

Physical Design Implementation to Enhance Performance of Ultra-Deep Sub-micron (UDSM) Hard Macro-based Design

PRASAD SHENOY^{1,3}, N. SHYLASHREE^{1,3,a}, NANDINI K. S.^{2,3}, PRAKASH TUNGA P.^{2,3}

¹Department of Electronics and Communication Engineering,
R. V. College of Engineering,
Bengaluru,
INDIA

²Department of Electronics and Communication Engineering,
RNS Institute of Technology,
Bengaluru,
INDIA

³Affiliated to Visvesvaraya Technological University,
Belagavi-590018, Karnataka,
INDIA

^aORCID: <https://orcid.org/0000-0003-4185-6190>

Abstract: - Lower technology nodes or Ultra Deep Sub-micron (UDSM) used in today's System on-chip (SoC) yield high performance and are faster when compared to the Deep Sub-micron (DSM) technology nodes. The challenges faced by a designer have increased multi-fold in SoC design with the technology node evolution. The designs that are trending these days are Hard Macro (HM) based designs. The entire block is divided in the sub-HMs or hierarchical blocks. The sub-HM layout shapes are decided during the partitioning of the top-level block. Each of the hierarchical blocks is implemented separately and the sub-HMs are integrated at the top level to reduce the huge run-time and to reduce the burden of improving the PPA. The processes involved in Physical Design (PD) are interlinked and the effect of the previous process can be seen in the subsequent stages. CTS is implemented using Flexible-H-tree (FHT) with Multiple tap point structure. Experimental results on an industrial design having more than a million instances show that the implementation of the proposed clock tree structure comprising of FHT with Multitap point CTS (DB2), in the design, shows huge improvement in terms of timing and clock metrics when compared to the conventional CTS (DB1). A reduction of 40.47% and 63.9% is seen in terms of hold WNS and hold TNS respectively from DB1 to DB2. The NFE is reduced by 27.33% from DB1 to DB2. DB2 has a clock latency of 1.073 ns which is 27% lesser than that of DB1. Global Skew is reduced by 15.48% and local skew by 20%. Innovus tool is used for the implementation of the design.

Key-Words: - Ultra Deep Sub-micron (UDSM), Physical Design, Multi-Tap point CTS, MSCTS, Flexible h-tree, Hard Macro design.

Received: April 11, 2024. Revised: August 13, 2024. Accepted: October 9, 2023. Published: November 14, 2024.

1 Introduction

The most challenging task for designers is timing closure of the design to meet tape-out targets in this competitive world. Especially when it is a multi-corner, multi-mode design (MCM), [1]. PD is a complicated process in Application Specific Integrated Circuit (ASIC) design and it is subdivided into many sub-steps. Different validation and verification procedures are performed in the layout design while PD is in progress. The major

steps involved in the PD flow are Partitioning, Floorplanning (FP), Placement, Clock-tree Synthesis, Routing, Timing closure, Sign-off Checks, etc. PD process starts with the partitioning of the design block into sub-HMs. Partition based floorplan methodology helps in achieving a better timing closure, [2]. Firstly, FP is done for the given design by importing the Verilog netlist provided by the front-end designer and other inputs such as Unified Power Format (UPF), Synopsys Design

Constraints (SDC), and Library Exchange Format (LEF) and Liberty Timing File (.lib). The core utilization is decided at the FP stage. After Floor-planning the design by placing the memories and pins at suitable locations in the given core area, placing standard cells in the core area will be done in the subsequent placement step. Congestion analysis is done to analyze the effectiveness of the placement stage. Areas having high standard cell density are spread in the design, [3]. After the placement stage, Scan Chain Reordering is performed to optimize the process of stitching which helps in the reduction of metal wire length and helps in the reduction of timing violations, [4]. After placement, a clock tree is constructed for the design during CTS where clock paths are realized to supply a clock signal to every sequential element present in the design.

