization convergence. Three to five CTS runs are typically required in case optimization of CCD is required with the parameters and constraints to get the optimum balance between conflicting objectives. Fourth, test results everywhere in the process and everywhere in the conditions of the environment and then consider CTS to be complete. The positive skew benefits must be very high in variation or the design may not work in silicon. Fifth, make sure that there is a high level of isolation between clock network design and data path optimization during the former and then fine tuning during the latter round of CCD optimization. The traditional sequential convergence methods may be beneficial during initial design stages and the CCD optimization is applied in later design steps, and optimizes multi-objective tradeoffs.

9 Conclusion

Future research should focus on integrating adaptive clock distribution with CCD optimization to address runtime variations, alongside designing dynamic clock gating and thermal-aware modeling to further enhance power reliability. Leveraging machine learning for faster design convergence and extending the framework to complex, multi-domain hierarchical ASIC designs remain critical next steps. Ultimately, this study on 28nm technology demonstrates that CCD-based optimization is a highly effective industrial methodology, achieving a 47.5% reduction in clock repeaters and an 8.6% decrease in total wirelength while eliminating design rule violations. By concurrently optimizing clock and data paths, the approach facilitates aggressive power savings without compromising timing closure. Compared to conventional balanced clock trees, the strategic use of positive skew through CCD-CTS provides superior performance, establishing it as a practical and necessary solution for modern, power-constrained VLSI design.

References

Datli, A., Eksi, U., & Isik, G. (2013). A clock tree synthesis flow tailored for low power.

Han, K., Kahng, A. B., & Li, J. (2018). Optimal generalized H-tree topology and buffering for high-performance and low-power clock distribution. Ieee transactions on computer-aided design of integrated circuits and systems, 39(2), 478-491.

Vishnu, P. V., Priyarenjini, A. R., & Kotha, N. (2019, May). Clock tree synthesis techniques for optimal power and timing convergence in soc partitions. In 2019 4th International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT) (pp. 276-280). IEEE.

Shan, W., Dai, W., Wan, L., Lu, M., Shi, L., Seok, M., & Yang, J. (2019). A bi-directional, zero-latency adaptive clocking circuit in a 28-nm wide AVFS system. IEEE Journal of Solid-State Circuits, 55(3), 826-836.

Anirudh, S., & Ramesh, T. K. (2022, December). An Enhanced Clock Tree Synthesis Methodology for Optimizing Power in Physical Design. In 2022 IEEE 3rd International Conference on VLSI Systems, Architecture, Technology and Applications (VLSI SATA) (pp. 1-9). IEEE.

Lu, Y. C., Chan, W. T., Guo, D., Kundu, S., Khandelwal, V., & Lim, S. K. (2023, July). RL-CCD: concurrent clock and data optimization using attention-based self-supervised reinforcement learning. In 2023 60th ACM/IEEE Design Automation Conference (DAC) (pp. 1-6). IEEE.

Li, W., Huang, Z., Yu, B., Zhu, W., & Li, X. (2024, June). Toward controllable hierarchical clock tree synthesis with skew-latency-load tree. In Proceedings of the 61st ACM/IEEE Design Automation Conference (pp. 1-6).

Wu, J., Ni, C., Wang, H., & Chen, J. (2025). Graph neural networks for efficient clock tree synthesis optimization in complex SoC designs. Applied and Computational Engineering, 150, 101-111.

Mo, X., Wu, J., Wary, N., & Carusone, T. C. (2021). Design methodologies for low-jitter CMOS clock distribution. IEEE Open Journal of the Solid-State Circuits Society, 1, 94-103.

Pasca, A., & Ciugudean, M. (2016). Highly Efficient, Zero-Skew, Integrated Clock Distribution Networks Using Salphasic Principles. Advances in Electrical and Computer Engineering, 16(1), 69-79.

Niranjana, M. I., Dhanasekar, J., Blesson, N., Immansingh, G. D., Boopathi, M., & Raman, S. M. (2023, November). A framework for block-level physical design using icc2 in 14nm technology. In 2023 7th International Conference on Electronics, Communication and Aerospace Technology (ICECA) (pp. 413-417). IEEE.

Ewetz, R. (2023, June). A clock tree optimization framework with predictable timing quality. In Proceedings of the 54th Annual Design Automation Conference 2017 (pp. 1-6).

Kourtev, I. S., Taskin, B., & Friedman, E. G. (Eds.). (2009). Timing optimization through clock skew scheduling. Boston, MA: Springer US.

Shan, W., Wan, L., Liu, X., Shang, X., Dai, W., Shao, S., ... & Shi, L. (2017). A low overhead, within-a-cycle adaptive clock stretching circuit with wide operating range in 40-nm CMOS. IEEE Transactions on Circuits and Systems II: Express Briefs, 65(11), 1718-1722.

Semeraro, G., Magklis, G., Balasubramonian, R., Albonesi, D. H., Dwarkadas, S., & Scott, M. L. (2002, February). Energy-efficient processor design using multiple clock domains with dynamic voltage and frequency scaling. In Proceedings Eighth International Symposium on High Performance Computer Architecture (pp. 29-40). IEEE.

Pouiklis, G., & Sirakoulis, G. C. (2016). Clock gating methodologies and tools: a survey. International journal of Circuit theory and Applications, 44(4), 798-816.

Li, Z., Chen, B., Wang, W., Lv, H., Lv, Q., Chen, J., ... & Zhang, C. (2025). A Method for Synthesizing Ultra-Large-Scale Clock Trees. Algorithms, 18(5), 249.

Chakraborty, A., Duraisami, K., Sithambaram, P., Macii, A., Macii, E., & Poncino, M. (2010). Thermal-aware clock tree design to increase timing reliability of embedded socs. IEEE Transactions on Circuits and Systems I: Regular Papers, 57(10), 2741-2752.

Lee, K., Lee, S. J., & Yoo, H. J. (2006). Low-power network-on-chip for high-performance SoC design. IEEE transactions on very large scale integration (vlsi) systems, 14(2), 148-160.

Sitik, C., Liu, W., Taskin, B., & Salman, E. (2016). Design methodology for voltage-scaled clock distribution networks. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 24(10), 3080-3093.

Dai, Y. Y., & Brayton, R. K. (2018). Verification and synthesis of clock-gated circuits. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 38(2), 366-379.

Prasad, D., Pan, C., & Naeemi, A. (2016, March). Impact of interconnect variability on circuit performance in advanced technology nodes. In 2016 17th International Symposium on Quality Electronic Design (ISQED) (pp. 398-404). IEEE.

Mathur, U., Pavlogiannis, A., Tunç, H. C., & Viswanathan, M. (2022, February). A tree clock data structure for causal orderings in concurrent executions. In Proceedings of the 27th ACM International Conference on Architectural Support for Programming Languages and Operating Systems (pp. 710-725).

Kuzmin, P. A. (2024, June). Comparative Analysis of the Characteristics of Clock Network Structures Buffer Tree, H-tree, Clock Mesh for Technological Nodes 28nm and 90nm. In 2024 IEEE 25th International Conference of Young Professionals in Electron Devices and Materials (EDM) (pp. 270-279). IEEE.

Wu, J., Ni, C., Wang, H., & Chen, J. (2025). Graph neural networks for efficient clock tree synthesis optimization in complex SoC designs. Applied and Computational Engineering, 150, 101-111.

Kahng, A. B. (2018, March). Machine learning applications in physical design: Recent results and directions. In Proceedings of the 2018 international symposium on physical design (pp. 68-73).