CTS plays a very essential role in the converging of the design. With moving down to lower technology nodes, the designs are the main parameters to be dealt with while the building of clock structure or network On-Chip Variation (OCV) effects, complexity of design, and low power considerations, [5]. The clock signal is the highest power consumer in any given design (35 to 40 %), [6]. Different CTS architectures like H- Tree, X-Tree, Fishbone, and conical fishbone are analyzed and proposed in the literature and it is inferred that the suitability of the CTS schemes depends on the architecture used, [7]. The CTS structures aim at reducing the clock skew and insertion delay, [8]. As the operational voltage is reduced, the drivability of the clock buffers is degraded, and the issue of slew comes into the picture, [9]. For designs with multiple clocks, it is challenging to balance the clock trees. The challenges are due to the dependency of timing constraints on the clock period and delay corners. The delay corner depends on the Process, Voltage, and Temperature (PVT) which are variables. Furthermore, the longer the non-common paths in the clock structure, the more the OCV induced clock uncertainties, thereby, increasing the clock uncertainties on the launch and capture paths. CTS must determine branching points in the clock tree which are optimal, to reduce the clock uncertainties on non-common paths due to OCV, [10]. The smaller the non-common paths, the smaller the OCV induced clock skew. Further, techniques are proposed wherein optimal buffer insertion points are determined during clock tree building, [11]. Then during the routing stage, all the metal interconnects for data paths are realized in the upper metal layers, followed by Post Route Optimization for the routed design. Static Timing

Analysis (STA) checks are performed on the design and timing violations such as, setup and hold timings are fixed to achieve timing closure of the design.

The rest of the paper is organized as follows. Section 2 explains the fundamental theory required for the understanding of the further sections. Section 3 illustrates the methodology used in the PD flow. Section 4 presents a design and implementation strategies employed in the FP, Placement and CTS stages of PnR implementation. Section 5 discusses the results and conclusion.

2 Fundamental Theory

To understand the techniques along with their design and implementations that have been explained hereon, these fundamental terminologies need to be understood.

2.1 Clock Insertion Delay / Clock Latency

Each of the sequential circuits in a design is triggered by a clock signal which originates from a source. The clock source may be a Phase Locked Loop (PLL). Sequential circuit necessarily means Flip Flops (FFs). The clock signal from the clock source reaches a clock definition point which is the port of a sub-block/hierarchy as indicated in Figure 1. From the clock definition point the clock signal is distributed to each of the FFs within the sub-block/hierarchy. The clock pin of the FF is the clock sink. The total amount of time (delay) taken from the clock source to the clock definition point is termed the source insertion delay or source latency.

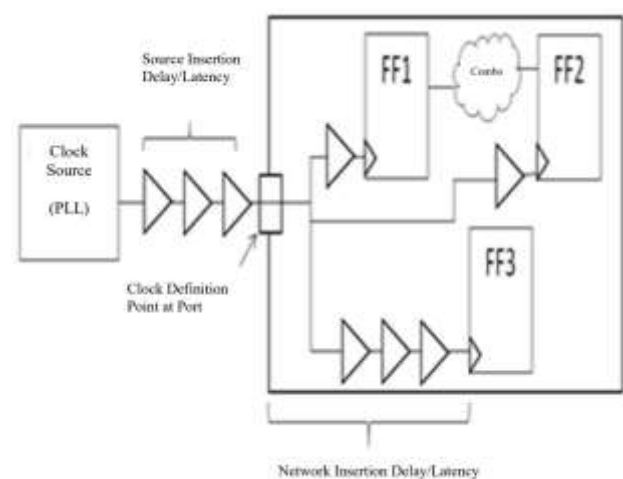


Fig. 1: Clock Latency

The total amount of time (delay) taken from the clock definition point to the clock pin for each of the FFs is termed as network latency. The sum total of

the source latency and network latency is defined as the clock latency or clock insertion delay.

2.2 Max, Min, and Avg Network Latency

Each clock sink has different network latencies. Max network latency (Max ID) is the maximum of all the network latencies for a clock domain. Min network latency (Min ID) will be the minimum of all the network latencies for a clock domain. Average network latency (Avg ID) is the average of all the network latencies for a clock domain. For example, as seen from Figure 1 FF1 is located nearest to the clock definition point. Thereby, the time taken for the clock signal to travel from the clock definition point to the clock pin of FF1 is termed as the min network latency/ Min ID.

2.3 Global Skew and Local Skew

Global Skew is the difference between Max ID and Min ID for a clock domain. Local skew is the difference Min ID and Max ID of a pair of registers which have a timing path between them in a clock domain. For instance, the skew between FF1 and FF2 is present in the Figure 1.

2.4 WNS, TNS, and NFE

For any timing path, Data Arrival Time (DAT) is the time taken for the data to travel through the data path. Data Required Time (DRT) is the time taken by the clock to traverse the clock path. For the setup relation not to be violated in a timing path, the data must be stable for a minimum time (setup time) before the rising edge of the clock (Setup relation: Data Arrival Time \leq Data Required Time).

For the hold relation not to be violated in a timing path, the data must be stable for a minimum amount of time (hold time), after the rising edge of the clock (Hold relation: Data Arrival Time \geq Data Required Time). Slack is the difference between the data required time and the data arrival time.

$$\text{Setup Slack} = \text{DRT} - \text{DAT} \quad (1)$$

$$\text{Hold Slack} = \text{DAT} - \text{DRT} \quad (2)$$

As seen from the above equations setup violations majorly depend on the data path delay and the hold violations majorly depend on the clock path delay. Therefore, setup violations are majorly fixed at the Placement stage of the PD flow (before CTS) as the clock will be ideal before CTS (skew = 0).

The actual clock tree is built only in the CTS stage and the clock is propagated. Actual clock

skew comes into the picture only after the CTS stage. As the hold violations majorly depend on the clock path delay, the hold violations are fixed only after the CTS stage of the PD flow.

If the slack is negative, it indicates that the timing path has a violation. If the slack is positive or 0, the path has no timing violation. Worst Negative Slack (WNS) is the critical path slack, which is the path worst negative slack. WNS can be negative, positive, or zero. Total Negative Slack (TNS) is the sum of all the negative slacks. TNS can be negative or 0. A number of Failing End Points (NFE/FEP) indicates the number of paths in the design which fail to meet the timing requirements which may be set-up/hold.

3 Methodology

The main steps involved in PD flow are as indicated in Figure 2. Each of the stages is implemented and optimization is carried out at each of the steps using EDA tools and their features. The FP process is carried out efficiently to reduce the efforts in the future stages. Further, the placement step is executed and timing and congestion analysis is performed to ensure the convergence of the design and quality of placement. Multi-tap point clock structure with flexible-H tree is found to be most suitable clock tree structure for the design and for improving the clock QoR. Further, the routing is carried out and further optimized based on the results obtained. Signoff checks are performed is carried to analyze the quality of routing. Thereafter, Timing checks are performed after each stage to estimate the setup/hold violations in the design. The Electronic Design and Automation (EDA) tool used for implementation is Innovus, [12], from Cadence.

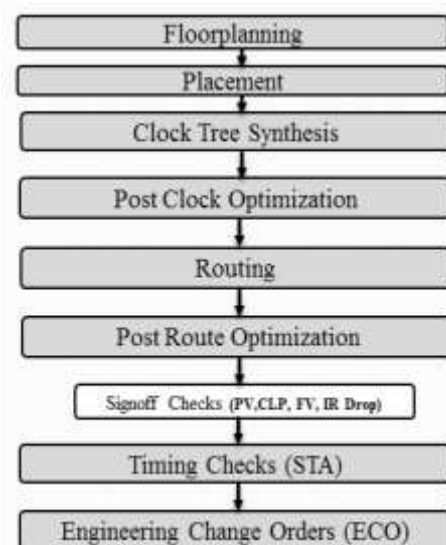


Fig. 2: PD Flow

4 Design and Implementation

This section deals with the design and implementation of the FP, Placement, and CTS stages implemented in the current work. The strategies used in each of the processes of PD flow implementation are described in this section.

4.1 Floorplanning (FP)

A partition level Floorplan approach is employed in the current work. The shape of the sub-HM is decided based on the target utilization and the standard cells in the design. The entire block is divided in the sub- HMs or hierarchical blocks. The sub-HM layout shapes are decided during the partitioning of the top-level block. Each of the hierarchical blocks is implemented separately and the sub-HMs are integrated at the top level to reduce the huge run time and to reduce the burden of improving the PPA. Figure 3 shows the FP flow.

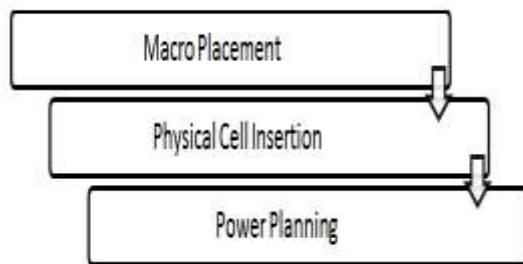


Fig. 3: FP Flow

FP is the process of placing memories/macros/IPs in the core area of the chip. Floor planning includes macro placement, pin placement, physical cell insertion, and Power grid (PG) design.

The quality of chip implementation depends on the quality of the floorplan. Efficient FP requires an understanding of the data flow of design, basic design, integration guidelines for the design, I/O/pin placement requirements, and the hierarchies present in the design. The ideal location for placing the memories is along the boundary of the core area. Thus, the center portion of the core area is completely utilized for placement of standard cells and also reduces the routing complexity. The macros have to be placed such that the pins of the macros face towards the core to ease the communication with the leaf cells placed in the core area. The hierarchy of memories having more leaf cells is not placed close to each other as it may be a prime region of congestion in the later stages. For a memory dominant design, proper spacing between macros must be provided to ensure ease of routing and power grid placements for minimizing

congestion issues. Notches are avoided during the placement of macros, as they gravitate the standard cells towards it in the placement stage.

During the physical cell insertion physical cells like tap cells, end cap cells, tie cells, and filler cells are added to the design. These cells have only power and ground pins and no signal pins. Tap cells are used to avoid the latch-up problem in the design and are placed in a checkerboard fashion in the design. End cap cells are placed in the design to mark the end of the rows and macros in the design. Filler cells are used to ensure the continuity of n-well and implant layers in the standard cell rows. Spare cells may also be added to the design.

During the power planning, horizontal and vertical power stripes are laid across the design to distribute power to the core and the memories. Power switches are laid in the core area in a daisy chain fashion for every defined um, depending on the power scheme used. These power switches tap the power from the power strips and lay out power to the standard cells and memories present in the core area. The power switches convert always-on power to an on-off power. Further, the following pins are present at the top and bottom of each of the standard cell rows to distribute power to the standard cells.

4.2 Placement

During the placement stage, the standard cells are placed in the design. The different criteria used at the placement stage can be congestion-driven, timing-driven, and power optimization, etc. The placement tool not only places the standard cells present in the synthesized verilog netlist but also performs design optimization by fixing setup timing violations that arise due to the placing of FFs having data paths very far from each other causing huge data path delay.

For timing critical designs, placement can be guided by providing more weight to critical path groups. Critical path groups like REG2REG and REG2ICG paths are optimized separately when compared to the low-effort path groups for achieving the best possible WNS for the specified path group, during timing optimization. Timing is majorly given importance because if the timing Quality of Results (QoR) is bad, placement is not qualified. Therefore, it is essential to make timing convergence the key task of placement optimization. The interconnect lengths to be created at the routing stage depend on the placement and it determines the routability of the design, making placement a very critical step in UDSM technology.

Congestion and overflow analysis at the placement stage play a very important role in the convergence of design. Congestion and overflow numbers indicate the congestion in an area. The entire core area is divided into smaller portions called g-cells and the cells are placed in the gcells. Each gcell will have a limited number of routing resources i.e., there will be a limit on the number of nets that can be routed over a particular area. If the number of routing tracks available in a particular area is less than the required routing tracks, then the area is said to be congested. This indicates that a greater number of cells are placed in each area. Congestion map indicates congested regions in the core area. The total hotspot score and the local hotspot scores (should be minimal) are compared with certain bounds, to proceed with the current placement. The pin density map indicates the regions having cells with high pin density. Such cells should not be clustered in a particular region, this results in routing congestion.

The overflow numbers are analyzed to determine the routing congestion. Module hierarchy is viewed at the placement stage by opening the placement database. It must be ensured that there is no module split in the placement of standard cells. For instance, the standard cells belonging to a particular hierarchy 'A' should be placed near memories (if present) belonging to the hierarchy 'A' in the core area. If the standard cells of hierarchy 'A' are sitting in a location far away from the corresponding memories of hierarchy 'A', there is a module split. This might result in congestion as more routing resources are required. The FP has to be revisited to revise the macro placement to ensure that the module split does not occur. Thus, FP plays a very essential role.

Placement is the initial phase where congestion analysis begins and must be under control. A few techniques which can be used to reduce congestion are the usage of placement blockages, Macro padding, cell padding, and use of density screens.

During placement, the standard cells gravitate towards the macro edges, towards the ports, and the notch regions. This causes major issues in the routing stage as the placement of standard cells near the ports will reduce the area and tracks that can be used for routing and results in shorts. Therefore, blockages are placed in such regions. Hard placement blockages are placed the region to be used only for routing. The soft blockage is placed in a region where only buffers can be placed. Partial placement blockage can be placed in the channels created in the core region due to macro placement. E.g., a partial placement blockage of 40% in an area will use only 40% of the area for placement of cells.

Such partial placement blockages are added in areas that are prone to congestion. Due to the placement of standard cells near the memory boundaries, the routing resources are reduced, which may result in shorts. Thus, neat the macro boundaries, layer-wise blockages are placed ie, a layer of soft blockage (buffer only) followed by a layer of partial blockage.

For designs, with a high amount of clustering of cells and cells having high pin density like AndOrInvert (AOI) logic cells, cell padding is used. Where a small additional area is created to the sides of the cell, to allocate more routing area for the pins.

4.3 Clock Tree Synthesis (CTS)

CTS is the process of inserting buffers/inverters along the clock path built from the clock definition point to each of the clock sinks in the design. The main aim of CTS is to build a balanced clock tree to balance clock skew and minimize clock latency to meet timing, constraints, and power requirements. The clock path is balanced such that no setup/hold timing violations occur in the design. Clock buffers /inverters are used for building the clock tree and these buffers are different from the normal buffers/inverters as they have equal rise time and fall time. QoR of the CTS stage decides the timing convergence and power of design.

Conventional CTS is implemented for the design where the clock tree is built starting from the clock source point and the clock tree is structured using an H-tree or conventional path from the clock source (clock root) to each of the clock sinks. The conventional clock tree is built with a single tap point and branched into the clock structure. A H-tree aims to minimize the clock skew but has geometrical symmetry and also has good robustness against variations. H-tree also consumes a considerable amount of wire length and clock power.

4.3.1 Multi Point CTS with Flexible H-Tree (FHT)

Figure 4 shows a block diagram of a Flexible H-tree (FHT) with multi tap point CTS. The top of the clock tree is implemented using a flexible H-tree structure.

The clock tree as indicated in the Figure 5 consists of a root pin. The root pin can be the main clock pin of the block or the output pin of a clock gate cell or clock multiplexer. The driver cell will be a clock buffer or a clock inverter, which will act as repeaters and reproduce the clock at the output of the tap driver. Tap drivers are the sinks of the FHT which act as tap points from where sub-trees are

built. The clock tree is further built from each of these tap drivers to the clock pins of the FFs in the design. Each of the tap drivers will be allocated a set of sinks.

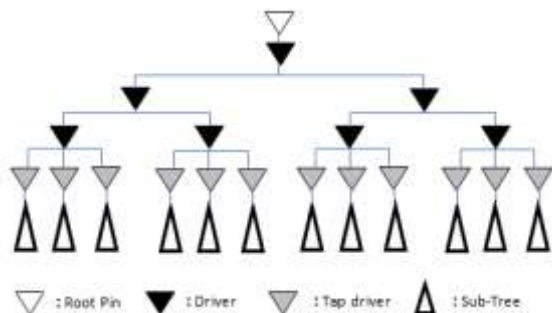


Fig. 4: Flexible H-tree with Multitap point CTS

A traditional H-tree, due to its geometric symmetry, places a constraint on the number of tap drivers (sinks for H-tree) and the location of these sinks. However, it does not place any constraint on the number of tap drivers and their locations. But even with uneven tap driver distributions, to maintain electrical symmetry, many tap drivers have additional delays in their path. Even though the FHT does not guarantee geometrical symmetry, if the floorplan area is rectangular with regular sink grids, the FHT will be electrically and geometrically symmetric. The clock tree is built in each PVT corner and mode. The stable corner, and dominant is chosen for building the clock-tree. The FHT reduces the cross-corner scaling. Conventional CTS obtains balancing in the clock tree by cell insertion, adjusting of wire length, and sizing. For a fast corner, the cell delays of different cell sizes or cell types (buffer and clock gate) scale differently to one another and differently to the RC delay of the connecting wires, leading to skew.

As the FHT is the top of the clock tree and is electrically symmetric, it maintains a nearly perfect skew over all delay corners and thus, the skew at the sinks is correspondingly reduced in the fast corner. As in the FHT, the top of the tree is electrically symmetric it helps in maintaining nearly perfect skew over all delay corners, resulting in reduced skew at the sinks of the fast corner. The FHT provides local buffering and balancing between the structured top (FHT) of the clock tree and the clock sinks. At each clock sink (FF) in the design, both data-path and clock delays are adjusted to improve negative setup timing slack specifically the high-effort path group(s) WNS.

The procedure followed for the building of FHT is explained hereafter. Initially, the clock for which FHT is built, the corresponding clock pin is placed at the center of the core area. The clock cells related

to the clock signal are pre-placed in a bounded region surrounding the clock pin present at the center. The pre-placement of the cells happens in the Placement stage. Further, the core area is divided into grids based on the dimensions of the core area. The grid may be a 3X2 grid, ie, the core area is divided into 6 equal grids. Further, the tap point's position can be provided manually, if not, is allocated by the tool. The tap points are usually placed at the center of each of the grids. This reduces the clock local skew within the grid and helps reduce the clock latency. The top FHT is built using higher metal layers and Non-Default Routing is applied to the clock nets. The clock inverters, and clock buffers to be used for building the clock tree are listed. The PVT corner and mode to be used for building the clock tree are specified. Further, the clock transition is constrained to be a fixed value at the top, trunk, and leaf level, to maintain the slew and build a slew-aware clock tree.

Each of the tap drivers are considered to be physically distinct clock tree roots (o/p pins of the tap drivers) from where the clock tree is built within each of the grids. The allocation of clock sinks to each of the sub-trees under each of the tap points is done by the tool. The clock cells (clock gating cells (cgc) or clock logic) common to multiple sinks, present under different tap drivers are cloned in each of the tap driver grids. The tool creates one clock tree group for each clock under each constraint mode. A Skew target can be set for each clock tree group. Each clock tree group may have one or more sources and several sinks. Skew and insertion delay targets can be set for each of the clock tree groups. Ignore pins are defined, if necessary, in each of the clock tree groups, such that the specific clock tree group does not propagate beyond the pin. Global skew balancing aims to achieve an equal delay, based on the target skew, from the source to all sinks within each clock tree group. The maximum distance from the root pin to each of the tap drivers can be specified. The Further process involved in the PD flow like post-cts optimizations, routing, and post-route optimizations are carried out on the design.

5 Results and Discussion

The strategies mentioned in the previous section are implemented in an industrial design with more than a million instances. The experimental results obtained by implementing the strategies on the industrial designs are discussed in this section.

The Conventional Clock tree is built in the design as indicated in Figure 5. The Clock tree is

built starting from the clock source point and the clock tree is structured using an H-tree or conventional path from the clock source (clock root) to each of the clock sinks.

The cells in green indicate the clock drivers and the cells in red indicate the clock sinks in the design. The triangles along the clock path from one driver to another driver are the clock inverters (cinv) or clock buffers (cbuf) used to balance the clock tree.

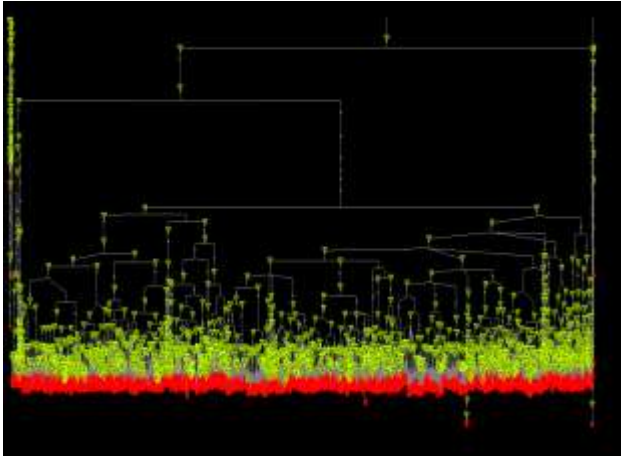


Fig. 5: Conventional Clock tree

Figure 6 shows the FHT with a multi-tap point built into the design. The clock tree consists of the top built with FHT with the clock pin at the center of the core.

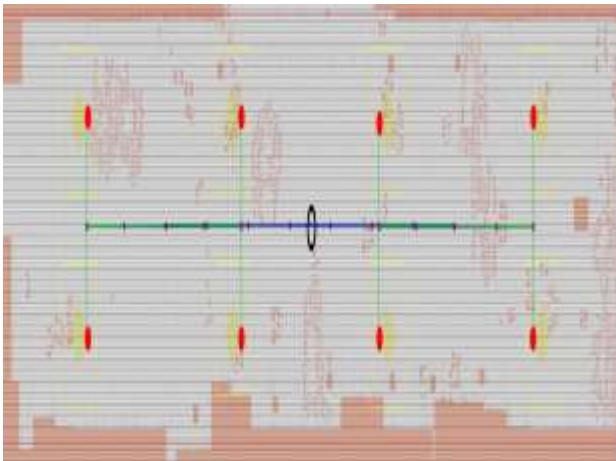


Fig. 6: FHT with Multitap point CTS

As seen in the design the core area is divided into a 4x2 grid indicated by the green parallel lines. The black circle at the center indicates the root pin of the clock tree. The clock tree is built from the root pin and is branched to each of the tap points indicated as a red dot. The purple dots along the path from the root pin to the tap drivers are the clock buffers, Figure 7. The purple points indicate the tap drivers and the root pins.

As seen in Figure 7, the top FHT consists of one root pin and 8 tap drivers (FHT sink pins). The cells in pink are the clock drivers and they are majorly clock inverters. The cells in orange indicate the clock buffers which are used for balancing the clock tree. The cells in yellow are the clock gate cells used in the design. The cells in light blue are the clock sinks present in the design.

The implementation of the strategies discussed for FP and Placement along with the implementation of FHT with Multi tap point CTS shows better QoR at the end of each of the stages.

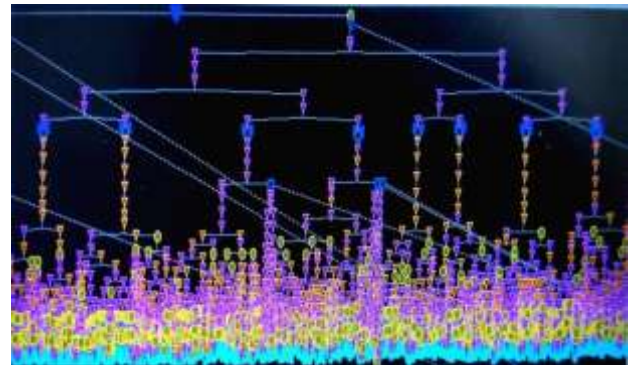


Fig. 7: Clock structure using FHT with Multitap point CTS

As indicated in Table 1, timing comparison for setup and hold violations are done for REG2REG paths in terms of WNS at CTS and Post-Route Optimization stage (PRO) stages. DB1 indicates the database implemented with default CTS and DB2 indicates the database implemented with FHT and Multi tap point CTS. As shown in Table 1, the huge reduction is seen in terms of WNS for DB2.

Table 1. Timing Metric (WNS)

Stage	DB	Setup	Hold
		WNS (ps)	WNS (ps)
CTS	DB1	145	211
	DB2	27	125
PRO	DB1	139	201
	DB2	24	55

Table 2. Timing Metric Reduction (% Reduction)

	Setup timing (% reduction)			Hold timing (% reduction)		
	WNS	TNS	NFE	WNS	TNS	NFE
CTS	81.3	53.3	6	40.75	63.9	27.33
Post-Route	82.73	95.1	95.5	72.63	87.3	49.97

Table 2, indicates the percentage reduction in terms of WNS, TNS, and NFE. The table lists the percentage difference in the timing metrics between DB1 and DB2. For instance, the setup WNS is reduced by 81.3% from DB1 to DB2 at the CTS stage. Hold timing metrics are the most important ones at the CTS stage. As seen in Table 2, there is a reduction of 40.47% and 63.9% in terms of hold WNS and hold TNS respectively from DB1 to DB2. The NFE is reduced by 27.33% from DB1 to DB2 which is a drastic reduction in the failing end points. Thus, a huge improvement is seen in terms of timing in DB2 when compared to DB1. The database with FHT and multi-tap point CTS shows huge improvement in terms of timing.

The same results are seen at the PRO stage. DB2 has a 72.63% reduction in Hold WNS when compared to DB1. Further, the NFEs are reduced by 50% at the PRO stage. This means that the PRO with conventional CTS has 50% more failing endpoints in hold violations when compared to the DB2.

Table 3. Clock Metrics at CTS stage

Database	Max ID (ns)	Min ID (ns)	Avg ID (ns)	Global Skew (ns)	Local Skew (ns)
DB1	1.431	1.192	1.470	0.239	0.19
DB2	1.117	0.915	1.073	0.202	0.152

DB1 indicates the database implemented with conventional CTS and DB2 indicates the database implemented with FHT and Multi tap point CTS. Table 3 indicates the clock metrics like Max ID, Min ID, global skew, local skew, and Avg ID, explained in section 2. As indicated in Table 3, clock latency (Avg ID) is reduced by 27% in DB2 as compared to DB1. Global Skew is reduced by 15.48% and local skew by 20%.

6 Conclusion

Experimental results on an industrial design having more than a million instances show that the proposed clock tree structure comprising of FHT with Multi tap point CTS (DB2) shows huge improvement in terms of timing and clock metrics when compared to the conventional CTS (DB1). A reduction of 40.47% and 63.9% is seen in terms of hold WNS and hold TNS respectively from DB1 to DB2. The NFE is reduced by 27.33% from DB1 to DB2. DB2 has a clock latency (Avg ID) of 1.073 ns which is 27% lesser than that of DB1. Global Skew is reduced by 15.48% and local skew by 20%.

References:

- [1] Subhendu Roy, Pavlos M. Mattheakis, Laurent Masse-Navette and David Z. Pan, "Clock Tree Resynthesis for Multi-Corner Multi-Mode Timing Closure", *IEEE Transactions On Computer-Aided Design Of Integrated Circuits And Systems*, vol. 34, no. 4, April 2015.
- [2] Yinan Zhang and Xiaohong Peng, "A Partition Level Floorplan Method Based on Data Flow Analysis for Physical Design of Digital IC", *2nd International Conference on Integrated Circuits and Microsystems*, pp. 74-77, 2017.
- [3] Vikram Gautam and Pawan Kumar Dahiya, "IC Design Physical Verification", *International Research Journal of Engineering and Technology (IRJET)*, vol.4, no.6, June 2017.
- [4] R. A. Wahab, R. Md. F. T. Aziz, N. Othman, S. Saleh, N. Razali, M. A. B. Z. Abidin and M. H. Md. Nasir, "Physical Verification Flow on Multiple Foundries" *International Journal of Electronics and Communication Engineering*, vol.9, no.10, 2015.
- [5] Subhendu Roy, Pavlos M. Mattheakis, Laurent Masse-Navette and David Z. Pan, "Evolving Challenges and Techniques for Nanometer SoC Clock Network Synthesis", *12th IEEE Conference on Solid-state and Integrated Circuit Technology*, Guilin, China, 2014.
- [6] Kishore Kollu, Trey Jackson, Farhad Kharas, and Anant Adke, "Unifying Design Data During Verification: Implementing Logic-Driven Layout Analysis and Debug", *IEEE International Conference on IC Design & Technology*, 2012.
- [7] Tomas Figliolia and Andreas G. Andreou, "The Conical-Fishbone Clock Tree: A Clock-Distribution Network for a Heterogeneous Chip Multiprocessor AI Chiplet", *22nd Euromicro Conference on Digital System Design*, 2019.
- [8] Soheil N Shahsavani and Massoud Pedram, "A Minimum-Skew Clock Tree Synthesis Algorithm for Single Flux Quantum Logic Circuits", *IEEE Transactions on Applied Superconductivity*, vol. 29, no. 8, 2019.
- [9] W. Liu, C. Sitik, E. Salman, B. Taskin, S. Sundareswaran and B. Huang, "SLECTS: Slew-Driven Clock Tree Synthesis", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 27, no. 4, 2019.
- [10] H. Kao, Y. Lee, S. Huang, W. Cheng, and Y. Chou, "An industrial design methodology for

the synthesis of OCV-aware top-level clock tree”, *6th International Symposium on Next Generation Electronics (ISNE)*, Keelung, 2017.

- [11] Yici Cai, Chao Deng, Qiang Zhou, Hailong Yao, FeifeiNiu, and Cliff N. Sze, “Obstacle-Avoiding and Slew-Constrained Clock Tree Synthesis With Efficient Buffer Insertion”. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 23, no. 1, 2015.
- [12] Sreevidya.S, Ravishankar Holla, Roopaka Raghu, “Low Power Physical Design and Verification in 16nm FinFET Technology”, *Proceedings of the Third International Conference on Electronics Communication and Aerospace Technology (ICECA)*, pp. 936-940, 2019.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

The authors equally contributed in the present research, at all stages from the formulation of the problem to the final findings and solution.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

No funding was received for conducting this study.

Conflict of Interest

The authors have no conflicts of interest to declare.

Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)

This article is published under the terms of the Creative Commons Attribution License 4.0

https://creativecommons.org/licenses/by/4.0/deed.en_US